

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

راهنمای هنرآموز توسعه برنامه‌سازی و پایگاه داده

رشته شبکه و نرم‌افزار رایانه

گروه برق و رایانه

شاخه آموزش فنی و حرفه‌ای

پایه یازدهم دوره دوم متوسطه



وزارت آموزش و پرورش
سازمان پژوهش و برنامه‌ریزی آموزشی

| | |
|-----------------------------------|---|
| نام کتاب: | راهنمای هنرآموز توسعه برنامه‌سازی و پایگاه داده - ۲۱۱۸۱۶ |
| پدیدآورنده: | سازمان پژوهش و برنامه‌ریزی آموزشی |
| مدیریت برنامه‌ریزی درسی و تألیف: | دفتر تألیف کتاب‌های درسی فنی و حرفه‌ای و کار دانش |
| شناسه افزوده برنامه‌ریزی و تألیف: | حسن‌رضا آرش‌نیا، عبدالمهدی بحرانی، مازیار خشنودپور، رحمن رحمن‌نژاد، سروش رستمی گوران، قربانعلی عربی (اعضای گروه تألیف) بتول حاجتی، صدیقه رسولی، افسانه رضایی، شیرین شعبانی، محسن عبدالهی علی بیگ، زهرا عسگری رکن‌آبادی، سارا غایی، گیتی قربانی، محمدرضا قشونی، مهناز کارکن، مهدی کیانی، محمدرضا یمقانی (اعضای شورای برنامه‌ریزی) |
| مدیریت آماده‌سازی هنری: | اداره کل نظارت بر نشر و توزیع مواد آموزشی |
| شناسه افزوده آماده‌سازی: | مجید ذاکری یونسسی (مدیر هنری) - ایمان اوجیان (طراح یونیفورم) - مریم پورغلامی (طراح جلد) - محمدعلی شاه میرزندی (صفحه‌آرا) |
| نشانی سازمان: | تهران: خیابان ایرانشهر شمالی، ساختمان شماره ۴ آموزش و پرورش (شهید موسوی) تلفن: ۹-۸۸۸۳۱۱۶۱، دورنگار: ۸۸۳۰۹۲۶۶، کد پستی: ۱۵۸۴۷۴۷۳۵۹ وبسایت: www.chap.sch.ir |
| ناشر: | شرکت چاپ و نشر کتاب‌های درسی ایران: تهران - کیلومتر ۱۷ جاده مخصوص کرج - خیابان ۶۱ (دارو پخش) تلفن: ۴۴۹۸۵۱۶۱-۵ دورنگار: ۴۴۹۸۵۱۶۰، صندوق پستی: ۱۳۹ - ۳۷۵۱۵ |
| چاپخانه: | شرکت چاپ و نشر کتاب‌های درسی ایران «سهامی خاص» |
| سال انتشار و نوبت چاپ: | چاپ دوم ۱۳۹۷ |

کلیه حقوق مادی و معنوی این کتاب متعلق به سازمان پژوهش و برنامه‌ریزی آموزشی وزارت آموزش و پرورش است و هرگونه استفاده از کتاب و اجزاء آن به‌صورت چاپی و الکترونیکی و ارائه در پایگاه‌های مجازی، نمایش، اقتباس، تلخیص، تبدیل، ترجمه، عکس‌برداری، نقاشی، تهیه فیلم و تکثیر به هر شکل و نوع بدون کسب مجوز ممنوع است و متخلفان تحت پیگرد قانونی قرار می‌گیرند.

ISBN 978-964-05-2921-8

شابک: ۹۷۸_۹۶۴_۰۵_۲۹۲۱_۸



دست توانای معلم است که چشم انداز آینده ما را ترسیم می کند.

- ۲.....پودمان اول: پیاده‌سازی پایگاه داده.....
واحد یادگیری ۱: شایستگی ایجاد پایگاه داده
واحد یادگیری ۲: شایستگی توسعه پایگاه داده
- ۵۷.....پودمان دوم: مدیریت مجموعه داده.....
واحد یادگیری ۳: شایستگی کار با ساختار تکرار
واحد یادگیری ۴: شایستگی کار با آرایه
- ۱۱۰.....پودمان سوم: طراحی واسط گرافیکی.....
واحد یادگیری ۵: شایستگی ایجاد واسط گرافیکی کاربر
واحد یادگیری ۶: شایستگی کار با کنترل‌های پیشرفته
- ۱۴۴.....پودمان چهارم: توسعه واسط گرافیکی.....
واحد یادگیری ۷: شایستگی کار با ماوس و منو
واحد یادگیری ۸: شایستگی کار با صفحه کلید
- ۱۹۰.....پودمان پنجم: مدیریت پایگاه داده.....
واحد یادگیری ۹: شایستگی مدیریت پایگاه داده

فهرست فیلم‌ها

■ پودمان اول

- فیلم 111h1 : تاریخچه پایگاه داده
- فیلم 111h2 : مفاهیم پایگاه داده
- فیلم 111h3 : آشنایی با زبان SQL
- فیلم 111h4 : پیاده‌سازی پایگاه داده در Access – ۱
- فیلم 111h5 : پیاده‌سازی پایگاه داده در Access – ۲
- فیلم 111h6 : بررسی پودمان اول

■ پودمان دوم

- فیلم 111h7 : کار حلقه WHILE
- فیلم 111h8 : کار حلقه FOR
- فیلم 111h9 : کار با BREAK
- فیلم 111h10 : برنامه ماکزیم
- فیلم 111h11 : کار با حلقه‌های متداخل
- فیلم 111h12 : تعریف آرایه
- فیلم 111h13 : کاربرد آرایه
- فیلم 111h14 : ویژگی length در آرایه
- فیلم 111h15 : مرتب سازی آرایه
- فیلم 111h16 : جستجوی خطی
- فیلم 111h17 : جستجوی دودویی
- فیلم 111h18 : کلاس array

■ پودمان سوم

- فیلم 111h19 : مفاهیم اصلی در پروژه ویندوزی
- فیلم 111h20 : نحوه دسترسی به اطلاعات رویدادها
- فیلم 111h21 : ایجاد یک پروژه ویندوزی
- فیلم 111h22 : آشنایی با کنترل فرم
- فیلم 111h23 : ایجاد اولین فرم
- فیلم 111h24 : دو پروژه ویندوزی
- فیلم 111h25 : ایجاد برنامه‌های ویندوزی با واکنش به رویدادها-۱
- فیلم 111h26 : ایجاد برنامه‌های ویندوزی با واکنش به رویدادها-۲
- فیلم 111h27 : ایجاد برنامه‌های ویندوزی با واکنش به رویدادها-۳
- فیلم 111h28 : کار با کنترل کادر تصویر
- فیلم 111h29 : حرکت اشیا روی صفحه نمایش
- فیلم 111h30 : طراحی بازی حدس تصویر
- فیلم 111h31 : استفاده از کادر محاوره ای انتخاب فایل
- فیلم 111h32 : استفاده از کادر های محاوره ای انتخاب رنگ و فونت
- فیلم 111h33 : کادر محاوره‌ای رنگ
- فیلم 111h34 : ساختار شمارشی MessageBoxButton
- فیلم 111h30 : طراحی بازی حدس تصویر
- فیلم 111h31 : استفاده از کادر محاوره ای انتخاب فایل

فهرست فیلم‌ها

- فیلم 111h32 : استفاده از کادر های محاوره ای انتخاب رنگ و فونت
- فیلم 111h33 : کادر محاوره‌ای رنگ
- فیلم 111h34 : ساختار شمارشی MessageBoxButton

پودمان چهارم

- فیلم 111h35 : استفاده از زمان سنج ، راه اندازی و غیرفعال کردن زمان سنج
- فیلم 111h36 : کادر پیام – کاربرد تایمر – تولید اعداد تصادفی
- فیلم 111h37 : کار با کنترل CheckBox
- فیلم 111h38 : کنترل RadioButton و ساختار DialogResult و GroupBox و Panel
- فیلم 111h39 : ایجاد پروژه پذیرش هتل-۱
- فیلم 111h40 : تکمیل پروژه پذیرش هتل-۲
- فیلم 111h41 : تکمیل پروژه پذیرش هتل-۳
- فیلم 111h42 : تکمیل پروژه پذیرش هتل-۴
- فیلم 111h43 : تکمیل پروژه پذیرش هتل-۵
- فیلم 111h44 : تکمیل پروژه پذیرش هتل-۶
- فیلم 111h45 : تکمیل پروژه پذیرش هتل-۷

پودمان پنجم

- فیلم 111h46 : پروژه تالار گفتگو
- فیلم 111h47 : پروژه شبیه ساز سیستم خودپرداز بانکی
- فیلم 111h48 : برنامه نقاشی
- فیلم 111h49 : برنامه فرم جدول مشتری
- فیلم 111h50 : برنامه انتخابات شورای دانش آموزی
- فیلم 111h51 : توسعه پروژه فروشگاه

فهرست فیلم‌های جدید

پودمان سوم

- فیلم 111h52 : انتقال اطلاعات بین فرم‌ها

سخنی با هنرآموزان گرامی

کتاب درسی و کتاب همراه هنرجو به همراه کتاب راهنمای هنرآموز از جمله اجزای بسته آموزشی تلقی می‌شوند که این بسته را سایر اجزا مانند فیلم و نرم‌افزار و ... کامل می‌کند. کتاب راهنمای هنرآموز جهت ایفای نقش تسهیل‌گری، انتقال‌دهنده و مرجعیت هنرآموز در نظام آموزشی طراحی و تدوین شده است. این کتاب براساس کتاب درسی **توسعه برنامه‌سازی و پایگاه داده** پایه یازدهم رشته تحصیلی - حرفه‌ای شبکه و نرم‌افزار رایانه تنظیم شده و دارای پودمان‌های ۱- **پیاده‌سازی پایگاه داده ۲** - مدیریت مجموعه داده ۳ - **طراحی واسط گرافیکی ۴** - توسعه واسط گرافیکی و ۵- مدیریت پایگاه داده است.

هنرآموزان گرامی در هنگام مطالعه این کتاب به موارد ذیل توجه فرمایند: در کتاب راهنمای هنرآموز مواردی از قبیل نمونه طرح درس، راهنما و پاسخ فعالیت‌های یادگیری و تمرین‌ها، ایمنی و بهداشت فردی و محیطی، نکات آموزشی شایستگی‌های غیر فنی، اشتباهات و مشکلات رایج در یادگیری هنرجویان، منابع یادگیری، نکات مهم هنرآموزان در اجرا، فرآیند اجرا و آموزش در محیط یادگیری، بودجه‌بندی زمانی و صلاحیت‌های حرفه‌ای و تخصصی هنرآموزان و دیگر موارد آورده شده است.

ارزشیابی در درس توسعه برنامه‌سازی و پایگاه داده بر اساس ارزشیابی مبتنی بر شایستگی است، این درس شامل ۵ پودمان است و برای هر پودمان، ارزشیابی مستقل از هنرجو صورت می‌گیرد. همچنین یک نمره مستقل برای هر پودمان ثبت خواهد شد. این نمره شامل یک نمره مستمر و یک نمره شایستگی است.

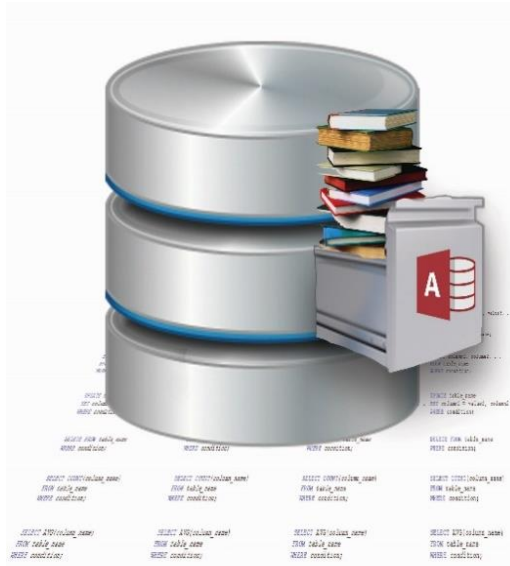
ارزشیابی از پودمان‌های این درس مطابق با جداول استاندارد‌های ارزشیابی پیشرفت تحصیلی تهیه شده توسط دفتر تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش سازمان پژوهش و برنامه‌ریزی آموزشی صورت می‌گیرد.

زمانی هنرجو در این درس، قبول اعلام می‌شود که در هر پنج پودمان درس، حداقل نمره ۱۲ را کسب نماید. در این صورت میانگین نمره‌های پنج پودمان به عنوان نمره پایانی درس در کارنامه تحصیلی هنرجو منظور خواهد شد.

ارزشیابی مجدد در پودمان یا پودمان‌هایی که حداقل نمره مورد نظر در آن کسب نشده است با برنامه‌ریزی هر هنرستان، انجام می‌شود و چنانچه هنرجو به هر دلیلی تا پایان خردادماه شایستگی لازم را در یک یا چند پودمان کسب ننماید، می‌تواند تا پایان سال تحصیلی برای ارزشیابی مجدد در ارزشیابی مبتنی بر شایستگی شرکت نماید.

مسلماً اجرای مطلوب برنامه‌های درسی، نیازمند مساعدت و توجه ویژه هنرآموزان عزیز و بهره‌مندی از صلاحیت‌ها و شایستگی‌های حرفه‌ای و تخصصی مناسب ایشان است.

دفتر تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش



پودمان اول

واحد یادگیری ۱ و ۲

پیاپاده‌سازی پایگاه داده

واحد یادگیری ۱

شایستگی ایجاد پایگاه داده

مقدمات تدریس

الف) مفاهیم کلیدی واحد یادگیری ۱ و ۲

| مفاهیم کلیدی | | | |
|---------------|-------------|-----------------|--------------------|
| محیط عملیاتی | پایگاه داده | بازیابی اطلاعات | ذخیره‌سازی اطلاعات |
| ارتباط | کلید اصلی | صفات موجودیت | موجودیت |
| جامعیت ارجاعی | کلید خارجی | رکورد | فیلد |
| SQL | گزارش | فیلتر کردن | پرس و جو |

ب) تجهیزات لازم

الزامات نرم‌افزاری:

برای تدریس این واحد یادگیری به نرم‌افزار زیر نیاز است:

- Microsoft Office 2016

تجهیزات سخت‌افزاری:

مشخصات سخت‌افزاری مورد نیاز برای نصب Microsoft Office 2016

- **CPU:** 1 gigahertz (GHz) or faster x86-bit or x64-bit processor with SSE2
- **OS:** Microsoft Windows 7, Windows 8, Windows Server 2008 R2, or Windows Server 2012
- **RAM:** 1 GB of RAM (3 GB recommended) for 32 bit; 2 GB of RAM for 64 bit
- **Hard disk:** 3 GB of available hard-disk space for installation;
- **Display:** 1024 x 768 display
- **Graphics:** Graphics hardware acceleration requires a DirectX 10 graphics card

ج) بودجه‌بندی

| جلسه | واحد بودجه‌ری | کارگاه (موضوع) | شماره صفحات | اهداف توانمندسازی | فعالیت‌های تکمیلی |
|------|------------------|---|----------------|---|--|
| ۱ | ۱ | تاریخچه ذخیره‌سازی و بازیابی اطلاعات، مفاهیم پایگاه داده | ۳-۱۰ | درک لزوم استفاده از پایگاه داده، شناسایی موجودیت‌های مطرح در یک محیط عملیاتی، تعیین صفات موجودیت‌ها، تعیین صفت شناسه (کلید) | ارائه یک روزنامه دیواری توسط هنرجویان: - انتخاب یک محیط دلخواه برای شروع یک پروژه عملی (تقسیم هنرجویان به چند گروه) |
| ۲ | ۱ | مفاهیم نمودار ER | ۱۱-۱۵ | شناسایی ارتباط بین موجودیت‌ها، تعیین ماهیت (چندی ارتباطها)، رسم نمودار ER | رسم نمودار ER محیط عملیاتی در هر گروه |
| ۳ | ۱ | ایجاد پایگاه داده - طراحی جدول جدید در پایگاه داده | ۱۵-۲۳ | ایجاد یک پایگاه داده - توانایی ایجاد جدول و تعیین نوع داده فیلدها - آشنایی با ویژگی‌های فیلدها- تعیین کلید اصلی | ایجاد پایگاه داده محیط عملیاتی موردنظر در Access و پیاده‌سازی جدول‌های مرتبط |
| ۴ | ۱ | تغییر ساختار جدول - تعیین کلید خارجی و ارتباطها | ۲۴-۲۸ | توانایی ایجاد تغییر در ساختار جدول و ایجاد ارتباط بین جدول‌ها، پیاده‌سازی کلید خارجی و کنترل یکپارچگی | برقراری ارتباط نهایی بین جدول‌ها در محیط نرم‌افزار Access و بررسی و کنترل یکپارچگی |
| | ۲ | اضافه کردن رکورد به جدول به‌صورت دستی و به کمک دستورات SQL | ۳۱-۳۴ | آشنایی با نحوه اضافه کردن رکورد به جدول به‌صورت دستی، درج سطر جدید به کمک دستور INSERT و رفع خطاهای احتمالی (تکراری بودن مقدار صفت کلید) | درج اطلاعات در جدول‌ها به کمک INSERT، کپی اطلاعات یک جدول در جدول همسان دیگر بررسی خطاهای احتمالی در صورت تکرار کلید و کنترل جامعیت |
| ۵ | ۲ | ویرایش و حذف رکوردها به‌صورت دستی و به کمک دستورات SQL | ۳۴-۳۸ | - آشنایی با نحوه ویرایش و اصلاح اطلاعات ثبت شده در جدول‌ها به‌صورت دستی - ویرایش رکوردها با دستور UPDATE و نحوه اعمال شرط با عبارت WHERE - آشنایی با نحوه حذف رکوردها به‌صورت دستی - حذف رکوردها با دستور DELETE و نحوه اعمال شرط - بررسی یکپارچگی در اثر حذف و ویرایش و رفع خطاهای احتمالی | - طراحی سناریوهای ویرایش و حذف اطلاعات بر اساس نیاز کاربران و اجرای آنها با دستورات UPDATE و DELETE - حذف رکوردهای اصلی و مشاهده اثر آن در جداول وابسته و کنترل یکپارچگی - ویرایش مقدار صفات کلید در رکوردهای جدول اصلی و مشاهده اثر آن در جداول وابسته و کنترل یکپارچگی |
| ۶ | ۲ | - ایجاد پرس‌وجو - ایجاد پرس‌وجو با استفاده از دستور SELECT | ۳۸-۴۴ | - آشنایی با نحوه ایجاد پرس‌وجو با استفاده از wizard و با استفاده از دستور SELECT - توانایی مرتب‌سازی رکوردها توسط عبارت ORDER BY - توانایی استخراج اطلاعات از چند جدول | - طراحی سناریوهای استخراج اطلاعات از سیستم بر اساس نیازهای اطلاعاتی کاربر سیستم و طرح سوالات مرتبط - پاسخ به سوالات با دستور SELECT |
| ۷ | ۲ | گروه‌بندی نتایج پرس‌وجو با استفاده از توابع تجمعی | ۴۴-۴۸ | - توانایی گروه‌بندی و دسته‌بندی اطلاعات و انجام محاسبات به کمک عبارت GROUP BY | - طراحی پرس‌وجوهایی برای انجام محاسبات و دسته‌بندی اطلاعات مانند محاسبه سود حاصل از فروش کالاها و ... |
| | ۲ | ایجاد و ویرایش گزارش | ۴۸-۵۳ | آشنایی با نحوه ایجاد گزارش با استفاده از Wizard - آشنایی با روش‌های ویرایش گزارش با کمک نمای Design | - طراحی گزارش از سیستم بر اساس سناریوها و آماده‌سازی برای چاپ |

طرح درس هفتگی (۸ ساعته) پیشنهادی

| پایه: یازدهم | | درس: تاریخچه ذخیره سازی و بازیابی اطلاعات و مفاهیم پایگاه داده | |
|--|---|---|-------------|
| پیام جلسه (هدف کلی): هنرجو بتواند مفاهیم کلیدی پایگاه داده را درک کند. | | | |
| اهداف یادگیری | فعالیت‌ها | زمان فیزیکی | |
| فعالیت | کار هنرآموز | کار هنرجو | مدت (دقیقه) |
| ارزشیابی رفتاری ورودی | <p>- نمایش پویانمایی شماره ۱۱۱۰۱</p> <p>- طرح سوالاتی در مورد ثبت و نگهداری اطلاعات توسط انسان‌های اولیه، تفاوت یک پرونده کاغذی با یک پرونده کلمپیوتری، چگونه یک فروشنده حساب ماهانه خود را نگه می‌دارد، چگونه می‌توانیم اطلاعات پایگاه داده را طراحی و ایجاد کنیم؟</p> | <p>- انجام فعالیت گروهی صفحه ۳ و ۴</p> <p>- ارائه پاسخ به سوالات مطرح شده</p> <p>- نام بردن مکان‌هایی که در آن از رایانه برای نگهداری اطلاعات استفاده می‌شود.</p> | ۶۰ |
| ایجاد انگیزه | <p>- شرح مشکلات عدم استفاده از پایگاه داده، افزونگی، ناسازگاری، مزایای پایگاه داده، معایب پایگاه داده، مقدمات طراحی پایگاه داده (از کجا شروع کنیم)</p> | <p>- انجام فعالیت صفحه ۵ و کنجکاوای صفحه ۶</p> <p>- شرکت در پرسش و پاسخ</p> | ۳۰ |
| ارائه مفاهیم کلیدی (توضیح هنرآموز) | <p>توضیح مفاهیم کلیدی پایگاه داده (همراه مثال): محیط عملیاتی، موجودیت، معیارها و ضوابط انتخاب موجودیت‌های مطرح، صفات موجودیت، معیار انتخاب صفات مهم، صفت شناسه</p> | <p>-انجام کنجکاوای و فعالیت‌های صفحه ۷ تا ۹</p> <p>- تقسیم شدن در چند گروه مختلف و مشارکت در موضوع مطرح شده</p> | ۹۰ |
| فعالیت کارگاهی (تمرین هنرجویان) | <p>در نظر گرفتن محیط عملیاتی مدرسه</p> <p>هنرآموز توضیح کلی در مورد فعالیت گروهی به هنرجو داده و از آن‌ها می‌خواهد به‌صورت گروهی به حل آن‌ها بپردازند.</p> | <p>هنرجویان با توجه به فعالیت‌های گروهی کتاب، به پیاده‌سازی مفاهیم کلیدی برای محیط عملیاتی خود می‌پردازند.</p> | ۶۰ |
| ارزیابی فعالیت‌ها ارائه تمرین | <p>شناخت موجودیت‌های محیط عملیاتی درج‌شده در کتاب</p> | <p>از هنرجویان خواسته می‌شود در گروه‌های تعیین شده به انجام یکی از فعالیت‌های کارگاهی بپردازند. قبلاً ملاک‌های ارزیابی تمرین‌ها به آن‌ها داده می‌شود.</p> | ۳۰ |

| طرح درس هفتگی (۸ ساعته) پیشنهادی | | | |
|--|---|--|---|
| پایه: یازدهم | | درس: تاریخچه ذخیره سازی و بازیابی اطلاعات و مفاهیم پایگاه داده | |
| پیام جلسه (هدف کلی): هنرجو بتواند مفاهیم کلیدی پایگاه داده را درک کند. | | | |
| اهداف یادگیری | فعالیت‌ها | زمان فیزیکی | |
| ارائه نکات تکمیلی (جمع‌بندی) | از هنرجویان بخواهید به بررسی صحت عملیات گروه مقابل بپردازند. مفاهیم کلی مطرح شده را مرور کنید. | ۴۵ | هنرجویان هر گروه، باید با توجه، دقت و پرسش و پاسخ، تک‌تک مفاهیم پیاده‌سازی شده در پروژه گروه مقابل را بررسی کنند و مشکلات موجود و یا پیشنهادهای خود را ارائه دهند. |
| ارزشیابی شایستگی (ارزشیابی پایانی) | این ارزشیابی در دوشاخه انفرادی و گروهی انجام می‌شود: از هنرجویان خواسته می‌شود در گروه‌های تعیین شده به انجام پروژه این بخش از واحد کار بپردازند. نمره‌ای به کارگروهی هنرجویان داده می‌شود. آزمون به‌صورت انفرادی برگزار می‌شود. قبل از شروع به کار، ملاک‌های ارزشیابی پروژه به آن‌ها داده می‌شود. نمره گروهی در نمره انفرادی افراد تأثیرگذار است. | ۳۰ | هنرجویان در گروه‌های خود سعی می‌کنند در مدت‌زمان تعیین شده با نهایت دقت و سرعت با ایجاد خلاقیت به حل پروژه بپردازند. هنرجویان به‌صورت انفرادی یک محیط عملیاتی را بررسی می‌کنند. |
| تمرین در منزل (تعیین تکلیف) | محیط‌های عملیاتی دیگر برای هر گروه تعیین شوند و هنرجویان به‌صورت گروهی، جزوه آموزشی برای یک محیط عملیاتی بسازند. حتماً موجودیت‌ها و صفات آن‌ها و کلید هر موجودیت تعیین شود. | ۱۵ | تقسیم‌بندی وظایف کار پروژه بین اعضای گروه و تحویل در زمان‌بندی تعیین شده |
| ابزارهای موردنیاز | ویدئو پروژکتور، رایانه، تخته آموزشی | | |

در گذشته‌های نه‌چندان دور نگهداری و ثبت اطلاعات به شیوه‌های مختلفی انجام می‌شد. مانند استفاده از سنگ‌نویسی و پوست تا اختراع کاغذ. نکته حائز اهمیت نگهداری اسناد و اطلاعات این بود که امکان از بین رفتن آنها توسط عواملی مانند آتش‌سوزی، پاک شدن جوهر به دلیل گذشت زمان، فرسوده شدن کاغذ، سرقت و ... بسیار زیاد بود. پس باید به این فکر می‌کردیم که چگونه این اطلاعات را محافظت کنیم؟

با پیدایش رایانه‌ها، پرونده‌ها بدین منظور ایجاد شده‌اند که قابلیت ذخیره و پشتیبانی از اطلاعات را داشته باشند. امروزه در طراحی یک برنامه یا یک تارنمای کاربردی ساده یا پیشرفته، ناگزیر از استفاده پایگاه داده در برنامه خود هستیم.

حال به پاسخ پرسش‌های زیر فکر کنید و بگویید راه حل چیست:

• آیا می‌توانید حدس بزنید که اطلاعات ثبت‌نام و کارنامه هر هنرجو چگونه نگهداری می‌شود؟

• کدام‌یک از شما از فروشگاه‌هایی خرید کرده‌اید که با استفاده از بارکدخوان، قیمت کالا و مبلغ پرداختی را محاسبه می‌کنند؟ آیا می‌توانید حدس بزنید این کار چگونه انجام می‌شود؟

• اگر حسابدار یک فروشگاه بزرگ، بخواهد در پایان هر روز یا ماه، میزان فروش‌ها، موجودی کالاها، سود (زیان) و ... را به کمک قلم و کاغذ و ماشین حساب محاسبه و کنترل کند، چه مشکلاتی ممکن است پیش آید؟

• اگر شما مسئول کتابخانه مدرسه خود باشید، چگونه به پرسش یک مراجع در موجود بودن یک کتاب یا کتاب‌های یک نویسنده خاص پاسخ می‌دهید؟

• امروزه چگونه می‌توانیم این حجم وسیع از اطلاعات را ذخیره، سازماندهی، دسته‌بندی و مدیریت کنیم؟

اهمیت و ضرورت یادگیری مبحث پایگاه داده

امروزه داده‌ها و اطلاعات جزو سرمایه‌های مهم سازمان‌ها و شرکت‌ها هستند. نگهداری اطلاعات کارکنان یک سازمان، نگهداری حساب‌های بانکی، نگهداری اطلاعات مشتریان، ذخیره و گزارش‌دهی فروش شرکت و صدور کارنامه تحصیلی، نمونه‌هایی از این اطلاعات هستند. با توجه به حجم وسیع اطلاعات در زمینه‌های مختلف و لزوم سازماندهی، طبقه‌بندی و ذخیره‌سازی آنها، داشتن یک سیستم قدرتمند اطلاعاتی برای مدیریت داده‌ها یک نیاز ضروری است. زیربنای سیستم‌های اطلاعاتی، پایگاه داده‌ها است.

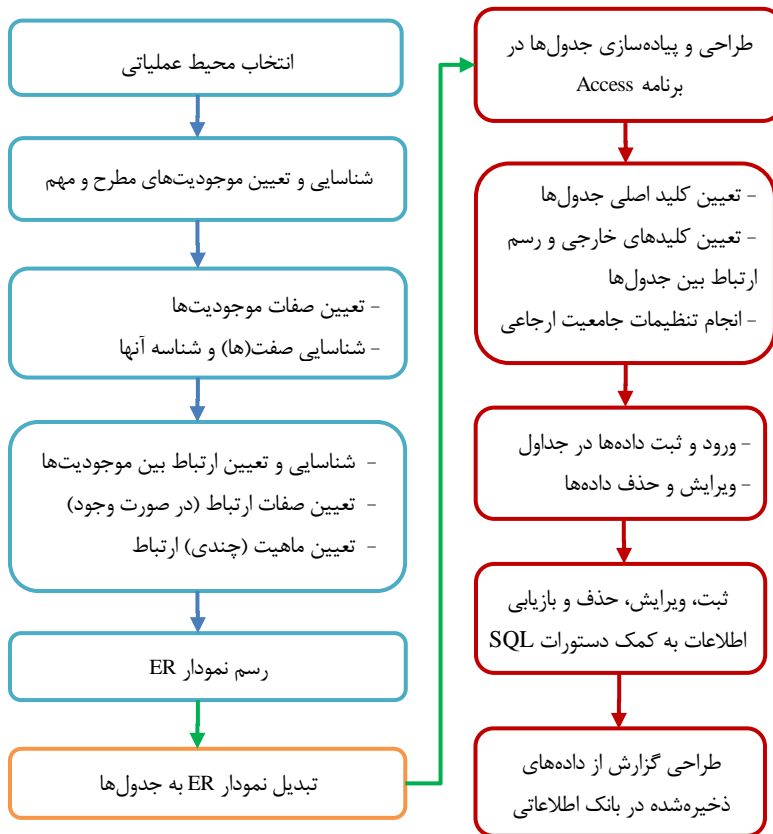
از طرفی، یادگیری این مبحث برای برخی از بودمان‌های دیگر در کتاب‌های درسی مانند موارد زیر، به عنوان یک پیش‌نیاز ضروری مطرح است:

- کتاب «توسعه برنامه‌سازی و پایگاه داده» - پودمان ۵
- کتاب «پایاده‌سازی سیستم‌های اطلاعاتی و طراحی وب» - پودمان‌های ۴ و ۵

کاربردهای سیستم پایگاه داده

امروزه پایگاه داده‌ها تقریباً بخشی ضروری از هر سازمان یا شرکتی را تشکیل می‌دهند. در زیر چند نمونه از کاربردهای بی‌شمار آن را مشاهده می‌کنید:

- امور بانکی: اطلاعات مشتریان، حساب‌ها، وام‌ها و تراکنش‌های مالی
- خطوط هوایی: اطلاعات خطوط هوایی، زمان‌بندی و سیستم رزرو پروازها. خطوط هوایی نمونه‌ای از کاربردهای پایگاه داده توزیع‌شده جغرافیایی در نقاط مختلف دنیا را به نمایش می‌گذارد.
- دانشگاه‌ها، مدارس و مؤسسات آموزشی: اطلاعات دانشجو یا دانش‌آموز، ثبت درس‌ها، برنامه‌ریزی درسی و نمرات
- تراکنش‌های کارت اعتباری: عابر با نک‌ها، خرید از طریق کارت اعتباری و صورت‌حساب‌های ماهانه
- مخابرات: ثبت تماس‌ها، قبوض ماهانه، نگهداری اعتبار سیم‌کارت، ذخیره اطلاعات شبکه‌های مخابراتی
- امور مالی: ذخیره اطلاعات دارایی‌ها، خرید و فروش اسناد مالی مانند سهام و اوراق قرضه
- امور فروش: اطلاعات مشتریان، کالاها، خریدها و فروش‌ها
- تولید و صنعت: مدیریت زنجیره مواد اولیه برای تولید محصولات، انبارداری، سفارش‌ها
- منابع انسانی: اطلاعات کارکنان و پرسنل، حقوق و دستمزد، مالیات



تدریس

تاریخچه ذخیره‌سازی و بازیابی اطلاعات

برای درک بهتر و روشن شدن موضوع درس، مروری بر مفاهیم پایه خواهیم داشت. داده: تعاریف مختلفی در متون علمی برای این مفهوم ارائه شده است که دو مورد آن را اینجا ذکر می‌کنیم.

- هر مجموعه‌ای از داشته‌ها (دانستنی‌ها).
- نمایش ذخیره‌شده اشیا فیزیکی، چیزهای مجرد (Abstract)، داشته‌ها، رویدادها یا چیزهای قابل مشاهده که در تصمیم‌سازی به کار می‌آیند.

اطلاع (اطلاعات): به طور خیلی ساده، اطلاعات، داده پردازش شده، تفسیر شده و دارای معنی و کاربرد مشخص است. اطلاعات بر خلاف داده خام، داده سازمان یافته‌ای است که شناختی را منتقل می‌کند و برای تصمیم‌گیری نیز به کار می‌رود.

سیستم ذخیره و بازیابی اطلاعات: سیستم ذخیره و بازیابی، سیستمی است که به کاربر امکان می‌دهد تا داده‌ها و اطلاعات خود را ذخیره، بازیابی و پردازش کند.

پایگاه داده‌ها: پایگاه داده‌ها (بانک اطلاعاتی) به مجموعه‌ای از اطلاعات با ساختار منظم و سازمان یافته گفته می‌شود. پایگاه داده‌ها معمولاً در قالبی که برای دستگاه‌ها و رایانه‌ها قابل خواندن و دسترسی باشد، ذخیره می‌شوند. پایگاه داده‌ها یکی از انواع سیستم‌های «ذخیره و بازیابی اطلاعات» است. آنچه ذخیره‌سازی داده‌ها در پایگاه داده‌ها را مؤثر می‌سازد وجود یک ساختار مفهومی برای ذخیره‌سازی و روابط بین داده‌ها است. پایگاه داده می‌تواند به سؤالات کاربر در مورد داده‌های سیستم، پاسخ دهد. بخش‌های بازیابی شده در هر پرسش به اطلاعاتی تبدیل می‌شود که برای اتخاذ یک تصمیم کاربرد دارد. در این بحث اصطلاح «پایگاه داده» و «بانک اطلاعاتی» با معانی یکسان به کار می‌روند.

مشکلات متداول در فرآیند یاددهی - یادگیری

نمونه سؤالاتی که ممکن است برای هنرجویان پیش بیاید:

- منظور از افزونگی (Redundancy) چیست؟

افزونگی یعنی وجود داده‌ها و اطلاعات تکراری ذخیره شده در سیستم یا به عبارتی دیگر، در اختیار داشتن نسخه‌ی تکراری دیگری از داده‌ها در محیط ذخیره‌سازی.

افزونگی بر دو نوع است:

الف) طبیعی: که ناشی از ماهیت خود داده‌های محیط است. مثلاً در لیست مشخصات هنرجویان ممکن است نام، نام خانوادگی، نام رشته یا پایه تحصیلی تکرار شود. به جدول ۱ نگاه کنید.

| شماره | نام | نام خانوادگی | نام پدر | پایه تحصیلی | رشته تحصیلی |
|-------|------|--------------|---------|-------------|-------------------------|
| ۱ | علی | احمدی | حسین | دهم | شبکه و نرم‌افزار رایانه |
| ۲ | محمد | کریمی | امیر | دهم | شبکه و نرم‌افزار رایانه |
| ۳ | علی | حسینی | رضا | یازدهم | شبکه و نرم‌افزار رایانه |
| ۴ | محمد | کریمی | حسن | یازدهم | شبکه و نرم‌افزار رایانه |

ب) تکنیکی: جهت بهبود کارایی سیستم از دید کاربران، توسط طراح سیستم ایجاد می‌شود مثل فهرست مطالب یک کتاب.

قابل توجه این است که افزونگی در کل قابل حذف کامل نیست؛ یعنی نمی‌توان سیستمی پیدا کرد که اصلاً افزونگی نداشته باشد، اما با طراحی درست می‌توان آن را به حداقل رساند (نرمال سازی).

– استفاده از سیستم پایگاه داده چه مزایا و معایبی دارد؟

الف) مزایای استفاده از پایگاه داده: محیط یکپارچه و مجتمع، مدیریت متمرکز، عدم تداخل و حفظ سازگاری داده، اشتراکی بودن داده‌ها بین کاربران مختلف، حداقل افزونگی و تکرار اطلاعات، اعمال امنیت آسانتر و ...

ب) معایب پایگاه داده: به علت متمرکز بودن و با خراب شدن یک رایانه ممکن است داده‌ها به خطر بیفتند که با پشتیبان‌گیری می‌توان آن را جبران کرد. برنامه‌نویسی و پیاده‌سازی همه مفاهیم آن پیچیده است. شاید سخت‌افزار اضافه نیاز باشد. آموزش طراحان و کاربران ممکن است هزینه‌بر باشد و ...

– مگر اطلاعات پایگاه داده که بر روی دیسک ذخیره شود در قالب فایل ذخیره نمی‌شود؟ پس پایگاه داده چه تفاوتی با فایل دارد؟

پاسخ این است که بله، درست است که هر چیزی که می‌خواهد ذخیره شود، در نهایت به صورت فایل ذخیره می‌شود، اما تفاوت اینجاست که در روش فایلینگ (Filing) ما مستقیماً با فایل کار می‌کنیم و فایل ما دقیقاً همان ساختاری را دارد که ما در برنامه تعریف می‌کنیم. اما در هنگام استفاده از سیستم مدیریت پایگاه داده، ما از طریق یک واسط (DBMS)، ساختار داده‌های موردنظر خود را تعریف می‌کنیم. پس از آن، این سیستم مدیریت پایگاه داده است که داده‌های ما را پس از نگاشت‌های مختلف در فایل‌هایی ذخیره می‌کند که استفاده از محتوای آن فایل‌ها فقط از طریق خود DBMS امکان‌پذیر است.

– منظور از مدل‌سازی معنایی داده‌ها چیست؟

منظور این است که یک مدل کلی از داده‌های محیط با استفاده از مفاهیم انتزاعی، فارغ از جزئیات پیاده‌سازی و بر اساس معنایی که کاربر برای داده‌ها قائل است، ارائه دهیم. این مدل‌سازی به کمک نمودار ER انجام می‌شود.

شیوه و الگوی پیشنهادی

کار تیمی در شرکت‌ها و سازمان‌های تولیدکننده نرم‌افزار، یک امر ضروری، اساسی و بدیهی است. هنرجویان باید یاد بگیرند که برای اتمام یک پروژه نرم‌افزاری، به افراد مختلف با تخصص، علایق و استعداد‌های متفاوت نیاز است. بر این اساس، برای جمع‌آوری اطلاعات، طراحی و پیاده‌سازی پروژه‌های بانک اطلاعاتی و به منظور استفاده از محیط فعال، هنرجویان کلاس را به گروه‌هایی تقسیم کنید. افراد هر گروه باید با همفکری و همیاری، کارها را به انجام برسانند. اگر هر گروه، بتواند کار گروهی دیگر را نیز داوری و ارزیابی کند، برای تقویت روحیه انتقادپذیری و استفاده از نظرات دیگران بسیار ایده‌آل خواهد بود.

اکثر کار گروه‌ها به‌ویژه در واحد یادگیری ۱ روی کاغذ انجام می‌شود و چون تولید نرم‌افزار یک کار فرآیندی و مرحله‌ای است، روی این قسمت تأکید بیشتری وجود دارد. لازم به ذکر است که انتظار نمی‌رود هنرجویان یک طراح حرفه‌ای پایگاه داده شوند یا وظایف یک مهندس نرم‌افزار را انجام دهند، بلکه هدف آشنایی اصولی آنها با حیطه و روش کار در حرفه متناسب با رشته خود به‌ویژه کار تیمی است.

هنرآموزان می‌توانند خود را یکی از اعضای هر گروه دانسته و با همه گروه‌ها مشارکت فعال داشته و هم‌فکری کنند. توصیه می‌شود ارزشیابی هنرجویان بر اساس تولید ایده و اظهار نظر باشد و نتیجه کار گروه مد نظر قرار گیرد.

پاسخ به فعالیت‌ها

فعالیت گروهی
ص ۵

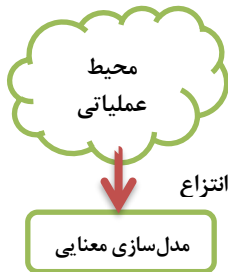
مشکلات به‌وجود آمده در خصوص جمع‌آوری اطلاعات هنرجویان ناشی از چیست؟
پاسخ: احتمال ایجاد مغایرت در داده‌ها وجود دارد. یعنی ممکن است اطلاعاتی از دانش‌آموز در دفتر معاون آموزشی تغییر پیدا کند، ولی این تغییر در دفتر معاون اجرایی اعمال نشود. لذا پس از مدتی با حجم انبوهی از مغایرت‌ها مواجه خواهیم شد. و همچنین این کار موجب افزونگی و ایجاد داده‌های تکراری می‌شود. به‌طور خلاصه مشکلات تلف شدن فضای ذخیره‌سازی، سخت بودن اعمال تغییرات، خطر ناسازگاری داده‌ها و عدم دقت را باعث خواهد شد.

کنجکاوی
ص ۶

دفترچه تلفنی را در نظر بگیرید که در آن نام و نام خانوادگی افراد و تلفن آن‌ها پشت سر هم و بدون هیچ قاعده‌ای ثبت شده است. به نظر شما این روش چه مشکلی در پی خواهد داشت؟ دلیل خود را ذکر کرده، راه‌حل موردنظر را ارائه دهید.
پاسخ: با این روش، جستجوی یک مخاطب بسیار دشوار می‌شود.
راه‌حل: نام خانوادگی افراد بر اساس حروف الفبا مرتب شوند.

مفاهیم پایگاه داده

محیط عملیاتی



برای مدل‌سازی، نقطه شروع شناسایی و تعیین محیط عملیاتی است. یعنی می‌خواهیم اطلاعات مربوط به کدام سازمان، شرکت، مؤسسه یا مکان را ذخیره کنیم؟ سپس به‌صورت کلی و صرف نظر از جزئیات پیاده‌سازی و به‌صورت انتزاعی مدل‌سازی انجام می‌شود.

مثال از چند محیط عملیاتی:

- بیمارستان
- دانشگاه
- کتابخانه
- داروخانه
- فروشگاه

موجودیت (Entity)

مفهوم کلی شیء، پدیده و به طور کلی هر آنچه که می‌خواهیم در مورد آن اطلاعاتی در پایگاه داده داشته باشیم و با استفاده از آن اطلاعات، شناخت خود را افزایش دهیم. اولین قدم عملی در مدل‌سازی معنایی، تشخیص درست موجودیت‌هاست. برای مثال فرض کنید محیط عملیاتی هنرستان باشد، موجودیت‌های آن عبارتند از:

- هنرجویان
- هنرآموزان
- درس‌ها
- امکانات آموزشی
- کلاس‌ها
- کارکنان اداری

چه ضابطه‌هایی (معیارهایی) برای تشخیص و لحاظ کردن موجودیت در یک محیط عملیاتی وجود دارند؟

- ۱) بتوان برای آن موجودیت، نمونه‌های (Instance) واقعی و متمایز پیدا کرد. نمونه‌ها، مصداق‌های آن موجودیت هستند و در واقع نمونه‌ها هستند که وجود خارجی دارند. مثلاً: هنرجو در محیط عملیاتی هنرستان دارای نمونه‌هایی مثل «هنرجویان حاضر در کلاس» است.
- ۲) هر موجودیت باید دارای ویژگی‌ها و صفاتی باشد که آن را توصیف می‌کنند و کاربر به مجموعه‌ای از اطلاعات در مورد آن نیاز دارد. مثلاً: صفات هنرجو عبارتند از: شماره، نام، نام خانوادگی، رشته تحصیلی و ...
- ۳) موجودیت باید نقشی داشته باشد که یا کاری را انجام دهد یا کاری بر روی آن انجام شود یعنی با موجودیت‌های دیگر حداقل یک ارتباط داشته باشد.
- ۴) مثلاً در محیط آموزشی هنرستان:

موجودیت درس ← توسط هنرجو اخذ می‌شود.
موجودیت هنرجو ← درس را اخذ می‌کند.

– هر سه ضابطه فوق باید در مورد هر موجودیت وجود داشته باشند. علاوه بر ضوابط فوق ممکن است ضابطه‌های دیگری هم وجود داشته باشند که با توجه به نیاز کاربر انتخاب یا حذف می‌شوند. مثلاً دیوار در یک محیط آموزشی ممکن است هر سه ضابطه فوق یعنی نمونه، ویژگی و کاربرد را داشته باشد اما جزء نیازهای اطلاعاتی کاربران نیست.

– یک مفهوم در یک محیط عملیاتی ممکن است موجودیت و در محیط دیگر تنها صفتی برای یک موجودیت باشد. مثلاً رنگ در یک کارخانه‌ی رنگ‌سازی موجودیت مطرح، اما در کارخانه‌ی ماشین‌سازی صفتی است برای ماشین.

– مفهوم محیط عملیاتی از مفهوم موجودیت جداست و نمی‌توان خود محیط عملیاتی را موجودیت در نظر گرفت. به بیان ساده، چیزی موجودیت تلقی می‌شود که بخواهیم لیستی

از آن را در سیستم نگه داریم. برای مثال، هنرستان را موجودیت به حساب نمی‌آوریم؛ چون سیستم قرار است اطلاعات هنرجویان یک هنرستان را مدیریت کند. مگر اینکه بخواهیم اطلاعات آموزشی همه هنرجویان شهرستان، استان یا کشور را در یک سیستم داشته باشیم.

صفت (Attribute)

حالت یا وضع موجودیت را توصیف می‌کند و ویژگی‌های مهم یا خاص آن را با توجه به محیط عملیاتی مشخص می‌کند و بر اساس آنها، نیازهای اطلاعاتی مورد نظر کاربران تأمین می‌شوند. انتخاب صفات برای موجودیت‌ها بر اساس نیاز کاربر به اطلاعات آنها، ملاحظات طراحی سیستم و ... است.

صفت شناسه (Identity)

صفتی که مقادیر آن در تمام نمونه‌های یک موجودیت تکراری نبوده و مقدار منحصر به فرد دارد. مانند صفت شماره دانش‌آموزی برای هنرجو یا کد ملی افراد. گاهی اوقات، یک موجودیت صفت با مقادیر غیر تکراری ندارد تا بتوان آن را به عنوان کلید در نظر گرفت. یک راه برای حل این مشکل استفاده از کلیدهای ترکیبی از چند صفت با هم است. مثلاً فرض کنید ترکیب سه صفت نام، نام خانوادگی و نام پدر بین هنرجویان، امکان تکرار ندارد. به جدول ۱ نگاه کنید. راه حل دیگر استفاده از کلید مجازی است. یعنی صفتی جدید مانند شماره سریال به موجودیت اضافه کنیم. مجازی بودن یعنی صفت مورد نظر، ویژگی ذاتی موجودیت نیست، بلکه بنا بر ملاحظات طراحی در سیستم، به آن نسبت داده شده است. مثلاً یک فرد به محض ثبت نام در مدرسه یک شماره دانش‌آموزی می‌گیرد و زمانی که فارغ‌التحصیل می‌شود دیگر این شماره را از دست می‌دهد. نقش شناسه ← یک نمونه موجودیت را از بقیه جدا و شناسایی می‌کند. مثلاً فقط یک نفر از هنرجویان کلاس دارای کد ملی مشخص شده‌ای خواهد بود.

پاسخ به فعالیت‌ها

کنجکاوی
ص ۷

آیا در محیطی مانند هنرستان، می‌توان میز و نیمکت‌ها را نیز به عنوان موجودیت در نظر گرفت؟ برای پاسخ خود دلیل بیاورید. معیار انتخاب موجودیت چیست؟ پاسخ: خیر. با توجه به نکته ۱، میز و نیمکت‌ها جزء نیازهای اطلاعاتی کاربران نیست. معیارهای انتخاب موجودیت یعنی نمونه، ویژگی و کاربرد در مطالب قبل بحث شد.

فعالیت کارگاهی
ص ۸

- با در نظر گرفتن محیط عملیاتی تاکسی تلفنی، جدول را تکمیل کنید.
پاسخ: راننده، ماشین (تاکسی)، مشتریان، پرسنل دفتر تاکسی تلفنی
- موجودیت‌های محیط عملیاتی کتابخانه را نام ببرید.
پاسخ: کتاب، اعضا، پرسنل اداری، نشریات دوره‌ای (روزنامه، مجله، فصل‌نامه) و ...

- آیا می‌توان تمامی صفتهای مربوط به یک موجودیت را در محیط عملیاتی مانند هنرستان در نظر گرفت؟ به نظر شما معیار انتخاب صفت برای یک موجودیت چیست؟
- برای پاسخ خود دلیل بیاورید.

پاسخ: خیر. زیرا تمام صفات برای ما حائز اهمیت نیستند. تنها باید صفاتی را مدنظر قرار داد که در گزارش‌های نهایی که قرار است از پایگاه داده گرفته شود نقش مهم و کاربردی داشته باشند. مثلاً اطلاعات شناسنامه‌ای دانش‌آموز مهم است اما رنگ چشم اهمیت و کاربردی ندارد. مدنظر قرار دادن صفات غیر کاربردی فقط موجب اضافه‌کاری و نگهداری داده‌های غیر ضروری (افزونگی) می‌شود.

ارتباط (Relationship)

ارتباط بیانگر نقش، تعامل و وابستگی بین دو یا بیش از دو نوع موجودیت است.

- ارتباط نیز مانند موجودیت می‌تواند دارای صفت باشد که آن را توصیف کند.
- یک ارتباط می‌تواند بین یک موجودیت و خودش برقرار باشد. مانند ارتباط «پیش‌نیازی» بین درس‌ها یا ارتباط «هم‌کلاسی» بین هنرجویان کلاس
- یک ارتباط می‌تواند بین بیش از دو موجودیت وجود داشته باشد. مانند ارتباط «تدریس» بین موجودیت‌های هنرجو، درس و هنرآموز
- بین دو موجودیت می‌توان چندین ارتباط را در نظر گرفت. مانند ارتباط فروش و سفارش بین کالا و مشتری در محیط عملیاتی فروشگاه

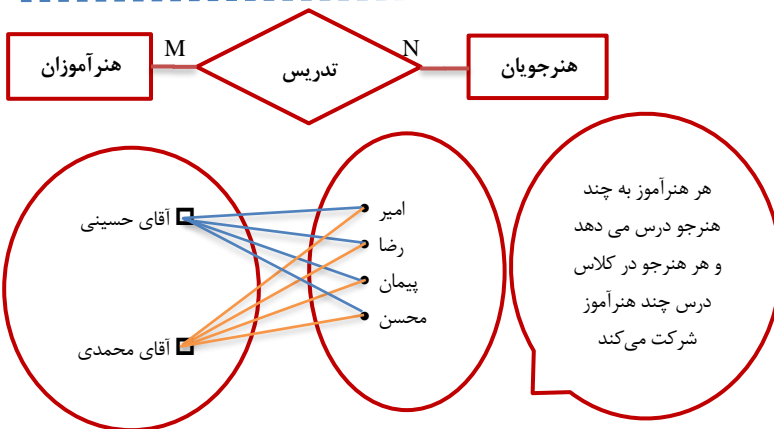
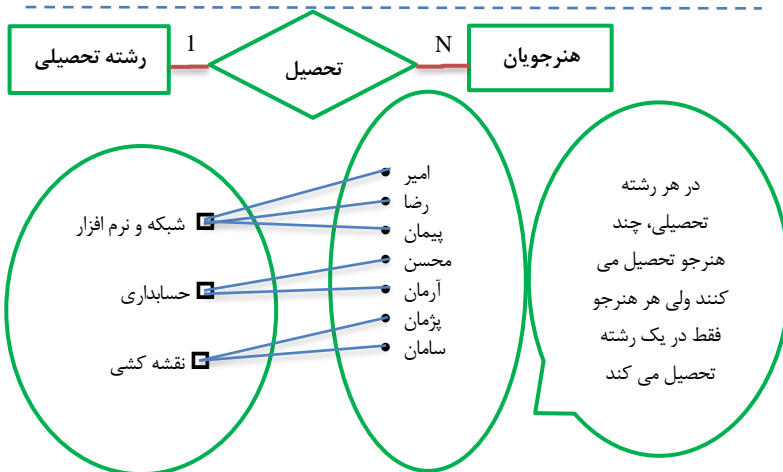
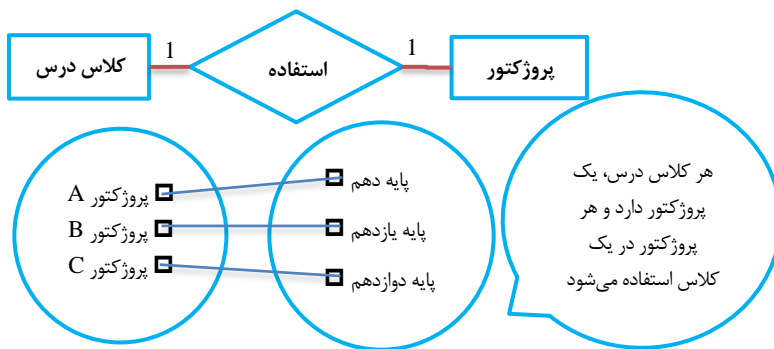
نمودار ER (Entity-Relationship Diagram)

موجودیت‌ها، صفات و ارتباط بین آنها به کمک یک نمودار شامل تعدادی شکل گرافیکی نمایش داده می‌شوند که به آن ER می‌گویند. رسم این نمودار و اجزاء آن، بر اساس فرضیات و قواعد محیط عملیاتی مورد نظر انجام می‌شود. مثلاً اینکه در محیط عملیاتی تاکسی تلفنی، موجودیت مسافر از سرویس استفاده می‌کند که توسط راننده‌ای ارائه می‌شود، بیانگر این است که بین موجودیت‌های مسافر و راننده ارتباط سرویس برقرار است. در ادامه بحث چند نمونه نمودار ER ساده شده از محیط‌های عملیاتی مختلف را مشاهده خواهید کرد.

- آیا همه نمونه‌های یک موجودیت در ارتباط مشارکت دارند؟
- تحقیق کنید که منظور از مشارکت الزامی و غیرالزامی یک موجودیت در ارتباط چیست؟

چندی (ماهیت) ارتباط:

عبارت است از چگونگی تناظر بین دو مجموعه از نمونه‌های دو نوع موجودیت شرکت‌کننده در ارتباط. منظور این است که هر نمونه از یک موجودیت، با چه تعداد از نمونه‌های موجودیت دیگر در یک ارتباط مشارکت دارد. مثلاً در ارتباط تدریس، چند هنرجو در کلاس درس یک هنرآموز حضور دارند. ماهیت ارتباط می‌تواند یک به یک (1:1)، یک به چند (N:1) یا چند به چند (M:N) باشد. ارتباط چند به چند را به صورت (N:N) هم نمایش می‌دهند. برای روشن شدن این موضوع شکل ۲ را ببینید.



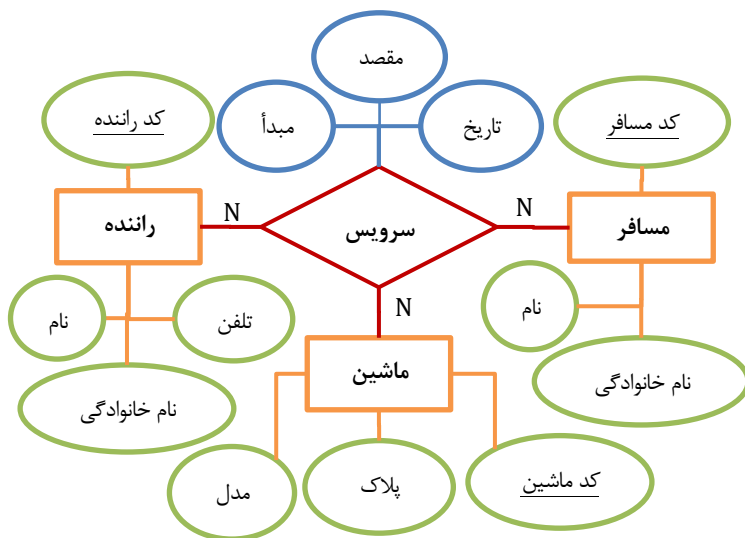
چگونه مدل ER یک محیط عملیاتی را رسم کنیم؟

- ۱) مطالعه، تحلیل و شناخت محیط عملیاتی مورد نظر
- ۲) شناسایی انواع موجودیت‌های مطرح با توجه به اینکه می‌خواهیم اطلاعاتی در مورد چه چیزهایی در سیستم ذخیره کنیم.
- ۳) تعیین ویژگی‌ها و صفات هر نوع موجودیت و همچنین صفت شناسه
- ۴) شناسایی ارتباط بین انواع موجودیت‌ها با توجه به تعامل بین موجودیت‌ها و تعیین چندی ارتباط (تناظر نمونه‌های موجودیت‌های مشارکت‌کننده در ارتباط با هم)
- ۵) بررسی و آزمایش: با طرح پرسش‌ها مدل‌سازی انجام شده تا اطمینان حاصل شود که مدل‌سازی پاسخگوی نیاز کاربران است. مثلاً آیا مدل‌سازی کتابخانه به این سوال می‌تواند پاسخ دهد که «چه کسانی فلان کتاب را به امانت برده‌اند»؟

شیوه و الگوی پیشنهادی

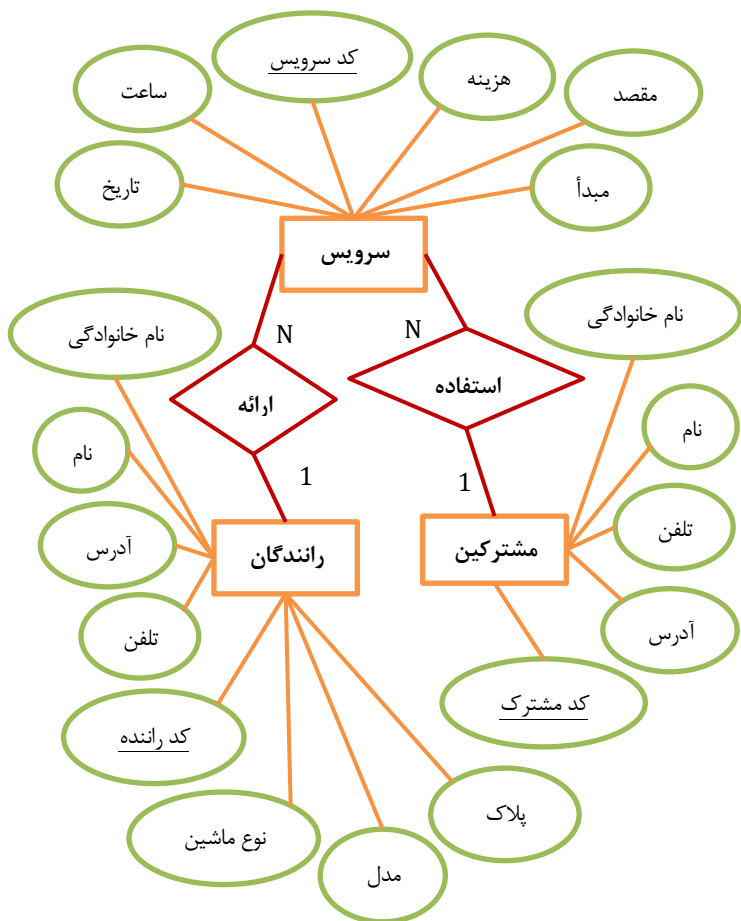
در مرحله مدل‌سازی معنایی (ER)، توصیه می‌شود برای محیط عملیاتی مدرسه یا کتابخانه، یکی از کارکنان مسئول مثلاً معاون فنی با هماهنگی قبلی، در کلاس حضور یابد و در مورد نیازمندی‌های داده‌ای و گزارش‌های مورد انتظار، به عنوان یک استفاده‌کننده از سیستم مدرسه، برای هنرجویان صحبت کرده و به پرسش‌های آنها پاسخ دهد. برخی از پرسش‌های رایج عبارتند از:

- در مورد چه چیزهایی می‌خواهیم اطلاعاتی داشته باشیم؟ (شناسایی موجودیت‌ها)
 - در هر مورد چه نوع اطلاعاتی مورد نیاز است؟ (شناسایی صفات)
 - تعامل موجودیت‌ها با هم چگونه است؟ هر موجودیت در محیط عملیاتی مورد نظر چه نقشی دارد؟ (ارتباط بین موجودیت‌ها)
- مثلاً در محیط عملیاتی آژانس تاکسی تلفنی، موجودیت‌های مطرح، عبارتند از: مسافر (مشترک)، راننده، ماشین سرویس‌دهی. مسافر ممکن است از سرویس چند راننده در زمان‌های مختلف استفاده کند و راننده هم به چندین مسافر سرویس ارائه دهد، این قاعده محیط، ماهیت ارتباط چند به چند را بین راننده، مسافر و ماشین به وجود می‌آورد (شکل ۳).

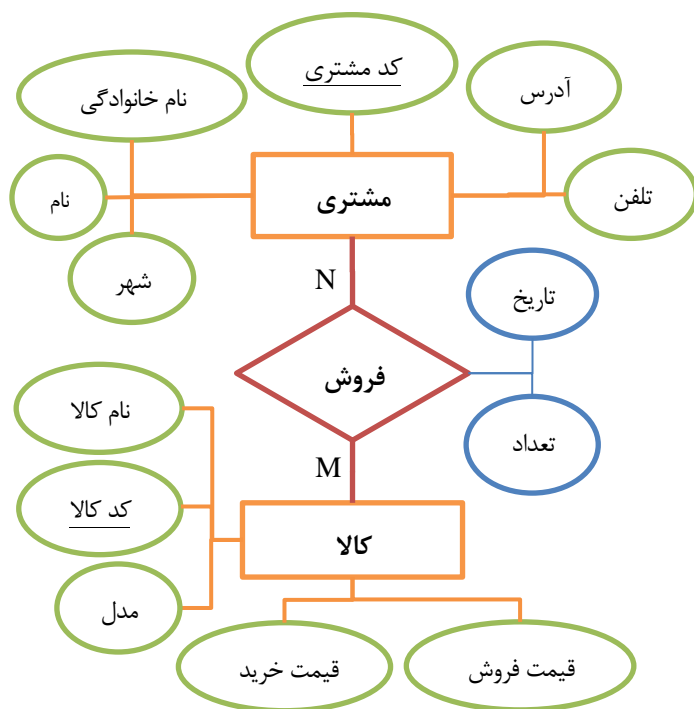


نکته قابل توجه این است که برای یک محیط عملیاتی خاص، بسته به معیارهای ذهنی طراحان و برخی ملاحظات، ممکن است چندین نمودار ER طراحی و رسم شود که همگی هم تا حدی درست باشند. اما یک طراحی ممکن است از دیگر طراحی‌ها جامع‌تر باشد. طراحی دیگری از محیط عملیاتی آژانس تاکسی تلفنی، به صورت شکل ۴ است که در آن مشخصات ماشین به عنوان صفات راننده منظور شده‌اند و سرویس هم به عنوان موجودیت تصور شده است.

توجه شود که اگر «سرویس» یک موجودیت فرض شود، صفت کد مشترک یا کد راننده نمی‌تواند صفت موجودیت «سرویس» باشد. چون مفهوم «کلید خارجی» در مدل‌سازی معنایی داده‌ها (نمودار ER) وجود ندارد، بلکه در مدل‌سازی منطقی (طراحی جدول‌ها) برای نمایش ارتباط بین موجودیت‌ها به کار می‌رود. برای اطلاعات بیشتر به بحث «تبدیل و نگاهت مدل‌سازی معنایی به جدول‌ها» و مبحث «کلید خارجی» در صفحات آتی مراجعه شود.



مثال محیط عملیاتی فروشگاه (ساده شده): فرض کنید در یک فروشگاه، کالاهایی به فروش می‌رسد و مشتریانی از آن فروشگاه خرید می‌کنند. هر مشتری ممکن است چند نوع کالا را بخرد و از هر نوع کالا به تعداد یا مقدار مورد نظر به چندین مشتری فروخته شود.



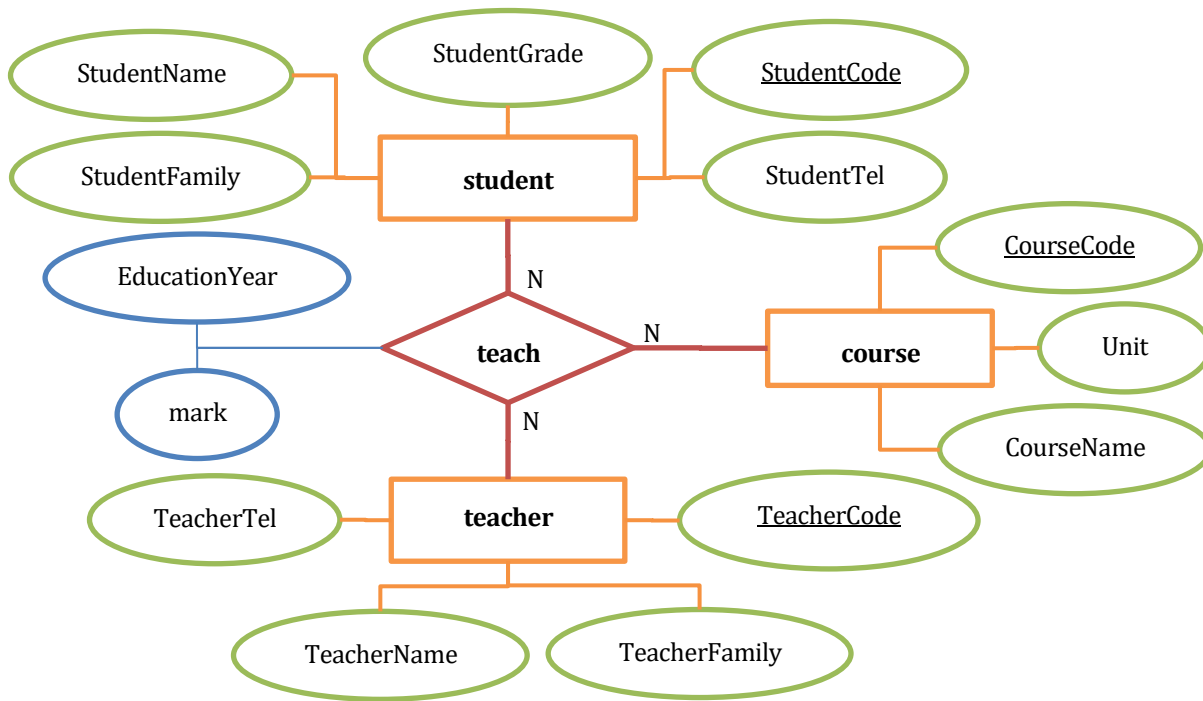
فعالیت تکمیلی

- مدل ER فروشگاه را با لحاظ کردن فاکتور، سفارش مشتری، خرید اولیه فروشگاه از تولیدکنندگان و ... بازنویسی یا تکمیل کنید.

- یک شرکت بیمه‌ی خودرو قصد دارد اطلاعات مربوط به بیمه‌شدگان و نوع خودرو آنها را نگهداری کند. بیمه‌شده می‌تواند حقیقی یا حقوقی باشد. خودرو نیز انواع مختلفی مانند سواری، باری و موتور دارد. مدل‌سازی ER آن را انجام دهید.

طراحی بانک اطلاعاتی سیستم مدرسه (ساده‌شده)

در سیستم مدرسه، موجودیت‌های هنرجو، هنرآموز و درس را در نظر می‌گیریم. هر هنرآموز به تعدادی هنرجو درس می‌دهد و هر هنرجو در کلاس چند درس و هنرآموز شرکت می‌کند. صفت نمره (mark) و سال تحصیلی (EducationYear) را برای ارتباط تدریس در نظر گرفته‌ایم. نمودار ER آن مطابق شکل ۶ طراحی شده است. دقت کنید که نمره صفت درس نیست، نمره پس از اخذ درس معنی پیدا می‌کند.



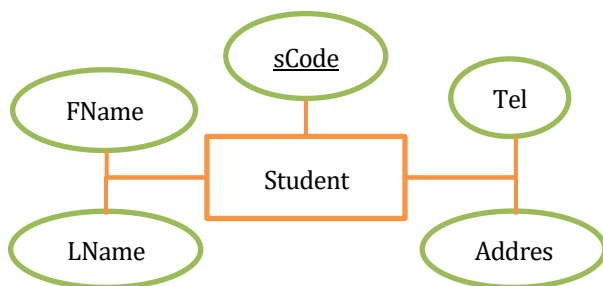
تبدیل و نگاشت مدل‌سازی معنایی (نمودار ER) به جدول‌ها

پس از مدل‌سازی معنایی، حال باید به طراحی منطقی پایگاه داده بپردازیم. یعنی موجودیت‌ها و ارتباط‌ها را به حالت نمایش جدولی تبدیل کنیم. در این جدول‌ها، صفات موجودیت‌ها، ستون‌ها را تشکیل می‌دهند و داده‌ها و اطلاعات واقعی که پس از پیاده‌سازی وارد می‌شوند، سطرها را می‌سازند.

قرارداد: واژه Code را برای صفات شماره یا کد در نظر می‌گیریم و شناسه با زیرخط‌دار کردن مشخص می‌شود. کافی است فقط اطلاعات نام جدول و نام ستون‌ها نوشته شود به شکل زیر:

(... , ستون سوم , ستون دوم , ستون شناسه) نام جدول

قاعده (۱) موجودیت‌ها: هر موجودیت مستقل در یک جدول مدل می‌شود و صفات آن ستون‌های جدول را تشکیل می‌دهند. به نمونه شکل ۷ دقت کنید:

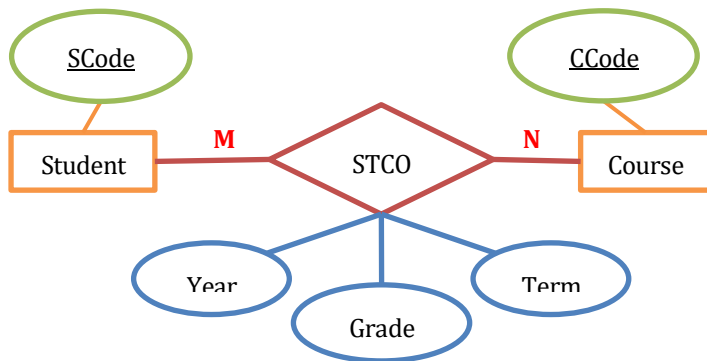


Student (SCode , FName, Lname, Address, Tel)

قاعده (۲) تبدیل ارتباط‌های بین موجودیت‌ها به جدول:

الف) چند به چند (M:N): ارتباط با یک جدول مجزا مدل می‌شود. ستون‌های این جدول جدید عبارتند از:

- صفات خود ارتباط (در صورت وجود)
 - شناسه موجودیت اول و دوم برای برقراری ارتباط
- کلید اصلی جدول جدید، ممکن است از ترکیب شناسه‌های موجودیت‌های مرتبط حاصل شود یا طراح خود کلید جدیدی مثل شماره سریال برای آن تعریف کند.



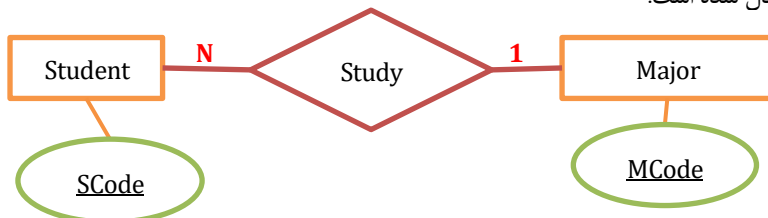
در شکل ۸ طبق قاعده ۱ برای موجودیت‌های مستقل Student, Course دو جدول مجزا طراحی می‌شود. دقت شود برای اجتناب از شلوغ شدن نمودار، از رسم بقیه صفات موجودیت‌های Student, Course خودداری شده است (علامت ...).

Student (SCode, ...) Course (CCode, ...)

طبق قاعده ۲-الف، برای ارتباط چند به چند (M:N)، یک جدول مجزا طراحی می‌شود. شناسه دو موجودیت که برای برقراری ارتباط به این جدول جدید آمده‌اند، با رنگ قرمز مشخص شده است و شناسه ترکیبی جدول سوم را تشکیل می‌دهند.

STCO (SCode, CCode, year, term, grade)

ب) یک به چند (1:N): جدول مجزا نیاز نیست. کافی است شناسه موجودیت طرف یک را در جدول متناظر با طرف N بیاوریم. در مثال شکل ۹ ارتباط هنرجو و رشته تحصیلی مدل شده است:



طبق قاعده ۱ دو جدول مجزا برای موجودیت‌های هنرجو و رشته تحصیلی، ساخته می‌شود و طبق قاعده ۲-ب شناسه MCode مربوط به موجودیت رشته (Major) در جدول Student برای ایجاد ارتباط به کار می‌رود:

Student (SCode, ..., MCode) Major (MCode, ...)

ج) یک به یک (1:1): نیازی به جدول مجزا نیست مشابه حالت فوق (ب) عمل می‌کنیم، بسته به نظر طراح و ... شناسه یکی از طرفین را در طرف دیگر می‌آوریم.

مثال طراحی جدول‌های بانک اطلاعاتی سیستم مدرسه:

بر اساس نمودار ER رسم‌شده قبلی (شکل ۶) و قواعد تبدیل، می‌توان مشخصات جدول‌ها را طبق جدول ۲ طراحی کرد. در جدول حاصل از ارتباط تدریس، برای سادگی بیشتر، صفت شناسه مجازی (شماره ردیف: RowID) به جای شناسه ترکیبی صفات کد هنرجو، کد درس و کد هنرآموز، به عنوان کلید اصلی استفاده شده است.

| کلیدها | عنوان ستون‌ها | عنوان جدول |
|---|---|--------------------|
| کلید اصلی: StudentCode | <u>StudentCode</u> , StudentName, StudentFamily, StudentGrade, StudentTel | هنرجو student |
| کلید اصلی: TeacherCode | <u>TeacherCode</u> , TeacherName, TeacherFamily, TeacherTel | هنرآموز teacher |
| کلید اصلی: CourseCode | <u>CourseCode</u> , CourseName, Unit | درس course |
| کلید اصلی: RowID کلیدهای خارجی: StudentCode, TeacherCode, CourseCode | <u>RowID</u> , StudentCode, TeacherCode, CourseCode, EducationYear, mark | تدریس teach |

شیوه و الگوی پیشنهادی

با توجه به شیوه پیشنهادی در مبحث ارتباط، می‌توان برای تکمیل و تفهیم مطالب این قسمت، پرسش‌ها و نیازمندی‌های اطلاعاتی زیر را نیز مطرح کرد:

- اطلاعات ورودی از کجا تأمین شده و چگونه وارد سیستم می‌شوند؟ (داده‌های ورودی)
- چه اطلاعات جدیدی هر روز به سیستم وارد و ثبت می‌شوند؟ (درج اطلاعات)
- چه ویرایش و اصلاحاتی ممکن است روی داده‌ها انجام شود؟ (ویرایش و حذف)
- چه اعمال، محاسبات و پردازش‌هایی انجام می‌شود؟ (پردازش داده‌ها)
- گزارش‌های خروجی کدامند و حاوی چه اطلاعاتی هستند؟ (پرس و جوا)

در این بخش بهتر است هنرجویان حتماً دفتری برای رسم نمودار ER به همراه داشته باشند و در آن برای محیط‌های مختلف مدل‌سازی ER انجام دهند. نخست لازم است روی کاغذ، مشخصات جدول‌ها را با توجه به قواعد تبدیل از نمودار ER به دست آورند. سپس در محیط Access اقدام به طراحی آنها کنند و کارگاه ۱ تا ۴ کتاب درسی را انجام دهند.

پاسخ به فعالیت‌ها

کنجکاو
ص ۱۳

صفت شماره کارمندی و نام خانوادگی چه تفاوتی در نمودار دارند؟
پاسخ: صفت شماره کارمندی یکتاست و تکراری نیست یعنی برای هر کارمند منحصر به فرد است در حالی که نام خانوادگی چنین ویژگی ندارد و ممکن است چند کارمند نام خانوادگی یکسانی داشته باشند.

کنجکاو
ص ۱۴

مثال، ارائه دهید که موجودیت با خودش در ارتباط باشد.
پاسخ: یک ارتباط می‌تواند بین یک موجودیت و خودش برقرار باشد. مانند ارتباط «پیش‌نیازی» بین درس‌ها یا ارتباط «هم‌کلاسی» بین هنرجویان کلاس

پایگاه داده

برنامه رایانه‌ای را که برای مدیریت، پرسش و پاسخ بین کاربران و پایگاه داده استفاده می‌شود، سیستم مدیریت پایگاه داده یا به اختصار (Database Management System) می‌نامیم. این نرم‌افزار واسط بین محیط فیزیکی ذخیره و بازیابی اطلاعات و محیط منطقی برنامه‌سازی در سیستم پایگاه داده است که هرگونه ارتباط بین کاربر و داده را کاملاً کنترل و مدیریت می‌کند و به کاربر امکان می‌دهد تا پایگاه داده‌های خود را تعریف کند، در پایگاه داده‌های خود عملیات ذخیره، بازیابی، پردازش را انجام دهد و آنها را مدیریت کند.

مقایسه سیستم پایگاه داده و سیستم فایل:

بخشی از یک مؤسسه پس‌انداز یا بانک را در نظر بگیرید که اطلاعات همه مشتریان و حساب‌های پس‌انداز را نگهداری می‌کند. یکی از روش‌های نگهداری اطلاعات آن، ذخیره در فایل‌های جداگانه است. برای اینکه کاربران با اطلاعات کار کنند، سیستم باید چندین برنامه کاربردی برای کار با فایل‌ها داشته باشد مانند:

- بدهکار (واریز) و بستانکار (برداشت) کردن حساب‌ها
- افزودن رکورد جدید (افتتاح حساب)
- یافتن موجودی یک حساب
- ایجاد صورت حساب ماهانه

برنامه‌های کاربردی جدیدی نیز حسب نیاز ممکن است به سیستم اضافه شوند. مثلاً برای افزودن حساب‌های جاری و چک‌ها، فایل‌های جدیدی ایجاد می‌شوند و در نتیجه برنامه‌های متناسب جدید باید طراحی شوند. پس به مرور زمان سیستم به فایل‌ها و برنامه‌های بیشتری نیاز پیدا می‌کند. این شیوه ثبت اطلاعات قبل از ظهور پایگاه داده مرسوم بوده است. نگهداری اطلاعات سازمان در سیستم پردازش فایل چندین مشکل اصلی را به دنبال دارد:

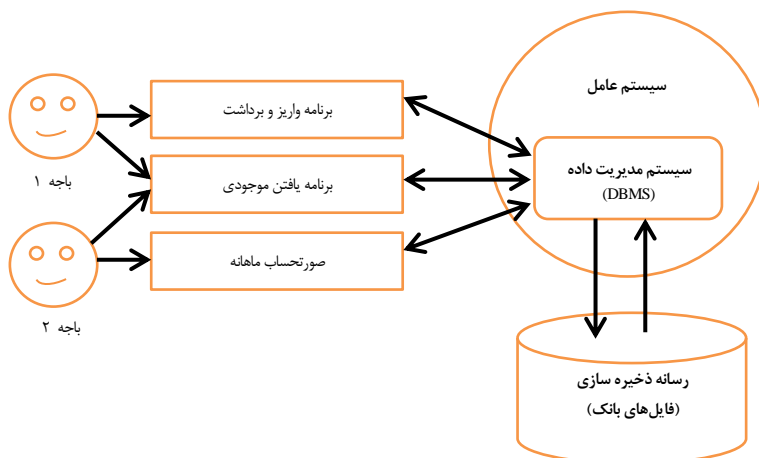
- افزونگی داده‌ها و ناسازگاری: تکرار اطلاعات در چندین فایل، افزونگی را به دنبال دارد. مثلاً مشخصات مشتری ممکن است هم در فایل حساب پس‌انداز و هم حساب

جاری موجود باشد. همچنین متفاوت بودن قالب داده‌ها در فایل‌های هر بخش از سیستم، ناسازگاری بین داده‌ها را ایجاد می‌کند.

- مشکل در دسترسی به داده‌ها: مثلاً یافتن لیست همه مشتریانی که در مکانی یکسان سکونت دارند، به دلیل تفاوت در قالب و ساختار فایل‌ها و برنامه‌های وابسته به آنها، بسیار مشکل است و شاید نیاز به طراحی برنامه جدید داشته باشد.
- مشکلات جامعیت: اعمال محدودیتهای جدید در سیستم مانند حفظ حداقل موجودی حساب‌ها در برداشت‌ها مشکل است. چون از قبل ممکن است مقدار آن تعیین نشده باشد.

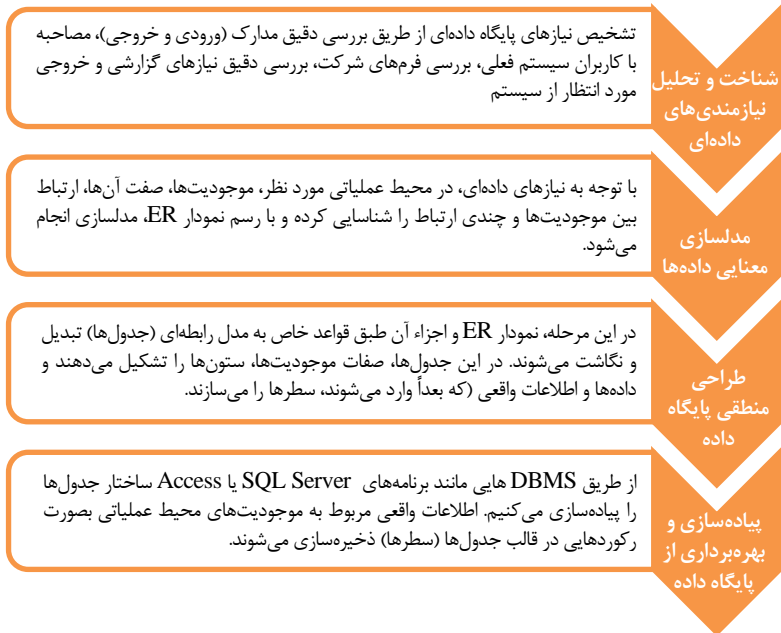
- مشکلات امنیتی: همه کاربران نباید به تمام داده‌ها دسترسی داشته باشند. مثلاً در سیستم بانک مورد نظر ما، پرسنل حقوق و دستمزد، فقط اطلاعات کارکنان را ببینند و نباید به حساب مشتریان دسترسی داشته باشند.

علاوه بر موارد مذکور، مشکلاتی از قبیل کدنویسی زیاد برای پیاده‌سازی بخش‌های مختلف سیستم، مشکل بودن پشتیبان‌گیری و ... هم در سیستم فایل وجود دارد. برای اجتناب از مشکلات فوق، از پایگاه داده‌ها جهت ذخیره و بازیابی داده‌ها استفاده می‌کنیم و عملیات خود را تحت کنترل یک سیستم مدیریتی قدرتمند به نام سیستم مدیریت پایگاه داده (DBMS) انجام می‌دهیم. در این روش نیازهای اطلاعاتی تمامی قسمت‌ها مورد مطالعه قرار می‌گیرد تا بتوان یک سیستم یکپارچه طراحی کرد که پاسخگوی نیاز همه کاربران باشد. محیط ذخیره‌سازی واحدی به صورت مجتمع و اشتراکی و تحت کنترل متمرکز وجود دارد که کاربران بر اساس نیاز خود، به بخش‌های مختلف به واسطه DBMS دسترسی دارند. هر کاربر ممکن است به چند بخش (برنامه) دسترسی داشته و هر برنامه توسط چند کاربر مورد استفاده قرار گیرد. شکل ۱۰ نمای کلی از روش پایگاه داده‌ای برای یک سیستم بانکی بسیار ساده را نشان می‌دهد.



مراحل ایجاد و تولید یک سیستم پایگاه داده‌ای برای یک محیط عملیاتی

ایجاد، طراحی و پیاده‌سازی یک پایگاه داده رابطه‌ای برای یک محیط عملیاتی مشخص را می‌توان در شکل ۱۱ خلاصه کرد. مراحل کار باید به ترتیب انجام شوند.



پاسخ به فعالیت‌ها

کنجکاو
ص ۱۶

چند نمونه از سیستم‌های مدیریت پایگاه داده مشهور را نام ببرید.
پاسخ:

Oracle, MySQL, Microsoft SQL Server, DB2, Microsoft Access

برای مقایسه و آگاهی از رتبه‌بندی انواع سیستم‌ها به تارنمای زیر مراجعه کنید:
<https://db-engines.com/en/ranking>

کارگاه ۲ - طراحی جدول جدید

کلید اصلی (Primary Key): یکی از صفات (شناسه) موجودیت به انتخاب طراح است که مقادیر آن به ازای تمام نمونه‌های آن موجودیت، دارای خاصیت یکتایی مقدار (غیرتکراری) باشد. نقش کلید اصلی این است که امکان ارجاع به سطر خاص از جدول یا امکان تشخیص یک نمونه موجودیت از بقیه را فراهم می‌کند.

توصیه‌هایی در مورد تعیین نوع داده فیلدها (ستون‌ها):

برای تعیین نوع داده یک فیلد، نخست باید مشخص کنیم که چه نوع داده‌ای قرار است در آن ذخیره شود (عدد، نویسه و ...). در مرحله دوم، باید محدوده مقادیر آن را، برای تمام نمونه‌های موجودیت، شناسایی کنیم. سپس عملیاتی که احتمالاً روی داده‌ها انجام می‌شود، تعیین می‌کنیم. بر این اساس، به راحتی نوع داده مناسب تشخیص داده می‌شود. به چند نمونه زیر دقت کنید:

- نام، نام خانوادگی: نوع داده Short Text با FieldSize ۳۰ نویسه کافی است.
- شماره تلفن و موبایل: عددی است، اما چون محاسبات ریاضی روی آن انجام نمی‌شود و صفر اول هم در حالت عدد ذخیره نمی‌شود، بهتر است نوع داده Short Text با FieldSize حدود ۱۱ نویسه تعیین شود.
- تاریخ شمسی: از آنجا که نوع داده Date/Time ذخیره تاریخ را به صورت میلادی فراهم می‌آورد لذا توصیه می‌شود نوع داده آن را Short Text انتخاب و الگوی ورودی آن را با استفاده از ویژگی Input mask با قالب مورد نظر تعیین کنید.
- کدهای عددی (شناسه موجودیت‌ها): عددی است و می‌توان آن را از نوع داده Number با FieldSize اندازه Long Integer یا به صورت شماره خودکار (AutoNumber) در نظر گرفت.
- تعداد واحد یک درس: عددی است و می‌توان آن را از نوع داده Number با FieldSize اندازه Byte در نظر گرفت.

پاسخ به فعالیت‌ها

به نظر شما دلیل وجود ستون Click to Add چیست؟
پاسخ: اگر روی آن کلیک کنیم، لیستی از انواع داده‌ها باز می‌شود و می‌توان با انتخاب از آن، ستون جدیدی با نوع داده انتخابی به جدول اضافه کرد.

کنجکاو
ص ۱۹

– چه روش دیگری برای تعیین کلید اصلی وجود دارد؟ اگر جدول را بدون ایجاد کلید اصلی ذخیره کنید چه اتفاقی می‌افتد؟
پاسخ: در سمت چپ فیلد مورد نظر راست کلیک کرده و گزینه Primary Key را انتخاب می‌کنیم.

کنجکاو
ص ۲۳

– اگر جدول را بدون ایجاد کلید اصلی ذخیره کنید چه اتفاقی می‌افتد؟
پاسخ: کلید اصلی، امکان ارجاع به سطر خاص از جدول یا امکان تشخیص یک نمونه موجودیت از بقیه را فراهم می‌کند. یکی دیگر از اهداف تعیین کلید اصلی در جدول، رعایت قاعده جامعیت است. یکی از مشکلاتی که با عدم تعریف کلید اصلی دچار آن می‌شویم افزونگی اطلاعات است. به عنوان مثال اگر کد دانش‌آموزی که منحصر به فرد است، به عنوان کلید اصلی معرفی نشود اطلاعات یک دانش‌آموز ممکن است چندین بار در جدول تکرار شود و متوجه نشویم.

در Access با عدم تعیین کلید اصلی، با پیامی مواجه می‌شویم که با تأیید آن (Yes) ستون کلید جدیدی با نام ID و به صورت AutoNumber (شماره خودکار) به جدول اضافه می‌شود. در صورت عدم تأیید (No)، جدول بدون کلید ذخیره می‌شود.

کنجکاوی
ص ۲۴

پیام هشدار هنگام حذف فیلد شامل داده را به همراه ترجمه آن بنویسید.
Do you want to permanently delete the selected field(s) and all the data in the field(s)? To permanently delete the field(s), click Yes.
پاسخ: آیا می‌خواهید فیلد(ها) انتخابی و همه داده‌های آن(ها) را به صورت دائمی حذف کنید؟ برای حذف دائمی فیلد(ها) روی دکمه Yes کلیک کنید.

کلید خارجی (Foreign key)

به طور کلی، اگر صفت (ستون) A در جدولی مانند Table1 کلید اصلی باشد و آن را در جدولی دیگر مانند Table2 استفاده کنیم، ستون A در جدول Table2 به عنوان کلید خارجی شناخته می‌شود.

در مثال سیستم مدرسه، در جدول student صفت شماره هنرجو (StudentCode)، در جدول course صفت کد درس (CourseCode) و در جدول teacher، صفت کد هنرآموز (TeacherCode) به عنوان کلید اصلی انتخاب می‌شوند. در جدول teach، چون صفات (StudentCode, TeacherCode, CourseCode) در جداول دیگر، کلید اصلی هستند و در این جدول استفاده شده‌اند، پس هر کدام به عنوان کلید خارجی محسوب می‌شوند.

نقش کلید خارجی: کلید خارجی ارتباط بین یک جدول که نمایش یک موجودیت است، و بقیه جدول‌ها را برقرار می‌کند؛ مثلاً کلید خارجی StudentCode در جدول teach، امکان ارتباط هنرجو با درس را مشخص می‌کند.

قاعده جامعیت ارجاعی (Referential Integrity) چیست؟

به زبان ساده، این قاعده پایگاه داده‌ای ناظر بر کلید خارجی است. بر اساس آن، مقدار وارد شده برای صفت (ستون) کلید خارجی حتماً باید در جدول مرجع یعنی جدول حاوی کلید اصلی وجود داشته باشد یا می‌تواند خالی باشد، به شرطی که جزء کلید اصلی نباشد. با فرض اینکه اطلاعات هنرجویان در جدول Student و درس‌ها در جدول Course ذخیره شده باشند. در این صورت، در جدول تدریس (teach)، در ستون StudentCode فقط کد هنرجویانی را می‌توان درج کرد که در جدول Student موجود باشند و همچنین در ستون CourseCode از این جدول، فقط کد درس‌هایی را می‌توان ثبت کرد که در جدول Course موجود باشند (شکل ۱۲).

قاعده جامعیت ارجاعی، سازگاری داده‌ها در جداول مرتبط را در هر زمان از حیات سیستم، تضمین می‌کند. بدین معنا که اگر کد هنرجویی در جدول Student تغییر کند (به‌روزرسانی)، در این صورت در جدول teach هم باید این تغییر منعکس شود

(به‌روزرسانی آبشاری Cascade). همچنین، اگر سطری از جدول اصلی (حاوی کلید اصلی) حذف شود، سطرهایی از جدول وابسته (حاوی کلید خارجی) که دارای همان مقدار کلید باشند، نیز حذف خواهند شد (حذف آبشاری).

| Student | | | | |
|-------------|-------------|---------------|------------|--------------|
| StudentCode | StudentName | StudentFamily | StudentTel | StudentGrade |
| ۱۱ | امیر | محمدی | ۴۴۵۵۶۶ | دهم |
| ۱۲ | حسین | علوی | ۵۵۶۶۳۳ | دهم |
| ۱۳ | پیمان | آزادی | ۳۳۲۲۱۱ | دهم |
| ۲۱ | ساسان | رحیمی | ۷۷۶۶۵۵ | یازدهم |

| Teacher | | | |
|-------------|-------------|---------------|------------|
| TeacherCode | TeacherName | TeacherFamily | TeacherTel |
| ۱۰۰ | حمید | محمدیان | ۶۶۸۸۹۹ |
| ۱۰۱ | محسن | علی زاده | ۶۶۸۸۸۸ |
| ۱۰۲ | جمشید | اصفهانی | ۶۶۷۷۵۵ |

| Course | | |
|------------|---------------------------|------|
| CourseCode | CourseName | Unit |
| ۱۰ | توسعه برنامه سازی | ۴ |
| ۲۰ | طراحی سیستم های اطلاعاتی | ۴ |
| ۳۰ | دانش قتی | ۲ |
| ۴۰ | الزامات محیط کار | ۲ |
| ۵۰ | تولید محتوا و برنامه سازی | ۴ |

این ردیف قابل ثبت نیست، چون هنرجویی با شماره ۱۶ در جدول هنرجویان موجود نیست!

این ردیف قابل ثبت نیست، چون درسی با شماره ۶۰ در جدول دروسها موجود نیست!

| teach | | | | |
|-------------|-------------|------------|---------------|------|
| StudentCode | TeacherCode | CourseCode | EducationYear | mark |
| ۱۱ | ۱۰۰ | ۱۰ | ۹۶-۹۷ | ۱۵.۵ |
| ۱۱ | ۱۰۰ | ۲۰ | ۹۶-۹۷ | ۱۶.۵ |
| ۱۲ | ۱۰۲ | ۳۰ | ۹۶-۹۷ | ۱۸ |
| ۱۲ | ۱۰۱ | ۴۰ | ۹۶-۹۷ | ۱۹ |
| ۳۱ | ۱۰۱ | ۶۰ | ۹۶-۹۷ | ۱۵ |
| ۱۶ | ۱۰۱ | ۱۰ | ۹۶-۹۷ | ۱۴ |

کارگاه ۴ – ایجاد ارتباط بین جدول ها

اصلاً چرا ما باید بین چند جدول ارتباط برقرار کنیم و چه ضرورتی برای انجام این کار است؟

در پاسخ می‌توان گفت که ارتباط بیانگر تعامل و رابطه بین موجودیت‌ها در محیط عملیاتی است. موجودیت‌ها در عمل به‌صورت جدول پیاده‌سازی می‌شوند و ارتباط آنها توسط کلیدهای خارجی برقرار می‌شود. همچنین با این کار، دقت، سازگاری، سرعت و کارایی فرآیند به‌روزرسانی رکوردها و فیلدها افزایش پیدا می‌کند.

پایه‌سازی جدول‌های بانک اطلاعاتی سیستم مدرسه در Access

پس از طراحی منطقی، نوبت به پایه‌سازی می‌رسد. با توجه به مشخصات جدول‌ها که

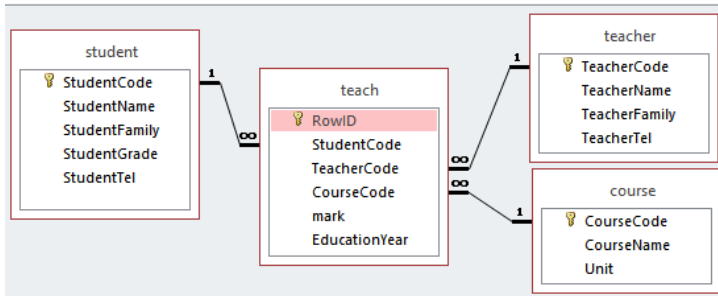
| teacher جدول هنرآموز | | | |
|----------------------|------------|--|--|
| Field Name | Data Type | بقیه مشخصات ستونها | |
| TeacherCode | Number | Field Size: Long Integer , Caption: کد هنرآموز | |
| TeacherName | Short Text | Field Size: 30 , Caption: نام هنرآموز | |
| TeacherFamily | Short Text | Field Size: 30 , Caption: نام خانوادگی هنرآموز | |
| TeacherTel | Short Text | Field Size: 11 , Caption: تلفن هنرآموز | |

| student جدول هنرجو | | | |
|--------------------|------------|--|--|
| Field Name | Data Type | بقیه مشخصات ستونها | |
| StudentCode | Number | Field Size: Long Integer , Caption: کد هنرجو | |
| StudentName | Short Text | Field Size: 30 , Caption: نام هنرجو | |
| StudentFamily | Short Text | Field Size: 30 , Caption: نام خانوادگی هنرجو | |
| StudentGrade | Short Text | Field Size: 20 , Caption: پایه تحصیلی | |
| StudentTel | Short Text | Field Size: 11 , Caption: شماره تلفن هنرجو | |

| course جدول درس | | | |
|-----------------|------------|--|--|
| Field Name | Data Type | بقیه مشخصات ستونها | |
| CourseCode | Number | Field Size: Long Integer , Caption: کد درس | |
| CourseName | Short Text | Field Size: 30 , Caption: نام درس | |
| Unit | Number | Field Size: Byte , Caption: تعداد واحد | |

| teach جدول تدریس | | | |
|------------------|------------|--|--|
| Field Name | Data Type | بقیه مشخصات ستونها | |
| RowID | AutoNumber | Field Size: Long Integer , Caption: شماره ردیف | |
| StudentCode | Number | Field Size: Long Integer , Caption: کد هنرجو | |
| TeacherCode | Number | Field Size: Long Integer , Caption: کد هنرآموز | |
| CourseCode | Number | Field Size: Long Integer , Caption: کد درس | |
| mark | Number | Field Size: Single , Caption: نمره | |
| EducationYear | Short Text | Field Size: 10 , Caption: سال تحصیلی | |

قبلاً طراحی شده‌اند (جدول ۲)، می‌توان جدول‌ها را طبق شکل ۱۳ پایه‌سازی کرد. همچنین برای اعمال یکپارچگی و جامعیت داده‌ها، باید طبق شکل ۱۴ پیوند جدول‌ها و ارتباط بین کلیدهای اصلی و خارجی را برقرار کرده و تنظیمات لازم را انجام داد.



Edit Relationships

Table/Query: teacher Related Table/Query: teach

TeacherCode TeacherCode

Enforce Referential Integrity
 Cascade Update Related Fields
 Cascade Delete Related Records

Relationship Type: One-To-Many

Edit Relationships

Table/Query: course Related Table/Query: teach

CourseCode CourseCode

Enforce Referential Integrity
 Cascade Update Related Fields
 Cascade Delete Related Records

Relationship Type: One-To-Many

Edit Relationships

Table/Query: student Related Table/Query: teach

StudentCode StudentCode

Enforce Referential Integrity
 Cascade Update Related Fields
 Cascade Delete Related Records

Relationship Type: One-To-Many

شرح گزینه های تنظیم ارتباط بین جدولها

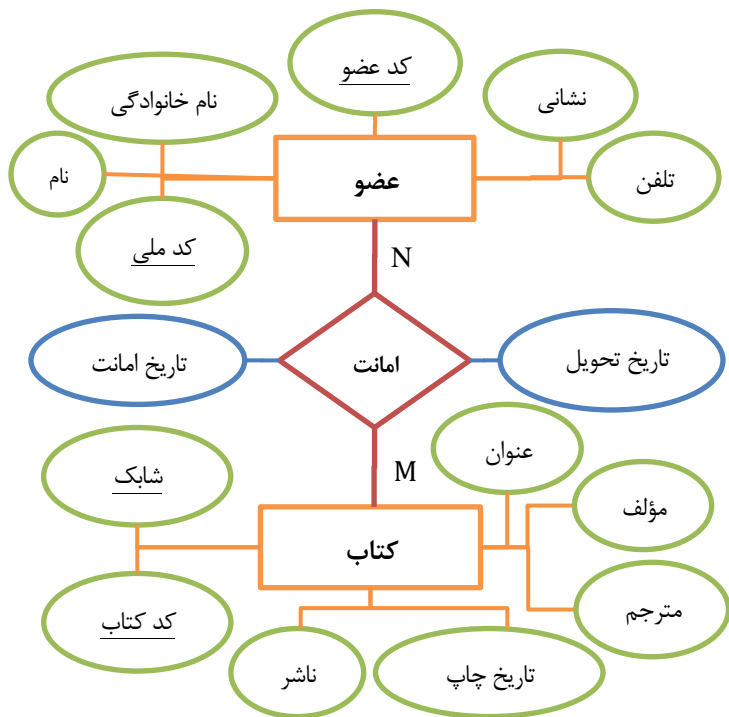
- (۱) جدول دارای کلید اصلی
- (۲) جدول دارای کلید خارجی
- (۳) اعمال قواعد جامعیت ارجاعی
- (۴) به روزرسانی آبخاری فیلدهای وابسته
- (۵) حذف آبخاری رکوردهای وابسته

پس از تدریس

الف) فعالیت تکمیلی

ارائه چند تمرین عملی مانند فروشگاه، آژانس تاکسی که قبلاً نمودار ساده شده آنها ارائه شد، با شروع از مدل سازی به کمک نمودار ER و سپس پیاده سازی جدول های مربوطه، به عنوان فعالیت تکمیلی در کلاس درس یا فعالیت منزل پیشنهاد می شود. به عنوان نمونه ای دیگر، در محیط عملیاتی کتابخانه، یک کتاب را چندین عضو می توانند به امانت ببرند و هر عضو هم می تواند چند کتاب را به امانت ببرد. نمونه ای ساده از مدل سازی ER یک کتابخانه را در شکل ۱۵ مشاهده می کنید.

دقت شود هر موجودیت ممکن است چند شناسه داشته باشد. در اینجا برای موجودیت عضو، دو صفت کد ملی و کد عضو، هر دو خاصیت یکتایی مقدار و شناسه بودن را دارند و طراح در مرحله طراحی جدول‌ها، به دلخواه یا بر اساس ملاحظات پیاده‌سازی، یکی را به عنوان کلید اصلی انتخاب می‌کند. این امر در مورد کد کتاب و شابک برای موجودیت کتاب هم صادق است.



ب) ارزشیابی پایانی

پیشنهاد می‌شود برای ارزشیابی پایانی این واحد یادگیری یک محیط عملیاتی مانند: آژانس تاکسی تلفنی، فروشگاه یا کتابخانه را انتخاب کرده و با طرح سوالاتی از مطالب کلیدی مطرح شده، سطح مهارت و شایستگی هنرجویان را مورد ارزیابی و سنجش قرار دهید. به عنوان نمونه در اینجا محیط عملیاتی فروشگاه را در نظر گرفته‌ایم (شکل ۵) و مطابق جدول زیر سطح شایستگی هنرجویان را مورد ارزیابی قرار می‌دهیم.

| نمره کسب شده | ایجاد پایگاه داده (۲- مرحله بحرانی) | | | تجزیه و تحلیل ارتباطها (۱) | | | تعیین موجودیت و صفات موجودیت (۱) | | | مراحل (حداقل نمره) | |
|--------------|--|-------------------------------------|-------------------|----------------------------|-----------------------------|------------------------------------|----------------------------------|---------------------------|----------------------|--------------------|------|
| | اصلاح ساختار جدول - رفع خطاهای احتمالی | ایجاد جدول، ایجاد ارتباط بین جدولها | ایجاد پایگاه داده | رسم نمودار ER | تعیین عملکرد و ماهیت ارتباط | تعیین موجودیت‌هایی که ارتباط دارند | تعیین صفت کلید هر موجودیت | تعیین همه صفات موجودیت‌ها | تعیین همه موجودیت‌ها | سوال / حداقل نمره | ردیف |
| ۴ | | | | | | | | | | ۱ | ۱ |
| ۲ | | | | | | | ✓ | ✓ | ✓ | ۲ | ۲ |
| ۱ | | | | ✓ | ✓ | ✓ | | | | ۳ | ۳ |

شایستگی توسعه پایگاه داده

مقدمات تدریس

ورود به بحث

هنرجو در این واحد یادگیری، قرار است با مفاهیم و عملیات ثبت، ویرایش، حذف و بازیابی اطلاعات و دسته‌بندی و گزارش‌گیری آشنا شود. ابتدا لازم است یک محیط عملیاتی مناسب در نظر گرفته شده و جدول‌ها طراحی و پیاده‌سازی شده باشند. سپس با طرح پرسش‌ها و نیازمندی‌های کاربری یک سیستم اطلاعاتی، اعمال مربوطه سازماندهی و در حین تدریس با داده‌های نمونه مفاهیم و کاربرد آنها تفهیم شود. تعیین سطح می‌تواند با روش‌های زیر و کنجکاو کردن هنرجویان آغاز شود:

- از هنرجویان سؤال کنید ثبت‌نام شما در سیستم مدرسه، چگونه انجام می‌شود؟
 - از هنرجویان سؤال کنید اگر اطلاعات ثبت‌نامی شما نیاز به ویرایش و اصلاح داشته باشد، چگونه این کار باید صورت گیرد؟
 - از هنرجویان بخواهید حدس بزنند چه تغییراتی را می‌توان بر روی یک جدول آماده اعمال کرد؟
 - دلیل انجام این تغییرات چیست؟
 - تهیه گزارش کارنامه آخر ترم شما چگونه انجام می‌شود؟
- سپس با توجه به پاسخ‌های هنرجویان و جمع‌بندی مطالب مطرحه، آنها را به سمت اعمال اساسی پایگاه داده (درج، حذف، ویرایش و بازیابی اطلاعات) راهنمایی کنید.

تدریس

کارگاه ۱ – اضافه کردن رکورد به جدول

زبان SQL (Structured Query Language) یک زبان استاندارد برای پیاده‌سازی، مدیریت، نگهداری و کار با بانک‌های اطلاعاتی رابطه‌ای (جدولی) است و تقریباً توسط تمام سیستم‌های کوچک و بزرگ که با پایگاه داده سروکار دارند، پشتیبانی می‌شود. این زبان دارای دستوراتی برای تعریف، ذخیره و به‌روزرسانی داده‌ها و طراحی پرس‌وجوها است. طراحان و افرادی که به نوعی با بانک‌های اطلاعاتی سروکار دارند و همچنین برنامه‌نویسانی که از این بانک‌ها استفاده می‌کنند، باید تا اندازه‌ای با این زبان آشنایی داشته باشند. البته این زبان به تنهایی کلیه نیازهای برنامه‌نویسان پایگاه داده‌ها را تأمین نمی‌کند و ممکن است یک زبان برنامه‌نویسی مکمل (مانند C#) برای طراحی برنامه‌های واسط کاربری

(منوها، فرمها، پردازشها، تهیه گزارشها و چاپ آنها) مورد نیاز باشد تا کاربر عادی بتواند از طریق این واسط کاربری با سیستم و دادهها کار کند.



مشکلات متداول در فرآیند یاددهی - یادگیری

ممکن است هنرجویان سوال کنند که وقتی به راحتی در محیط برنامه Access می توان اطلاعات را وارد، ویرایش یا حذف نمود، چه لزومی دارد با دستورات SQL این کارها را انجام دهیم؟

پاسخ: SQL یک زبان استاندارد برای ذخیره، ویرایش و بازیابی اطلاعات در پایگاه داده است. برخی اعمال مانند آمارگیری از دادهها و انتخاب اطلاعات بر اساس شرایط خاص، ویرایش و بهروزرسانی ستونها در تمام رکوردها و ... بدون SQL بسیار مشکل است.

پاسخ به فعالیتها

دو راه دیگر برای درج رکورد پیدا کنید.
پاسخ:

(۱) با تایپ مقادیر فیلدها در ردیف آخر رکوردها (علامت *) یک رکورد جدید ایجاد می شود.

(۲) کلیک روی دکمه NewRecord در پایین نمای Datasheet

کنجکای
ص ۳۴

با استفاده از دستور INSERT INTO رکوردهای زیر را به جدول teacher و student اضافه کنید.

INSERT INTO teacher (TeacherCode, TeacherName, TeacherFamily, TeacherTel)

VALUES (1011, 'علی', 'رنجبر', '34558896')

فعالیت کارگاهی
ص ۳۴

```
INSERT INTO student (StudentCode, StudentName, StudentFamily, StudentTel)
```

```
VALUES (156, 'حامد', 'عزیزی', '35429902')
```

- جدول teach را باز کرده و مقدار ۵۵۶۳ را برای فیلد کد درس، مقدار ۱۰۱۱ را برای فیلد کد کارمندی و مقدار ۱۵۳ را به برای فیلد کد دانش‌آموزی وارد کنید. پس از ثبت این رکورد چه اتفاقی رخ می‌دهد؟
پاسخ: اگر قبلاً درسی با کد ۵۵۶۳ در جدول درس‌ها (Course) ثبت نشده باشد، اطلاعات این ردیف ثبت نمی‌شود و با پیام خطا مواجه می‌شویم (به پاسخ سوال ۳ در بخش فعالیت تکمیلی پایانی مراجعه کنید).
- امکان ورود مقادیر تکراری برای فیلدهای کلید خارجی را بررسی کنید.
پاسخ: مشکلی ایجاد نمی‌کند. چون مثلاً یک هنرجو چند درس اخذ می‌کند و کد هنرجو ممکن است تکرار شود.
- امکان خالی بودن فیلدهای کلید خارجی را بررسی کنید.
پاسخ: اگر مقدار کلید خارجی خالی باشد، با پیام خطا مواجه می‌شویم و مفهوم آن این است که مقدار کلید خارجی باید با جدول اصلی مطابقت داشته باشد.

کارگاه ۲ - ویرایش رکوردها

در زمان ویرایش رکورد می‌توان از دکمه‌های قبلی (prev)، بعدی (next)، اولین (first)، آخرین (last) در پایین پنجره Datasheet استفاده کرد.

پاسخ به فعالیت‌ها

کنجکاو
ص ۳۵

- اگر از عبارت WHERE در دستور UPDATE صرف‌نظر شود، کدام رکوردهای جدول به‌روزرسانی خواهند شد؟
پاسخ: در این صورت تمام رکوردها (سطرها) ی جدول به‌روزرسانی خواهند شد.

فعالیت کارگاهی
ص ۳۵

- شماره تلفن هنرآموزی که در فعالیت‌های قبلی به جدول teacher اضافه شد را از ۳۴۵۵۸۸۹۶ به ۴۵۵۷۷۹۶ تغییر دهید.

```
UPDATE teacher  
'SET TeacherTel = '34557796'  
WHERE TeacherTel = '34558896'
```

- شماره دانش‌آموزی مربوط به هنرجویی با نام 'حامد عزیزی‌نیا' را به ۵۳ تغییر دهید.
UPDATE student
SET StudentCode = 53
WHERE StudentCode = 156
(عزیزی‌نیا' = StudentFamily) AND (StudentName='حامد') WHERE یا

کارگاه ۳ - حذف رکوردها

در نظر داشته باشید که اعمال روی داده‌های پایگاه داده قابل برگشت نیست. مگر اینکه با بازنشانی پشتیبان تهیه‌شده قبلی، داده‌ها را به وضعیت قبل برگردانیم.

پاسخ به فعالیت‌ها

روش دیگری برای حذف رکورد وجود دارد؟
پاسخ: با کلیک راست روی سطر موردنظر در نمای Datasheet و انتخاب گزینه Delete Record می‌توان یک رکورد را حذف کرد.

کنجکاوی
ص ۳۶

- با استفاده از دستور DELETE، رکورد هنرآموزی با کد کارمندی ۱۰۱۱ را حذف کنید.

```
DELETE FROM teacher  
WHERE TeacherCode = 1011
```

- اطلاعات تمامی هنرجویانی که کد دانش‌آموزی آنها دورقمی است را از جدول student حذف کنید.

پاسخ: منظور از کد دو رقمی یعنی بین ۱۰ تا ۹۹ باشد:

```
DELETE FROM student  
WHERE (StudentCode >=10) AND (StudentCode <100)
```

- همه رکوردهای جدول را حذف و نتیجه را بررسی کنید. آیا جدول حذف می‌شود؟
پاسخ: خیر جدول حذف نمی‌شود. فقط رکوردهای آن حذف می‌شود.

فعالیت کارگاهی
ص ۳۷

کارگاه ۵ - ایجاد پرس‌وجو با استفاده از دستور SELECT

پاسخ به فعالیت‌ها

با استفاده از دستور SELECT یک پرس‌وجو ایجاد کنید که فهرستی از هنرجویانی که دروسی را اخذ کرده‌اند، نمایش دهد. نام و نام خانوادگی آنها در یک فیلد باهم ادغام شده و شماره دانش‌آموزی هنرجویان نیز در خروجی نمایش داده شود.

```
SELECT DISTINCT student.StudentCode , StudentName + ' '  
+ StudentFamily AS StudentFullName, StudentTel  
FROM student , teach  
WHERE (student.StudentCode = teach.StudentCode)
```

فعالیت کارگاهی
ص ۴۲

کارگاه ۶ – مرتب‌سازی رکوردها

پاسخ به فعالیت‌ها

فعالیت کارگاهی
ص ۴۴

- با استفاده از دستورات SQL و عبارت ORDER BY جدول course را براساس نام دروس و سپس تعداد واحد آنها به صورت صعودی مرتب کنید.

```
SELECT CourseCode, CourseName, Unit  
FROM Course  
ORDER BY CourseName, Unit ASC
```

کارگاه ۷ – گروه‌بندی نتایج پرس‌وجو و استفاده از توابع تجمعی

دقت کنید که تمام فیلدهای جلوی عبارت SELECT غیر از آنهایی که در توابع تجمعی Sum, Min, Count استفاده شده‌اند، باید در لیست قسمت GROUP BY آورده شده و گروه‌بندی باید بر اساس فیلدهای مورد نظر پرس‌وجو باشد. برای شرح بیشتر به سوال ۹ و پاسخ آن در قسمت فعالیت تکمیلی مراجعه کنید.

پاسخ به فعالیت‌ها

فعالیت کارگاهی
ص ۴۷

دستور SELECT را به گونه‌ای بنویسید که نام و نام خانوادگی هنرجویان را براساس پایه تحصیلی و نام درس گروه‌بندی کرده به صورت توأم نمایش دهد.

```
SELECT  
StudentName,  
StudentFamily, StudentGrade, CourseName  
FROM student, teach, course  
WHERE (student.StudentCode = teach.StudentCode) AND  
(course.CourseCode = teach.CourseCode)  
GROUP BY StudentGrade, CourseName, StudentName,  
StudentFamily
```

کارگاه ۸ – ایجاد گزارش با استفاده از Wizard

در کتاب برای ایجاد گزارش روش Wizard بیان شده است؛ اما یک روش بسیار آسان و سریع برای ایجاد گزارش استفاده از Auto Reports است. این نوع گزارش فقط نام فیلدها و مقادیر متناظر را لیست می‌کند. مراحل کار به صورت زیر است:

- (۱) جدول یا پرس‌وجوی مورد نظر را انتخاب کنید.
- (۲) روی گزینه Reports از زبانه Create کلیک کنید.
- (۳) برنامه Access، گزارش مرتبط را ساخته و آن را در نمای Preview نشان می‌دهد.
- (۴) گزارش را ذخیره کنید.

Auto Report طراحی گزارش را ایجاد می‌کند و نیازی به استفاده از ابزارهای نمای Design و ساخت گزارش نداریم.

پروژه فاز (۱)

پایگاه داده کتابخانه

موجودیت‌های کتاب و عضو، ارتباط امانت و صفات آنها در نمودار ER در آخر مبحث واحد یادگیری اول در شکل ۱۵ ارائه شد. با توجه به نمودار ER، جدول‌های مربوطه را به شکل زیر طراحی می‌کنیم. جدول ۱- مشخصات جدول‌های سیستم کتابخانه مدرسه

| کلیدها | عنوان ستون‌ها | عنوان جدول |
|--|---|------------------------|
| کلید اصلی: <u>BookCode</u> | <u>BookCode</u> , Title, Subject, Author, Publisher, PublishYear | کتاب Books |
| کلید اصلی: <u>MemberCode</u> | <u>MemberCode</u> , FirstName, LastName, NationalCode, Tel, Address | عضو Members |
| کلید اصلی: <u>BookCode</u> , <u>MemberCode</u> کلیدهای خارجی: <u>BookCode</u> , <u>MemberCode</u> | BookCode, MemberCode, BorrowDate, DeliveryDate | امانت Borrow |

پیاده‌سازی جدول‌ها در Access، تعریف کلیدهای اصلی و خارجی و ارتباط بین جدول‌ها و قواعد جامعیت را مطابق شکل ۲۰ و ۲۱ تنظیم می‌کنیم:

Books مشخصات جدول کتاب

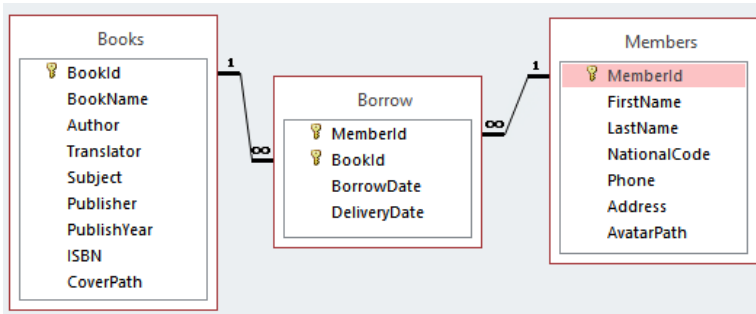
| Field Name | Data Type | |
|-------------|------------|----------------|
| BookId | AutoNumber | کد کتاب |
| BookName | Short Text | نام کتاب |
| Author | Short Text | نام مؤلف |
| Translator | Short Text | مترجم |
| Subject | Short Text | موضوع |
| Publisher | Short Text | ناشر |
| PublishYear | Short Text | سال چاپ |
| ISBN | Short Text | شماره شابک |
| CoverPath | Short Text | مسیر تصویر جلد |

Members مشخصات جدول عضو

| Field Name | Data Type | |
|--------------|------------|------------------|
| MemberId | AutoNumber | کد عضو |
| FirstName | Short Text | نام عضو |
| LastName | Short Text | نام خانوادگی عضو |
| NationalCode | Short Text | کد ملی |
| Phone | Short Text | تلفن |
| Address | Short Text | نشانی |
| AvatarPath | Short Text | مسیر عکس عضو |

Borrow مشخصات جدول امانت

| Field Name | Data Type | |
|--------------|------------|-------------|
| MemberId | Number | کد عضو |
| BookId | Number | کد کتاب |
| BorrowDate | Short Text | تاریخ امانت |
| DeliveryDate | Short Text | تاریخ تحویل |



Edit Relationships ? [X]

Table/Query: Books Related Table/Query: Borrow

BookId BookId

Enforce Referential Integrity

Cascade Update Related Fields

Cascade Delete Related Records

Relationship Type: One-To-Many

[OK] [Cancel] [Join Type..] [Create New..]

Edit Relationships ? [X]

Table/Query: Members Related Table/Query: Borrow

MemberId MemberId

Enforce Referential Integrity

Cascade Update Related Fields

Cascade Delete Related Records

Relationship Type: One-To-Many

[OK] [Cancel] [Join Type..] [Create New..]

مشخصات چندین کتاب، عضو و امانت را در جدول‌ها وارد کنید (هرکدام حدود ۱۰ رکورد).
به نمونه‌های شکل ۲۲ دقت کنید.

| سال چاپ | ناشر | موضوع کتاب | مترجم | نام مؤلف | نام کتاب | کد کتاب |
|---------|-------------------------------|-----------------|-------|-----------------|-----------------------------|---------|
| ۱۳۹۶ | چاپ و نشر کتاب های درسی ایران | برنامه نویسی | | مجید کربلایی | برنامه سازی پاسکال (۲) | ۱ |
| ۱۳۹۲ | چاپ و نشر کتاب های درسی ایران | برنامه نویسی وب | | زهرا سلیمی زاده | بسته های نرم افزاری (۳) | ۲ |
| ۱۳۹۴ | چاپ و نشر کتاب های درسی ایران | برنامه نویسی | | مجید کربلایی | برنامه سازی (۱) | ۳ |
| ۱۳۸۶ | جلوه | پایگاه داده | | محمد تقی روحانی | مفاهیم بنیادی پایگاه دادهها | ۴ |
| ۱۳۸۵ | چاپ و نشر کتاب های درسی ایران | پایگاه داده | | محمد رضا یقانی | بانک اطلاعاتی | ۵ |

| نشانی | تلفن | کد ملی | نام خانوادگی عضو | نام عضو | کد عضو |
|-------|--------|------------|------------------|---------|--------|
| تهران | ۳۳۴۴۵۵ | ۱۲۳۴۵۶۷۸۹۰ | محمدی | پهروز | ۱ |
| تهران | ۳۳۴۴۶۶ | ۹۰۸۷۶۵۴۳۲۱ | میرزایی | بهرام | ۲ |
| تهران | ۳۳۴۴۷۷ | ۸۰۹۷۶۵۴۳۲۱ | احمدی | امیر | ۳ |

| تاریخ تحویل | تاریخ امانت | کد کتاب | کد عضو |
|-------------|-------------|---------|--------|
| | ۱۳۹۷/۰۴/۱۱ | ۱ | ۱ |
| | ۱۳۹۷/۰۳/۳۰ | ۲ | ۱ |
| ۱۳۹۷/۰۴/۰۷ | ۱۳۹۷/۰۳/۳۰ | ۳ | ۲ |
| | ۱۳۹۷/۰۴/۰۸ | ۴ | ۲ |

حال می‌خواهیم برای موارد زیر پرس‌وجو طراحی کنیم:
 • فهرست اعضا

`SELECT * FROM Members;`

• فهرست کتاب‌ها

`SELECT * FROM Books;`

• فهرست کتاب‌هایی با موضوع برنامه نویسی

`SELECT * FROM Books`

`WHERE Subject = 'برنامه نویسی';`

• فهرست اعضای که کتابی را به امانت گرفته‌اند.

منظور اعضای است که در ارتباط امانت مشارکت دارند. از علامت * برای انتخاب و نمایش همه ستون‌های جدول مورد نظر استفاده می‌کنیم. برای اینکه سطرهای تکراری یک بار ظاهر شوند، واژه DISTINCT به کار رفته است:

`SELECT DISTINCT Members.*`

`FROM Members, Borrow`

`WHERE Members.MemberCode = Borrow.MemberCode;`

• فهرست اعضای که هنوز کتاب امانت گرفته شده را برنگردانده‌اند.

برنگرداندن کتاب به این معناست که فیلد تاریخ تحویل خالی باشد که توسط عملگر IS NULL در SQL، بررسی خالی بودن یک ستون انجام می‌گیرد.

`SELECT DISTINCT Members.*`

```
FROM Members, Borrow
WHERE (Members.MemberCode = Borrow.MemberCode)
AND (DeliveryDate IS NULL);
```

• فهرست اعضا و تعداد کتابهایی که تاکنون امانت گرفته‌اند.

تعداد کتاب‌های امانت گرفته شده هر عضو با گروه‌بندی آنها توسط عبارت GROUP BY و تابع تجمعی COUNT() محاسبه می‌شود:

```
SELECT Members.MemberCode, FirstName, LastName,
COUNT(*) AS [تعداد کتاب]
```

```
FROM Members, Borrow
WHERE Members.MemberCode = Borrow.MemberCode
GROUP BY Members.MemberCode, FirstName, LastName;
```

• فهرست کتاب‌های امانت گرفته‌شده به همراه اسامی کسانی که کتاب را امانت گرفته‌اند.

چون مشخصات کتاب‌ها، اعضا و امانت در سه جدول مجزا قرار دارند، پس باید سه جدول را با هم ترکیب کرد و شرط پیوند دوبه‌دوی آنها را در عبارت شرطی نوشت:

```
SELECT Books.BookCode, Title, Subject, Author, FirstName,
LastName
```

```
FROM Books, Members, Borrow
WHERE (Books.BookCode = Borrow.BookCode)
AND (Members.MemberCode = Borrow.MemberCode);
```

توجه: پایگاه داده فاز یک پروژه (پایگاه داده کتابخانه) در پرونده پیوست لوح همراه کتاب موجود است.

پس از تدریس

الف) نمونه فعالیت تکمیلی

با فرض اینکه قبلاً بانک اطلاعاتی مدرسه را پیاده‌سازی کرده باشیم، حالا می‌توانیم با طرح پرسش و سناریوهایی از محیط واقعی یک سیستم کامپیوتری مبتنی بر پایگاه داده، نیازمندی‌های کاربر سیستم را برآورده سازیم. برای انگیزه بیشتر، هنرجویان را تشویق کنید که خود را مثلاً به جای مسئولین مدرسه تصور کنند و ببینند که از سیستم چه انتظاراتی دارند و چه عملیات و گزارش‌هایی را می‌توانند از سیستم بخواهند. سپس پرسش‌ها، انتظارات و نیازمندی‌ها را یادداشت و دسته‌بندی کنید و در لابه‌لای این واحد یادگیری در کارگاه‌ها پاسخ آنها را پیدا کنید.

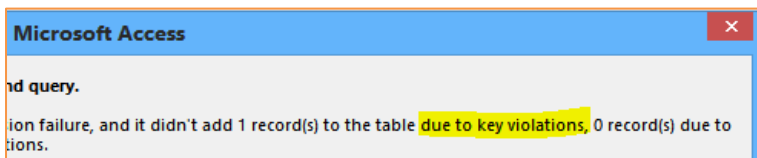
برای شروع می‌توانید از پرسش‌ها و نیازمندی‌های زیر استفاده کنید که پاسخ آنها را در ادامه خواهید دید:

۱) یک هنرجوی جدید را چگونه ثبت نام کنیم؟ اگر هنگام ثبت نام یک هنرجو، کد دانش‌آموزی تکراری وارد شود، چه اتفاقی می‌افتد؟
پاسخ: فرض کنید جدول Student برای ثبت اطلاعات هنرجویان طراحی شده است. از دستور زیر برای ثبت اطلاعات هنرجو استفاده می‌شود:

```
INSERT INTO student (StudentCode, StudentName,  
StudentFamily, StudentGrade, StudentTel)  
VALUES (11, 'امیر', 'محمدی', 'دهم', '445566')
```

اگر قبلاً هنرجویی با کد "۱۱" ثبت شده باشد، ستون شناسه (کلید اصلی) نمی‌تواند تکراری باشد، در نتیجه سیستم با خطا مواجه می‌شود و باید یک کد غیر تکراری جدید به آن اختصاص دهیم تا دستور اجرا شود.

بخشی از متن پیام خطای تکراری بودن مقدار ستون کلید به صورت زیر است:



یا

... duplicate values in the index, primary key, or relationship....

۲) اخذ درس «توسعه و برنامه‌سازی» با کد ۴۰ برای هنرجویی با کد ۱۱، که هنرآموزی با کد ۱۰۱ آن را در سال تحصیلی جاری تدریس می‌کند، ثبت کنید.

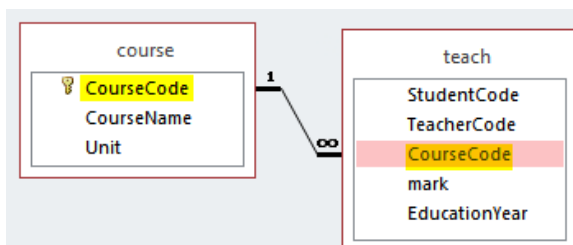
پاسخ: اطلاعات ثبت این درس به صورت زیر است:

```
INSERT INTO teach (StudentCode, TeacherCode, CourseCode,  
EducationYear)  
VALUES(11, 101, 40, '96-97')
```

اگر درسی با کد "۴۰" در جدول درس‌ها (course) موجود نباشد، اطلاعات ثبت نمی‌شوند و پیام خطا (مشابه مورد سوال قبل) صادر می‌شود. یعنی نمی‌توان درسی را اخذ نمود که موجود نیست (ثبت نشده است). این قانون و کنترل توسط تعریف ارتباط بین موجودیت‌ها و تعیین کلیدهای خارجی اعمال می‌شود (شکل ۱۶). پس باید اول این درس را در جدول درس‌ها ثبت کرد و سپس اخذ درس را انجام داد.

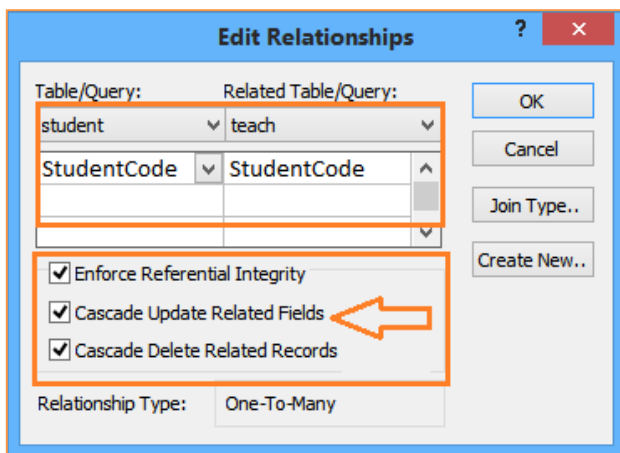
۳) اگر قسمتی از اطلاعات یک هنرجو (مانند کد هنرجو) حین ثبت نام، اشتباه ثبت شده باشد، برای ویرایش و اصلاح آن چه باید کرد؟
 پاسخ: اگر هنرجوی مورد نظر قبلاً با کد ۱۵۳ ثبت نام شده ولی کد ۵۳ صحیح است، با دستور زیر اصلاح می شود:

```
UPDATE student
SET StudentCode = 53
WHERE StudentCode = 153
```



اگر هنرجویی با شرایط تعیین شده در عبارت WHERE موجود نباشد، دستور اجرا خواهد شد اما بر هیچ سطر تأثیر نخواهد گذاشت.

همچنین اگر این هنرجو قبلاً درس هایی هم اخذ کرده باشد (جدول teach)، در هر ردیفی که کد هنرجو (StudentCode) ثبت شده باشد، به دلیل تعیین کلید خارجی و قاعده جامعیت ارجاعی (شکل ۱۷)، به صورت خودکار توسط سیستم، کد آن از ۱۵۳ به ۵۳ اصلاح خواهد شد.



۴) اگر پایه تحصیلی تعداد زیادی از هنرجویان (از کد ۱۵ تا ۲۳) اشتباه ثبت شده باشد، روش سریع و آسان برای اصلاح آن چیست؟ اگر فقط دو هنرجوی با کد ۱۵ و ۲۳ نیاز به اصلاح داشته باشند چگونه عمل کنیم؟
 پاسخ: اگر فرض کنیم پایه تحصیلی هنرجویان با کد ۱۵ تا ۲۳ به جای "یازدهم"، "دهم" ثبت شده باشد، با دستور زیر اصلاح می‌شود:

```
UPDATE student
SET StudentGrade = 'یازدهم'
WHERE (StudentCode >= 15) AND (StudentCode <=23)
```

شرط‌هایی که روی یک بازه یا محدوده پیوسته هستند، به صورت زیر نوشته می‌شوند:
 (مقدار پایان <= نام ستون) AND (مقدار شروع >= نام ستون)
 (StudentCode >=15) AND (StudentCode <=23)

برای کنترل یک بازه یا محدوده پیوسته از عبارت BETWEEN هم می‌توان استفاده کرد که شامل مقدار ابتدا و انتهای محدوده هم می‌شود:

```
مقدار پایان AND مقدار شروع BETWEEN نام ستون
StudentCode BETWEEN 15 AND 23
```

اگر فقط هنرجویان با کد ۱۵ و ۲۳، نیاز به اصلاح دارند، با ترکیب منطقی OR شرط را می‌نویسیم تا هر دو نفر را شامل شود. دقت کنید که اگر ترکیب منطقی AND را به کار ببریم، دستور عمل نمی‌کند، چون معنای شرط، هنرجویانی خواهد بود که همزمان دارای کد ۱۵ و ۲۳ باشند!

```
...
WHERE (StudentCode=15) OR (StudentCode =23)
```

۵) فرض کنید طبق بخشنامه جدید، کد هنرجو به جای دو رقمی، چهار رقم است. یعنی به هر کد هنرجو باید عدد ۱۰۰۰ اضافه شود. این کار را چگونه انجام دهیم؟
 پاسخ: اصلاح کد هنرجویان با دستور زیر انجام می‌شود:

```
UPDATE student
SET StudentCode = StudentCode + 1000
```

چون همه کدها باید اصلاح شوند، شرط و عبارت WHERE را برمی‌داریم.

۶) اگر هنرآموزی به دلیل ناهمانگی وقت خود با برنامه هفتگی مدرسه، نتواند همکاری خود را ادامه دهد، چگونه اطلاعات او را از سیستم آموزشی مدرسه حذف کنیم؟
 پاسخ: حذف هنرآموز (مثلاً با کد ۱۰۹) با دستور زیر انجام می‌شود:

```
DELETE FROM teacher WHERE (TeacherCode = 109)
```

اگر هنرآموزی با شرط تطبیق نکند، دستور اجرا خواهد شد؛ اما تأثیری روی اطلاعات جدول teacher نخواهد داشت.

۷) فرض کنید هنرجویی پس از ثبت نام و اخذ درس‌ها، به دلیل مشکلات رفت‌وآمد، به هنرستان دیگری منتقل می‌شود. چگونه می‌توان اطلاعات ثبت نامی او را حذف کرد؟ پاسخ: حذف هنرجوی موردنظر (مثلاً با کد ۱۸) با دستور زیر انجام می‌شود:

```
DELETE FROM student WHERE (StudentCode = 18)
```

با اجرای دستور فوق، اطلاعات هنرجو از جدول Student حذف می‌شود. همچنین به دلیل تعیین کلید خارجی و قاعده جامعیت ارجاعی (شکل ۱۵ گزینه آخر)، به‌صورت خودکار توسط سیستم، اطلاعات اخذ درس‌های مربوط به این هنرجو نیز حذف خواهند شد.

۸) لیست حضور و غیاب هنرجویان در سال تحصیلی جاری برای یک درس خاص را به ترتیب الفبایی نام خانوادگی و سپس نام آنها تهیه کنید.

پاسخ: برای نمایش هنرجویانی که درس اخذ کرده‌اند، باید دو جدول هنرجو و تدریس را بر اساس ستون مشترک (کد هنرجو) به هم مرتبط ساخت که در عبارت شرطی WHERE به صورت زیر نوشته می‌شود:

```
WHERE (student.StudentCode = teach.StudentCode)
```

شرط مسأله هم روی سال تحصیلی جاری و کد درس مورد نظر (مثلاً کد ۲۰) تنظیم می‌شود که با عملگر منطقی AND با شرط پیوند جدول‌ها ترکیب می‌شود. در آخر توسط عبارت ORDER BY بر اساس نام خانوادگی و سپس نام خروجی مرتب خواهد شد.

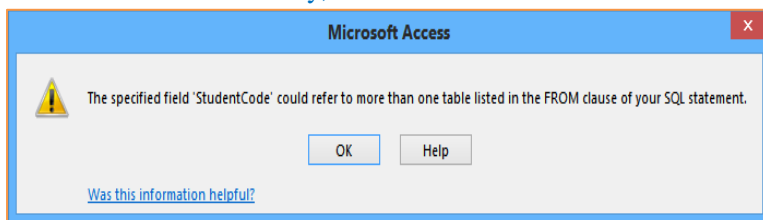
```
SELECT student.StudentCode, StudentName, StudentFamily
```

```
FROM student, teach
```

```
WHERE (student.StudentCode = teach.StudentCode)
```

```
AND (EducationYear='96-97') AND (CourseCode=20)
```

```
ORDER BY StudentFamily, StudentName
```



در ضمن چون کد هنرجو (StudentCode) در هر دو جدول وجود دارد، برای جلوگیری از ابهام، باید نام جدول را همراه با نقطه کنار نام ستون مشترک نوشت (student.StudentCode). در غیر این صورت با خطای زیر روبه‌رو خواهیم شد:

۹) عداد هنرجویان هر پایه تحصیلی را به تفکیک به دست آورید.
پاسخ: برای به دست آوردن تعداد و شمارش سطرها، تابع محاسباتی COUNT(*) به کار می‌رود. اگر بخواهیم بر اساس ستون خاص شمارش صورت گیرد، نام ستون داخل پرانتز نوشته می‌شود. برای انجام شمارش روی هر پایه تحصیلی باید از گروه‌بندی (GROUP BY) استفاده کنیم. پس پایه تحصیلی را جلوی عبارت SELECT می‌نویسیم و عیناً این ستون را جلوی عبارت GROUP BY قید می‌کنیم.

```
SELECT StudentGrade, COUNT(*) AS [تعداد هنرجو]
```

```
FROM student
```

```
GROUP BY StudentGrade
```

عبارت GROUP BY بر اساس ستون‌های مورد نظر، مقادیر ستون‌های استفاده شده در تابع تجمعی را در نظر گرفته، محاسبات مربوطه را انجام داده و دسته‌بندی می‌کند. عملکرد آن را در شکل ۱۸ می‌بینید.

۱۰) تعداد هنرجویانی که در هر درس اخذ شده خود، نمره بالای ۱۷ را بدست آورده‌اند، به تفکیک درس نمایش دهید.

پاسخ: تعداد هنرجویان از تابع COUNT روی کد هنرجو، بدست می‌آید. برای انجام شمارش روی هر درس باید از گروه‌بندی (GROUP BY) استفاده کنیم. مشخصات درس را جلوی عبارت SELECT می‌نویسیم و عیناً ستون‌های خواسته شده را جلوی عبارت GROUP BY قید می‌کنیم. چون اطلاعات از دو جدول به دست می‌آیند، در عبارت شرطی WHERE شرط پیوند جدول‌ها و نیز شرط مسئله که روی نمره (mark) بالای ۱۷ است، با AND ترکیب می‌کنیم:

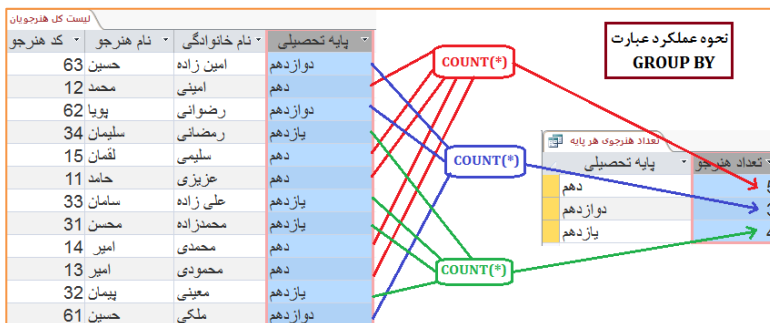
```
SELECT CourseName, COUNT(StudentCode) AS [تعداد هنرجو]
```

```
FROM course, teach
```

```
WHERE (course.CourseCode= teach.CourseCode)
```

```
AND (mark>17)
```

```
GROUP BY CourseName;
```



۱۱) مدیر مدرسه می خواهد برای دروسی که در آنها تعداد هنرجویان با نمره زیر ۱۲، بیش از ۱۰ نفر باشد، کلاس های تقویتی با همکاری هنرآموزان تشکیل دهد. لیست دروس را تهیه کنید.

پاسخ: پاسخ این سوال مشابه سوال قبلی است ولی شرط مسئله روی نمرات زیر ۱۲ است. برای اعمال شرط روی تعداد محاسبه شده، نمی توان از عبارت WHERE استفاده کرد و عبارت شرطی HAVING در پایان دستورات برای این منظور به کار می رود که روی نتایج، شرط تعداد بالای ۱۰ نفر (COUNT(StudentCode)>10) را اعمال می کند.

SELECT CourseName, COUNT(StudentCode) AS [تعداد هنرجو]

FROM course, teach

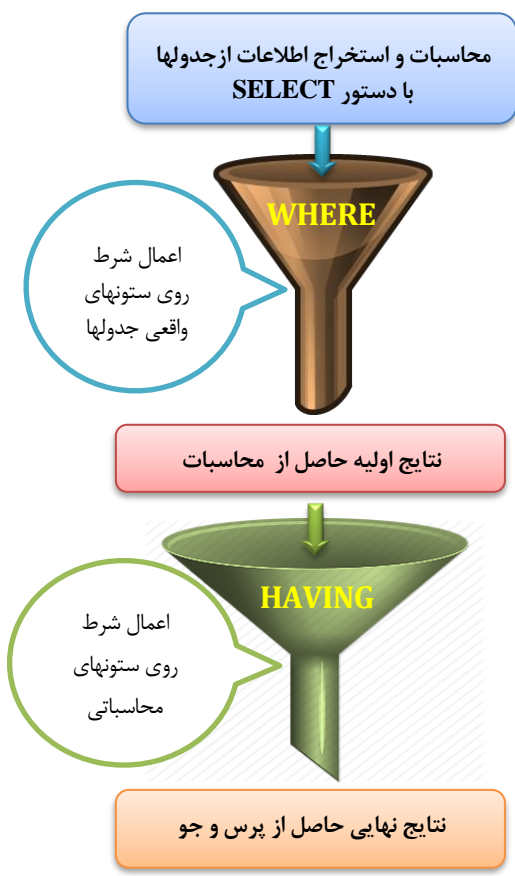
WHERE (course.CourseCode= teach.CourseCode)

AND (mark<12)

GROUP BY CourseName

HAVING COUNT(StudentCode)>10;

نکته مهم و قابل بحث این است که هرگاه بخواهیم شرطی را روی ستون های واقعی جدول ها، اعمال کنیم آن را می توان در عبارت WHERE نوشت. اما اگر شرط روی نتایج محاسباتی اعمال شود (به ویژه همراه عبارت GROUP BY) حتماً باید از عبارت HAVING برای نوشتن شرط استفاده کرد (شکل ۱۹). عبارت WHERE پس از عبارت FROM و قبل از GROUP BY ولی عبارت HAVING پس از GROUP BY نوشته می شود.



۱۲) لیست هنرجویان به همراه معدل آنها را نمایش دهید. سپس لیست را به هنرجویانی محدود کنید که معدل بالای ۱۵ دارند.

پاسخ: برای محاسبه معدل، باید مجموع حاصلضرب نمره در تعداد واحد هر درس را بر مجموع واحدها تقسیم نمود. از تابع تجمعی SUM برای این منظور استفاده می‌شود. مشخصات هنرجو را جلوی عبارت SELECT به همراه توابع محاسبه معدل نوشته و چون اطلاعات لازم برای محاسبه معدل در سه جدول درس، هنرجو و تدریس قرار دارند، پس در شرط WHERE باید پیوند جدولها را با عملگر منطقی AND بنویسیم. عبارت GROUP BY را هم برای گروه‌بندی بر اساس هنرجو به کار می‌بریم. در نهایت برای نمایش هنرجویان بر اساس شرط معدل بالای ۱۵ باید از عبارت HAVING استفاده کرد.

```

SELECT student.StudentCode, StudentName, StudentFamily,
SUM(mark*Unit)/SUM(Unit) AS [معدل هنرجو]
FROM student, teach, course
WHERE (course.CourseCode= teach.CourseCode)
AND (student.StudentCode = teach.StudentCode)
GROUP BY student.StudentCode, StudentName, StudentFamily
HAVING (SUM(mark*Unit)/SUM(Unit) > 15)

```

نمونه فعالیت تکمیلی:

الف) با توجه به نمودار ER محیط عملیاتی فروشگاه که در شکل (۵) صفحه ۳۲ ارائه شد، جدول‌های مربوطه را به صورت زیر طراحی کنید:

| کلیدها | عنوان ستون‌ها | عنوان جدول |
|---|---|--------------------------|
| کلید اصلی: <u>PCode</u> | <u>PCode</u> , PName, Model, BuyPrice, SalePrice | کالا Product |
| کلید اصلی: <u>CCode</u> | <u>CCode</u> , FirstName, LastName, City, Tel, Address | مشتری Customer |
| کلید اصلی: <u>RowID</u> کلیدهای خارجی: PCode , CCode | <u>RowID</u> , PCode , CCode , Quantity, SaleDate | فروش Sales |

ب) ارتباط بین جدول‌ها، کلیدهای خارجی و قواعد جامعیت را مشابه سیستم کتابخانه، که در شکل ۲۱ انجام شد، تنظیم کنید.

برای جدول فروش (Sales) به جای کلید ترکیبی کد کالا و کد مشتری، از شماره فروش (به صورت شماره خودکار) به عنوان کلید اصلی استفاده کرده‌ایم. دلیل این است که یک مشتری ممکن است در زمان‌های مختلف یک کالا را چند بار خریداری کند که در این صورت ترکیب کد کالا و کد مشتری با هم تکراری خواهد شد.

ج) اطلاعات مربوط به مشخصات حداقل ۵ مورد کالا (مثلاً نوشت افزار با کد ۱۰۱ تا ۱۰۵)، ۵ مورد مشتری (با کد ۱۰۰۱ تا ۱۰۰۵) و ۱۰ مورد فروش کالا به مشتریان را به صورت دستی در محیط Access ثبت کنید. به نمونه‌های شکل ۲۳ دقت کنید.

| ثبت چند نمونه کالا | | | | |
|--------------------|-------------|----------|-----------|-----------|
| کد کالا | نام کالا | مدل کالا | قیمت خرید | قیمت فروش |
| ۱۰۱ | دفتر ۴۰ برگ | ته چسب | ۱۶۰۰ | ۲۰۰۰ |
| ۱۰۲ | دفتر ۶۰ برگ | ته چسب | ۲۱۰۰ | ۲۵۰۰ |

| ثبت چند نمونه مشتری | | | | |
|---------------------|------|--------------|-------|---------------|
| کد مشتری | نام | نام خانوادگی | شهر | آدرس |
| ۱۰۰۱ | محسن | امیری | تهران | خیابان انقلاب |
| ۱۰۰۲ | رضا | حسینی | کرج | خیابان آزادی |

| ثبت چند نمونه فروش کالا | | | | |
|-------------------------|-------|---------|----------|------------|
| تاریخ | تعداد | کد کالا | کد مشتری | شماره فروش |
| ۱۳۹۶/۰۵/۰۸ | ۳ | ۱۰۱ | ۱۰۰۱ | ۱ |
| ۱۳۹۶/۰۶/۰۶ | ۴ | ۱۰۲ | ۱۰۰۲ | ۲ |
| ۱۳۹۶/۰۶/۱۲ | ۵ | ۱۰۲ | ۱۰۰۱ | ۳ |

د) به پرسش‌ها و نیازمندی‌های زیر با نوشتن دستورات به زبان SQL پاسخ دهید.

- ۱) اطلاعات یک نوع مداد با مشخصات زیر را به کمک دستور INSERT در جدول کالاها ثبت کنید. اگر قبلاً کالایی با کد ۱۱۹ ثبت شده باشد، آیا اطلاعات به درستی ثبت می‌شود؟
 مشخصات کالا → (مداد، 'B', 'سری B', 400, 650, 119)
- ۲) اطلاعات فروش ۵ عدد «مداد رنگی ۱۲ رنگ کتابی» با کد ۱۱۲ به یک مشتری با کد ۱۱۲۳ را در تاریخ ۱۰ شهریور ۱۳۹۷ ثبت کنید. اگر مشتری با کد ۱۱۲۳ در سیستم موجود نباشد، چه اتفاقی می‌افتد؟
- ۳) اگر فروشنده بخواهد به میزان ۱۰٪ از قیمت فروش دفترهای ۴۰ و ۶۰ برگ تخفیف بدهد، به کمک دستور UPDATE این درخواست را انجام دهید. اگر بخواهیم این تخفیف شامل همه کالاها شود، چه تغییری در دستور می‌دهید؟
- ۴) فروشنده به اشتباه سطر مربوط به مشتری شماره ۱۰۹۹ را در جدول مشتریان ثبت کرده‌است. دستوری بنویسید که اطلاعات این سطر حذف شود.
- ۵) یک مشتری وارد فروشگاه شده و می‌خواهد قیمت همه نوع دفترهای موجود در فروشگاه را به ترتیب قیمت بداند تا بتواند در مورد خرید خود تصمیم بگیرد. با نوشتن دستور SQL مربوطه به فروشنده کمک کنید.
- ۶) فروشنده می‌خواهد بداند چه مشتریانی «بسته مداد رنگی ۱۲ رنگ» را از فروشگاه خریداری کرده‌اند. دستور SQL مربوطه را چگونه می‌نویسید که با اجرای آن فروشنده لیست را تهیه کند؟
- ۷) لیست مشتریانی را که در شش ماهه نخست سال (یعنی از تاریخ اول فروردین تا آخر شهریور) خرید داشته‌اند، نمایش دهید.

- ۸) فاکتور فروش یک مشتری با کد ۱۰۰۱ را به صورت جدول شکل (۲۴) تهیه کنید و نمایش دهید.
- ۹) مبلغ قابل پرداخت هر مشتری را به تفکیک نمایش دهید.
- ۱۰) فروشنده می‌خواهد تعداد کل و مبلغ کل حاصل از فروش مداخلهای خود را تهیه کند. در انجام این کار به فروشنده با نوشتن دستور SQL مربوطه کمک کنید.
- ۱۱) لیست همه کالاهایی که در ماه شهریور و مهر به فروش رفته‌اند به ترتیب تاریخ نمایش دهید.
- ۱۲) میزان سود حاصل از فروش کالاهای فروشگاه را به تفکیک کالا نمایش دهید.

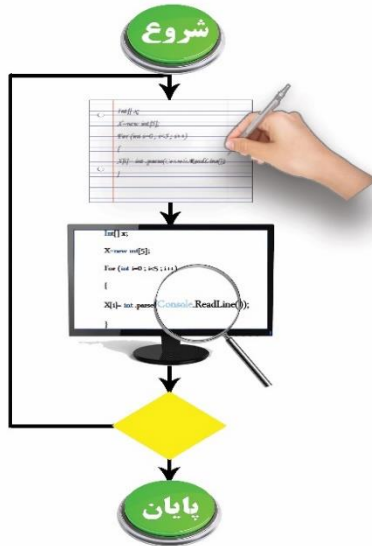
| فاکتور مشتری | | | | | |
|--------------|------------------|----------|-------|-----------|---------|
| کد کالا | نام کالا | مدل کالا | تعداد | قیمت فروش | مبلغ کل |
| ۱۰۱ | دفتر ۴۰ برگ | ته چسب | ۳ | ۲۰۰۰ | ۶۰۰۰ |
| ۱۰۲ | دفتر ۶۰ برگ | ته چسب | ۵ | ۲۵۰۰ | ۱۲۵۰۰ |
| ۱۰۳ | دفتر ۱۰۰ برگ | ته چسب | ۴ | ۳۰۰۰ | ۱۲۰۰۰ |
| ۱۰۴ | دفتر ۲۰۰ برگ | ته چسب | ۲ | ۴۰۰۰ | ۸۰۰۰ |
| ۱۱۷ | مداد رنگی ۱۲ رنگ | لوله ای | ۱ | ۱۵۰۰۰ | ۱۵۰۰۰ |
| ۱۲۲ | مداد فشاری | ۰.۵ mm | ۳ | ۱۰۰۰۰ | ۳۰۰۰۰ |

ب) ارزشیابی پایانی

پیشنهاد می‌شود یکی از محیط‌های عملیاتی آژانس تاکسی تلفنی، فروشگاه یا کتابخانه را در نظر گرفته و مطابق جدول زیر سطح شایستگی هنرجویان را مورد ارزیابی قرار دهید. برای ارزشیابی این واحد یادگیری محیط عملیاتی فروشگاه لوازم التحریر انتخاب شده است. به ازای هر هنرجو یا گروهی از آنها می‌توانید پرسش‌های هر دسته را عوض کنید. به دو نمونه مشابه زیر دقت کنید:

- اطلاعات یک نمونه دفتر را درج کنید.
- اطلاعات یک نمونه مشتری را درج کنید.

| مراحل (حداقل نمره) | | درج و ویرایش رکورد (۲ - مرحله بحرانی) | استخراج داده‌ها و مرتب‌سازی آنها (۱) | ایجاد گزارش (۱) | | | نمره کسب شده |
|--------------------|---|---------------------------------------|--------------------------------------|------------------|-------------------------|----------------------------------|--------------|
| ردی ف | سوال / حداقل نمره | حذف رکورد | درج و ویرایش رکورد | رفع خطای دستورات | مرتب‌سازی رکوردهای جدول | تولیع تجمعی - رفع خطاهای احتمالی | |
| | | ۱ | ۲ | ۳ | ۱ | ۲ | ۳ |
| ۱ | اطلاعات یک نمونه دفتر را درج کنید. - مشتری شماره ۵ را حذف کنید. - از قیمت خودکارها ۱۰۰ تومان کم کنید. | ✓ | ✓ | ✓ | | | |
| ۲ | - مشخصات همه دفترها را به ترتیب صعودی قیمت فروش نمایش دهید. - مشخصات کالاهای خریداری شده مشتری شماره ۳ (فاکتور) را نمایش دهید. - لیستی تهیه کنید که از هر کالا چه تعداد فروش رفته است. | | | | ✓ | ✓ | ✓ |
| ۳ | - گزارش "فاکتور مشتری شماره ۳" را با استفاده از wizard ایجاد کنید. - فونت و رنگ عنوان گزارش را تغییر دهید. - تاریخ و شماره صفحه در گزارش درج کنید. - تعداد رکوردها را در پاصفحه و لوگوی گزارش را در سرصفحه درج کنید. | | | | | | ✓ ✓ ✓ |



پودمان ۲

واحد یادگیری ۳ و ۴

مدیریت مجموعه داده

واحد یادگیری ۳

شایستگی کار با ساختار تکراری

مقدمات تدریسی

الف) مفاهیم کلیدی

| مفاهیم کلیدی | | | | |
|--------------|----------------|--------------|-----------------|-------|
| شمارنده حلقه | Trace | خروج از حلقه | حلقه‌های متداخل | حلقه |
| حلقه نامعین | خروجی‌های حلقه | حلقه معین | ورودی‌های حلقه | Break |

ب) تجهیزات لازم

مشخصات سخت‌افزاری برای نصب نرم‌افزار Visual Studio Express 2012:

- پردازنده حداقل ۱/۶ گیگاهرتز
- حافظه RAM حداقل ۱ گیگابایت
- حداقل فضای موجود در دیسک سخت ۴ گیگابایت
- کارت ویدئویی متناسب با DirectX 9 و قابل اجرا در رزولوشن 1024×768 و بالاتر
- مطمئن شوید که نرم‌افزار VS در تمام رایانه‌های موجود در کارگاه فعال (Active) است.

ج) بودجه‌بندی

برای بودجه‌بندی پودمان دوم می‌توانید از نمونه پیشنهادی استفاده کنید. برای انجام فعالیت‌های تکمیلی می‌توانید از کدهای نوشته شده موجود در «پرونده پیوست فعالیت‌های تکمیلی» استفاده کنید.

| جلسه | واحد یادگیری | کارگاه (موضوع) | شماره صفحات | اهداف توانمندسازی | فعالیت‌های تکمیلی |
|------|--------------|----------------|-------------|--|--|
| ۸ | ۳ | کارگاه ۱-۲-۳ | ۶۴-۶۵ | آشنایی با حلقه معین و استفاده از آن برای حل مسئله - بررسی شرایط حلقه معین - تغییر در مقادیر ابتدایی و انتهای و شرط اجرای حلقه - ایجاد حلقه‌های تأخیر در مسئله با استفاده از حلقه معین | از هم‌کلاسی خودتان بخواهید سه بار اسم خودش را بنویسد. |
| ۹ | ۳ | کارگاه ۴-۵ | ۶۶-۶۸ | کار با متغیر خارج بلاک - کار با دستورات داخل حلقه - ایجاد خروجی‌های مناسب در داخل و بیرون حلقه - کار با دستور break - ایجاد وقفه در روند اجرای حلقه تکرار | برنامه محاسبه مجموع تعداد خودکارهای قرمز و آبی دانش‌آموزان یک کلاس. |
| ۱۰ | ۳ | کارگاه ۶-۷ | ۷۰-۷۳ | تولید خروجی‌های خاص مانند اعداد زوج و فرد - کار با حلقه‌های نامعین - تبدیل حلقه‌های معین به نامعین - جست‌وجوی اعداد خاص | برنامه محاسبه تعداد دانش‌آموزان با رایانه‌های شماره فرد یا زوج |
| ۱۱ | ۳ | کارگاه ۸-۹ | ۷۵-۷۶ | کار با حلقه‌های متداخل - توسعه حلقه‌های متداخل - آشنایی با مفاهیم حلقه‌های متداخل - استفاده از این نوع حلقه‌ها در حل مسائل روزمره | برنامه جدول ضرب اعداد - برنامه نمایش یک ساعت دیجیتال - برنامه خودپرداز |
| | ۴ | کارگاه ۱ | ۸۲-۸۳ | آشنایی با مفهوم و کاربرد آرایه - شناخت انواع آرایه - تعریف یک آرایه - شناخت عناصر آرایه - مقداردهی به عناصر آرایه | برنامه‌ای بنویسید که حروف نام شما را به صورت تک‌تک در یک جدول نمایش دهد و حرف چهارم آن را تعیین کند. |
| ۱۲ | ۴ | کارگاه ۲-۳ | ۸۵-۸۷ | چگونگی دسترسی به مقادیر عناصر آرایه - دسترسی به یک عنصر خاص در آرایه - نمایش مقادیر ذخیره‌شده در آرایه - نمایش محتوای یک عنصر آرایه | برنامه‌ای بنویسید که حرف سوم اسم شما را نشان دهد. برنامه‌ای بنویسید که رقم‌های یک عدد چهاررقمی را تک‌تک ذخیره کند. |
| ۱۳ | ۴ | کارگاه ۴-۵ | ۸۸-۹۰ | جست‌وجوی مقدار یا مقادیر خاصی در خانه‌های آرایه ویرایش محتوای خانه‌های آرایه شناخت روش جست‌وجوی خطی - شناخت روش جست‌وجوی دودویی - شناخت مکانیسم Flag کدنویسی یکی از روش‌های جست‌وجوی خطی یا دودویی | برنامه‌ای بنویسید که با استفاده از روش جست‌وجوی خطی نام دانش‌آموزی را که شغل پدرش معلم (teacher) است بیابد. برنامه‌ای بنویسید که با استفاده از مفهوم Flag وجود یا عدم وجود حرف a را در نام یک دانش‌آموز بررسی کند. |
| ۱۴ | ۴ | کارگاه ۶-۷ | ۹۲-۹۵ | شناخت کلاس آرایه شناخت متدهای کلاس آرایه به‌کارگیری متدهای کلاس آرایه در برنامه | برنامه‌ای بنویسید که با استفاده از متدهای کلاس آرایه مقادیر ذخیره‌شده در آرایه را به صورت صعودی مرتب و مقدار دلخواه کاربر را در آن جست‌وجو کند. |

طرح درس روزانه (هفتگی) پیشنهادی

| کلاس: یازدهم | | درس: ساختارهای تکرار | |
|---|--|--|----------------|
| پیام جلسه (هدف کلی): هنرجو بتواند مسئله تکرار را تشخیص داده و آن را با حلقه حل کند. | | | |
| اهداف یادگیری | فعالیت‌ها | زمان فیزیکی | |
| فعالیت | کار هنرآموز | کار هنرجو | مدت (دقیقه) |
| طبقة هدف: حیطه عاطفی / شناختی / روانی - حرکتی | | | |
| ارزشیابی رفتار ورودی | نشان دادن انواع حلقه و مسائل حل شده با حلقه | مشارکت در پاسخگویی و تعامل | ۳۰ |
| ایجاد انگیزه | با استفاده از حلقه یک مسئله را حل کند. | یکروندنمای حلقه ترسیم کند. | ۳۰ |
| ارائه مفاهیم کلیدی (توضیح هنرآموز) | مفهوم حلقه‌ها را کامل توضیح داده و انواع حلقه‌ها را نام برده و مثال مناسب بنویسد. تفاوت حلقه‌ها را بیان کرده و نکات مربوط به حلقه‌ها را توضیح دهد. | هنرجویان مطالب پایه را می‌آموزند. | ۸۰ |
| فعالیت کارگاهی (تمرین هنرجویان) | هنرآموز توضیح کلی در مورد فعالیت کارگاهی به هنرجو داده و از آنها می‌خواهد به صورت گروهی به انجام آن بپردازند. | با توجه به فعالیت‌های کارگاهی کتاب هنرجویان در گروه‌های خود، گزینه‌های دیگر را با آزمون و خطا می‌آموزند. | ۵۰ |
| ارزیابی فعالیت‌ها ارائه تمرین | از هنرجویان خواسته می‌شود در گروه‌های تعیین شده به انجام فعالیت‌های کارگاهی بپردازند. قبل از شروع به کار هنرجویان، ملاک‌های ارزیابی تمرین‌ها به آنها داده شود. | هنرجویان در گروه‌های خود با نهایت دقت و سرعت با ایجاد خلاقیت به انجام فعالیت کارگاهی بپردازند. | ۳۰ |
| ارائه نکات تکمیلی (جمع‌بندی) | هنرجو باید مسئله را شناخته، راه حل مناسب را انتخاب، روندنمای آن را ترسیم و کد نویسی کند. | توجه، دقت، پرسش و پاسخ - انجام فعالیت‌های گروهی و ارائه راه‌حل‌های جدید برای مسائل. | ۴۰ |

| طرح درس روزانه (هفتگی) پیشنهادی | | | | |
|---|--|---|---|---|
| درس: ساختارهای تکرار | | کلاس: یازدهم | | |
| پیام جلسه (هدف کلی): هنرجو بتواند مسئله تکرار را تشخیص داده و آن را با حلقه حل کند. | | | | |
| زمان فیزیکی | فعالیت‌ها | | اهداف یادگیری | |
| ۴۰ | <p>هنرجویان در گروه‌های خود سعی می‌کنند در مدت‌زمان تعیین شده با نهایت دقت و سرعت با ایجاد خلاقیت به انجام پروژه بپردازند.</p> | <p>این ارزشیابی در دو شاخه انفرادی و گروهی انجام می‌شود: از هنرجویان خواسته می‌شود در گروه‌های تعیین شده به انجام پروژه این بخش از واحد کار بپردازند. نمره‌ای به کار گروهی هنرجویان داده می‌شود. آزمون برای هنرجویان به صورت انفرادی برگزار می‌شود. نمره گروهی در نمره انفرادی افراد تأثیرگذار است. نکته: قبل از شروع به کار، ملاک‌های ارزیابی پروژه به آنها داده می‌شود.</p> | <p>هنرجو باید بتواند یک مسئله را شناخته و حلقه مناسب برای حل آن انتخاب کرده، کدنویسی را عملی انجام دهد.</p> | <p>ارزشیابی شایستگی (ارزشیابی پایانی)</p> |
| <p>ابزارهای موردنیاز VS</p> | | | <p>ویدئو پروژکتور، رایانه، تخته آموزشی، برنامه VS</p> | |

د) ورود به بحث

قبل از تدریس این واحد بهتر است هنرجو را با مفاهیم «تکرار» و «حلقه» آشنا کنید. از دیرباز انجام فعالیت‌های مشابه، خسته‌کننده و وقت‌گیر بوده است. ساختارهای تکراری بخش جدانشدنی برنامه‌نویسی هستند که به برنامه‌نویسان کمک می‌کنند تا از انجام فعالیت‌های مشابه پرهیز کنند. در اطراف و زندگی روزمره ما کارهای تکراری بسیاری به چشم می‌خورد. کارمندی که هرروز صبح به محل کار خود می‌رود، هنرجویانی که هر هفته در مدرسه حاضر می‌گردند، روزهای هفته، روزهای سال و ... نمونه‌هایی از تکرارهای طبیعی زندگی ما هستند. به هنرجوی خود یاد دهید که در روند یادگیری این واحد مثال‌هایی از زندگی عادی و روزمره را که به هر شکلی دارای تکرار باشند، یادداشت‌برداری و در کلاس ارائه کند.

ایجاد انگیزه در هنرجویان

از هنرجویان خود بخواهید که پنج بار نام خودشان را بر روی کاغذ یادداشت کنند. حال از آنها بخواهید مراحل انجام این فعالیت را به صورت یک الگوریتم بنویسند. سپس روندنمای این مراحل را ترسیم کنند. برای تشریح بهتر مفهوم واحد یادگیری به گروه اول بگویید این کار را ۱۰ بار و به گروه دوم بگویید این کار را ۲۰ بار و همین‌طور الی آخر انجام دهند. از آنها بخواهید بنویسند هر گروه چند بار کار تکراری کرده‌اند؟ به هر گروه چند خط الگوریتم و چند شکل اضافه شده است؟

به‌منظور تعیین سطح دانش‌آموزان می‌توانید سؤالاتی از این دست مطرح کنید:

سؤال ۱: به نظر شما در محیط مدرسه چه کارهای تکراری انجام می‌شود؟

سؤال ۲: کدام یک از شما می‌تواند یک مثال برای کار تکراری بزند؟ سپس بخواهید تا مراحل تکرار را مشخص کنند و تعداد آنها را بشمارند. از آنها سؤال کنید آیا ابتدا و انتهای این کارها مشخص است یا نامشخص؟

سؤال ۳: چرا به تکرار در کارهایمان احتیاج داریم؟

سؤال ۴: آیا تا به حال در صف نانواپی ایستاده‌اید؟

سؤال ۵: آیا تا به حال از جدول ضرب استفاده کرده‌اید؟

سؤال ۶: آیا می‌توانید اعداد ۱۰۰ تا ۳۰۰ را سه تا سه تا بشمارید.

سؤال ۷: فرض کنید در یک کارخانه سازنده ماشین برای هر ماشین چهار حلقه تاپیر گذاشته می‌شود. آیا این یک کار تکراری است؟

سؤال ۸: از شما خواسته می‌شود تا صفحاتی از یک کتاب را بخوانید. شما ممکن است اصلاً صفحه‌ای را مطالعه نکنید یا به تعداد دلخواه، صفحاتی از کتاب را مطالعه کنید.

سؤال ۹: آیا تا به حال با دوستانتان بازی گل یا پوچ را انجام داده‌اید؟ آیا به جدول لیگ برتر فوتبال ایران نگاه کرده‌اید؟ تعداد ست‌های یک مسابقه والیبال معین است یا نامعین؟

سؤال ۱۰: به ساعت کارگاه نگاه کنید و تکرارها را تشخیص دهید.

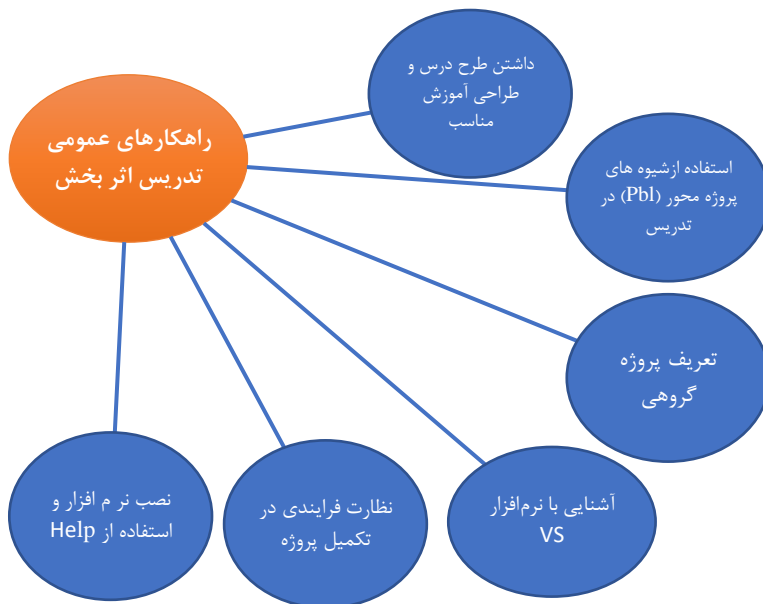
سؤال ۱۱: چند نمونه از کاربردهای تکرار را نام ببرید.

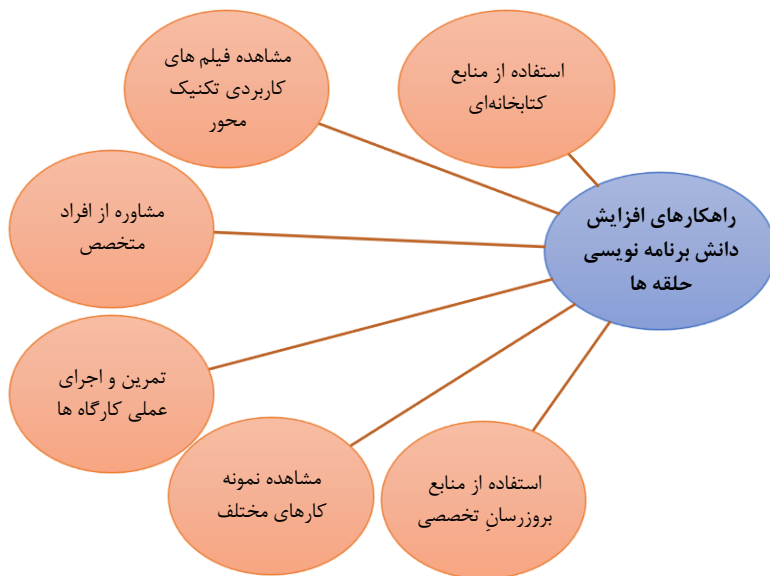
سؤال ۱۲: ۱۰۰ بار واژه سلام را بر روی دفتر بنویسید.

سؤال ۱۳: ۱۰ نماد روی صفحه رومیزی و پندوز خود را در یک گوشه جمع کنید.

سؤال ۱۴: ۱۰۰ نقطه برای بازی نقطه - خط بر روی کاغذ خود بگذارید.

هنگامی که با کارهای مشابه برخورد می‌کنید به این فکر خواهید کرد که چطور می‌شود کار را یک بار انجام داد سپس به تعداد مناسب آن را تکرار کرد. تکرارهای طبیعی زندگی به صورت تکرارهای مشخص و یا غیرقابل پیش‌بینی رخ می‌دهند.





تدریسی

حلقه‌ها

حلقه تعداد تکرار یک کار است که در دو نوع معین و نامعین تعریف شده است. وقتی سه بار سلام می‌کنید از یک حلقه معین استفاده کرده‌اید. یا زمانی که منتظر نتیجه یک مسابقه والیبال هستید یک حلقه نامعین را دنبال کرده‌اید. وقتی به ثانوی می‌روید و در صف قرار می‌گیرید که نوبت شما رسیده و نان را سفارش دهید، در واقع از دو نمونه حلقه نامعین و معین استفاده کرده‌اید.

هر حلقه for دارای چهار بخش است: متغیر و مقداردهی اولیه حلقه، عبارت منطقی، مقدار گام حلقه، دستورات تکرارشونده.

شکل دستوری برنامه‌نویسی حلقه for به صورت زیر است.

```

for ( تغییر مقدار متغیر : عبارت منطقی : مقدار اولیه - نام متغیر )
{
    دستوراتی که در داخل حلقه باید تکرار شود
}
  
```

در حلقه for می‌توان از اعداد صحیح و اعشاری، کاراکتر و عبارت در قسمت‌های مقدار اولیه، گام حلقه و مقدار نهایی استفاده کرد. حلقه‌های معین را می‌توانیم به دو صورت افزایشی یا کاهشی بنویسیم.

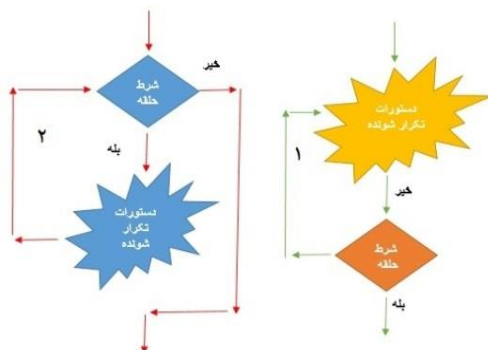
- مقدار اولیه کوچک‌تر از مقدار نهایی
- گام حلقه مثبت

- مقدار اولیه بزرگ‌تر از مقدار نهایی
- گام حلقه منفی

حلقه افزایشی

حلقه کاهشی

حلقه نامعین حلقه‌ای است که تعداد تکرار آن معین و مشخص نیست. به‌طور کلی دو نمونه حلقه نامعین را در شکل‌های زیر می‌بینید.



شکل ۱ حلقه‌ای است که ابتدا دستورات تکرار شونده داخل حلقه انجام می‌شوند و سپس شرط حلقه بررسی می‌شود. تعداد تکرار این حلقه، یک‌بار یا بیشتر از یک‌بار است.

شکل شماره ۲ حلقه‌ای است که ابتدا شرط حلقه بررسی شده و سپس در صورتی که جواب شرط درست باشد دستورات داخل حلقه اجرا می‌شوند و در صورتی که شرط درست نباشد حلقه به پایان می‌رسد. تعداد تکرار این حلقه، صفر بار یا بیشتر است.

مشکلات متداول در فرآیند یاددهی - یادگیری

این واحد یادگیری به دلیل تأثیر واضح یا پنهان در سایر واحدهای کتاب از اهمیت فوق‌العاده‌ای برخوردار است. شاید بزرگ‌ترین دغدغه هنرجویان در فراگیری این درس این باشد که کجا از حلقه معین و کجا از حلقه نامعین استفاده کنند؟ درست است که تشخیص استفاده مناسب از حلقه‌ها با تمرین به دست می‌آید ولی نشانی‌هایی مانند میان، بار، مرتبه، تا، بین و از این دست کلمات در برنامه‌ها هنرجو را به استفاده از حلقه معین و کلماتی شبیه تا زمانی که، تا وقتی که، تا هنگامی که و ... که معنای انتظار نامشخص را می‌دهند ایشان را به استفاده از حلقه نامعین هدایت می‌کنند.

شیوه و الگوی پیشنهادی

برنامه‌نویسی بدون داشتن شیوه و الگوی خاص تقریباً غیرممکن است. سعی کنید پیش‌زمینه‌های مناسب را برای فراگرفتن این واحد یادگیری در هنرجو ایجاد کنید. پیشنهاد می‌شود به هنرجویان تکالیف گروهی و البته مختلف داده شود و همچنین ذهن آنها را با مسئله‌های تکرار روزمره درگیر کنید. همچنین پیشنهاد می‌شود ابتدا به هنرجویان روش طراحی و پیاده‌سازی روی کاغذ را آموزش داده، سپس اجازه دسترسی به سیستم داده شود.

پاسخ به فعالیت‌ها

فعالیت گروهی
ص ۵۹

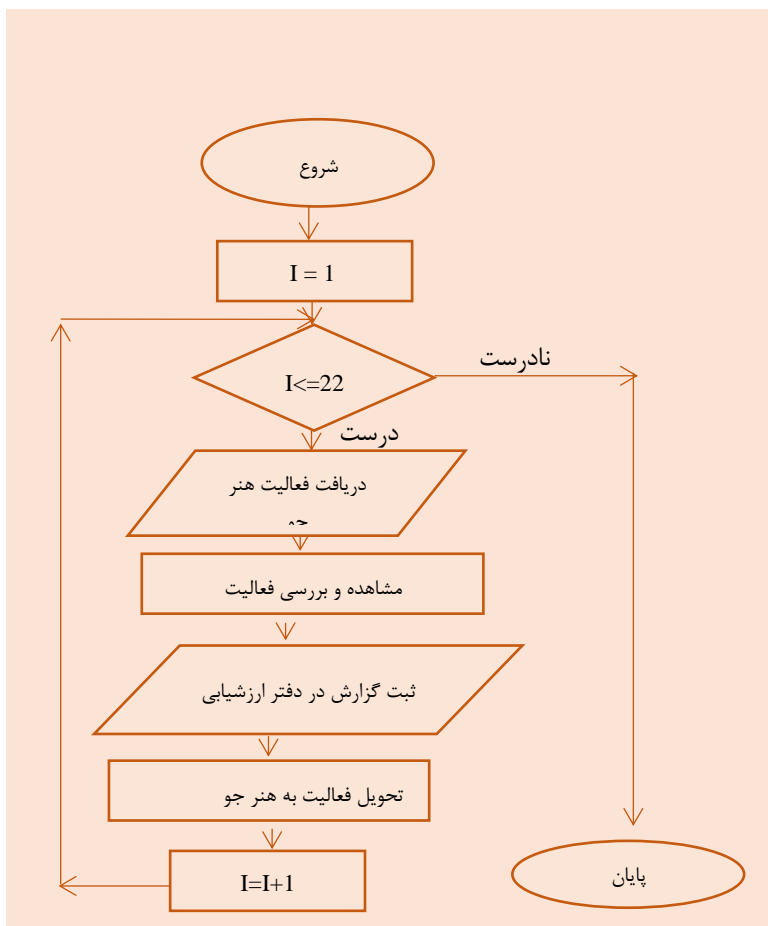
در جدول زیر مشخص کنید کدام‌یک از آنها معین یا نامعین هستند.

| نوع حلقه | مثال |
|----------|---|
| معین | یک نجار روزانه ۱۲ صندلی می‌سازد. |
| نامعین | تا زمانی که دبیر ورزش سوت پایان را بزند همه هنرجویان چندین نرمش کششی انجام می‌دهند. |
| معین | در یک نانوايي بربری، نانوا در هر تنور ۶۰ عدد نان را می‌پزد. |
| نامعین | یک نقاش ساختمان در یک روز تعدادی دیوار با ابعاد مختلف را رنگ می‌زند. |

مثال
ص ۵۹

یک هنرآموز، فعالیت منزل هنرجویان را تحویل گرفته، پس از بررسی، گزارش هر کدام را در دفتر ارزشیابی خود وارد می‌کند.

| | |
|-----------|--|
| ورودی‌ها | فعالیت‌های انجام‌شده هنرجویان |
| خروجی‌ها | ثبت گزارش هر هنرجو در دفتر ارزشیابی |
| بدنه حلقه | دریافت فعالیت، بررسی فعالیت، ثبت گزارش برای هنرجو، تحویل فعالیت به هنرجو |



می‌خواهیم سیستم نوبت‌دهی یک مطب دندان پزشکی را مکانیزه کنیم، به طوری که برای نوبت‌دهی به ۲۰ بیمار خود از شماره‌های چاپ‌شده ۱ تا ۲۰ استفاده کند.

مسئله
ص ۶۱

پاسخ :

- (۱) شروع
- (۲) مقدار i را برابر یک قرار بده
- (۳) اگر $i \leq 20$ برو به مرحله بعد وگرنه برو به مرحله ۶
- (۴) مقدار i را نمایش بده.
- (۵) i را یک واحد اضافه کن و برو به مرحله ۲
- (۶) پایان

فعالیت کارگاهی
ص ۶۱

- اگر تعداد بیماران ۳۰ نفر باشد، چه تغییری لازم است؟
خط دوم: ۲- اگر $i \leq 30$ برو به مرحله بعد وگرنه برو به مرحله ۶
- روندنما و الگوریتم را تغییر دهید تا تعداد بیماران را از ورودی دریافت کند.

- (۱) شروع
- (۲) مقدار total را دریافت کن.
- (۳) مقدار counter را برابر یک قرار بده.
- (۴) اگر $counter \leq total$ برو به مرحله ۴، در غیر این صورت برو به مرحله ۷
- (۵) مقدار counter را نمایش بده.
- (۶) counter را یک واحد اضافه کن
- (۷) برو به مرحله ۳
- (۸) پایان

مسئله
ص ۶۱

شرط قبولی در یک آزمون کسب حداقل نمره ۱۲ از ۲۰ است. می‌خواهیم نمرات
هنرجویان یک کلاس ۲۰ نفره را دریافت کرده، تعداد قبول شده‌ها را مشخص کنیم.
پاسخ:

- (۱) شروع
- (۲) مقدار i را برابر یک قرار بده
- (۳) مقدار count را برابر صفر قرار بده.
- (۴) اگر $i \leq 20$ برو به مرحله بعد، در غیر این صورت برو به مرحله ۹
- (۵) نمره آزمون هنرجوی i ام را دریافت کن و در متغیر mark ذخیره کن
- (۶) اگر $mark \geq 12$ است یک واحد به count اضافه کن
- (۷) i را یک واحد اضافه کن و برو به مرحله ۴
- (۸) مقدار متغیر count را نمایش بده
- (۹) پایان

کارگاه ۱ – تبدیل روندنما به برنامه

تکمیل کارگاه
ص ۶۴

- برنامه را طوری تغییر دهید که برای یک درمانگاه خیریه با تعداد بیماران بیشتر
قابل استفاده باشد. شماره‌های موردنیاز از ۱۰۰ تا ۹۹۹ است.
پاسخ:

```
for (int i = 100; i <= 999; i++)  
    Console.WriteLine(i);
```

- برنامه را طوری تغییر دهید که حداکثر تعداد بیماران را از ورودی دریافت کند.
پاسخ:

```
int total = int.Parse(Console.ReadLine());  
for (int counter= 1 ; counter <= total ; counter++)  
    Console.WriteLine(counter);
```

- پس از پرانتز دستور for علامت ; قرار داده خروجی را بررسی کنید.
پاسخ: در صورتی که در انتهای دستور حلقه معین for از علامت ; استفاده کنیم، بدنه حلقه اجرا نخواهد شد ولی متغیر افزایش یا کاهش خواهد یافت.
- بررسی کنید در صورتی که آکولادهای قطعه کد بالا حذف شوند خروجی چه تغییری می‌کند؟
پاسخ:
فقط یک دستور و آن هم نزدیک‌ترین دستور به for اجرا خواهد شد.

```
int i;  
for (i = 2; i <= 30; i++)  
    i++;  
Console.WriteLine(i);
```

در این قطعه کد تنها دستور i++ تکرار خواهد شد و حلقه گردش خواهد کرد و در انتهای گردش حلقه دستور نمایش متغیر i، آخرین مقدار i یعنی مقدار ۳۲ نمایش داده خواهد شد.
- قطعه کد مرحله ۲ را طوری تغییر دهید تا شماره هنجویانی که باید کارگاه را نظافت کنند چاپ کند.

```
for (int i = 1; i <= 30; i += 2)  
    Console.WriteLine("{0,5}", i);
```

- در عبارت منطقی، علامت >= را به <= تبدیل کنید، خروجی چه تغییری می‌کند؟
پاسخ: برنامه خطای منطقی دارد و خروجی نمایش داده نمی‌شود، زیرا شرط قسمت دوم حلقه برقرار نیست.

فعالیت گروهی
ص ۶۵

خروجی قطعه کدهای زیر را با کمک هم گروهی خود و بدون استفاده از رایانه بنویسید، سپس آن را اجرا و نتیجه را مقایسه کنید.
پاسخ:

| خروجی برنامه پس از اجرا | خروجی برنامه از نظر شما | حلقه |
|---|-------------------------|---|
| چاپ ۵۴۳۲۱ | | <pre>for (int i=5;i>=1;i--) Console.Write(i);</pre> |
| چاپ ۰-۳-۶-۹ زیر هم | | <pre>for (int i=0;i<10;i+=3) Console.WriteLine(i);</pre> |
| چاپ اعداد زوج نزولی از ۱۰ تا ۰ به صورت زیر هم | | <pre>for (inti=10;i>=0;i-=2) Console.WriteLine(i);</pre> |

کنجکاوی
ص ۶۵

آیا متغیر حلقه می‌تواند از نوع float, byte یا char باشد؟
پاسخ: بله، به‌طور کلی هر نوع داده‌ای که ترتیبی باشد می‌تواند به‌عنوان متغیر حلقه مورد استفاده قرار بگیرد. به این مثال‌ها توجه کنید.

```
for (float i = 1.5f ; i <= 3.75 ; i=i+0.25f)  
Console.WriteLine(i);
```

```
for (char i = 'a' ; i <= 'g' ; i++)  
Console.WriteLine(i);
```

فعالیت منزل
ص ۶۵

هنرآموزی می‌خواهد ده نفر از هنرجویان یک کلاس را به گروه‌های ۲ نفره تقسیم کند، او قصد دارد در هر گروه به ترتیب یک نفر از اول فهرست و یک نفر از آخر فهرست را قرار دهد. برنامه‌ای بنویسید که شماره هنرجویان هر گروه را در خروجی نمایش دهد.
پاسخ:

```
int j=10;  
for (int i = 1; i <= 5; i++)  
{  
Console.WriteLine("{0} --> {1}",i,j);  
j--;  
}
```

کارگاه ۳ – کار با متغیر خارج بلاک

پاسخ به فعالیت‌ها

دستور زیر را به انتهای برنامه اضافه کنید تا مقدار نهایی متغیر i پس از اتمام حلقه چاپ شود. دلیل خطای زیر چیست؟ به کمک هنرآموز خود راه‌حل آن را پیدا کنید.

تکمیل کارگاه
ص ۶۵

پاسخ:

چون متغیر i در حلقه تعریف و مقداردهی شده ولی در خارج از حلقه استفاده شده باعث بروز خطا شده است. برای حل این مشکل تعریف متغیر i را در ابتدای برنامه انجام داده، برنامه را بدون خطا اجرا کنید.

کارگاه ۴ – محاسبه مجموع

پاسخ به فعالیت‌ها

- روندنمای مسئله را کامل کنید.

- (۱) شروع
- (۲) $i=1$
- (۳) $sum=0$
- (۴) اگر $i < 8$ بود برو به مرحله ۵ و در غیر این صورت برو به مرحله ۸
- (۵) مقدار $minute$ را دریافت کن
- (۶) $Sum=sum+minute$
- (۷) $i=i+1$ و به مرحله ۴ برگرد
- (۸) مقدار sum را نمایش بده
- (۹) پایان

تکمیل کارگاه
ص ۶۶

- قطعه کد زیر را در متد $Main()$ وارد کرده برنامه را کامل کنید.

```
int sum;
int minute;
sum = 0;
for (int i = 1; i < 8; i++)
{
    Console.WriteLine("please rate the day {0} enter in minute : ", i);
    minute = int.Parse(Console.ReadLine());
    sum = sum + minute;
}
Console.WriteLine("you studied for {0} minutes a week.", sum);
Console.ReadKey();
```

- برنامه را طوری تغییر دهید که مجموع ساعات مطالعه ماهانه هنرجو را نمایش دهد.

```
int sum;
int minute;
sum = 0;
for (int i = 1; i < 31; i++)
{
    Console.WriteLine("please rate the day {0} enter in minute :
", i);
    minute = int.Parse(Console.ReadLine());
    sum = sum + minute;
}
Console.WriteLine("you studied for {0} minutes a month.
",sum);
Console.ReadKey();
```

- برنامه را توسعه دهید تا میانگین ساعت مطالعه ماهانه هنرجو را نمایش دهد.

```
int sum;
int minute;
sum = 0;
for (int i = 1; i < 31 ; i++)
{
    Console.WriteLine("please rate the day {0} enter in minute :
", i);
    minute = int.Parse(Console.ReadLine());
    sum = sum + minute;
}
Console.WriteLine("you studied for {0} minutes a month.
",(float)sum/30);
Console.ReadKey();
```

کارگاه ۵ - خروج زودرس از حلقه

در هنگام برنامه نویسی، مواقعی که نیاز دارید برنامه زودتر از موعد از بلاک خارج شود، می‌توانید از دستور break برای خروج زودرس از حلقه استفاده کنید.

```
for (int i = 10; i <= 100; ++i)
{
    if (i % 10 == 0)
    {
        break;
    }
}
```

پاسخ به فعالیت‌ها

تکمیل کارگاه

ص ۶۷

- در کد زیر متغیرهای موردنیاز برنامه را اعلان کرده، دستورات دریافت آنها از ورودی را تکمیل کنید.

```
int guess;
int month;
int score = 5;
Console.Write("Enter number your month:");
month = int.Parse(Console.ReadLine());
Console.Clear();
for (int i = 1; i <= 5; i++)
{
    Console.Write("your guess? ");
    guess = int.Parse(Console.ReadLine());
    if (guess == month)
    {
        Console.ForegroundColor = ConsoleColor.Green;
        Console.WriteLine("you win ");
        Console.WriteLine("your score: {0} ", score);
        break;
    }
    else
    {
        score -= 1;
        Console.WriteLine("try again ! ");
    }
}
```

مقدار month را یازده وارد کنید. (month = 11)

| i | Guess | score | خروجی |
|---|-------|-------|-------------|
| 1 | 3 | 4 | Try again ! |
| 2 | 8 | 3 | Try again ! |
| 3 | 2 | 2 | Try again ! |
| 4 | 4 | 1 | Try again ! |
| 5 | 5 | 0 | Try again! |

- برنامه را طوری تغییر دهید که پس از ۵ بار حدس اشتباه کاربر دوم، پیام مناسب دیگری چاپ شده، شماره ماه تولد را نمایش دهد.

پاسخ:

```
Console.Clear();
for (int i = 1; i <= 5; i++)
{
    Console.Write("your guess? ");
    guess = byte.Parse(Console.ReadLine());
    if (guess == month)
    {
        Console.ForegroundColor = ConsoleColor.Green;
        Console.WriteLine("you win ");
        Console.WriteLine("your score: {0} ", score);
        break;
    }
    else
    {
        score -= 1;
        if (score == 0)
        {
            Console.WriteLine("you lose");
            Console.WriteLine(" month is: " + month);
            break;
        }
        Console.WriteLine("try again ! ");
    }
}
```

- این برنامه را می‌توان به صورت زیر هم نوشت. این برنامه تعداد حدس‌های زده شده پس از موفق شدن کاربر دوم، را با پیام مناسب چاپ می‌کند.

```
int c = 0;
Console.Write("your month? ");
string month = Console.ReadLine();
Console.Clear();
Console.Write("your guess? ");
string guess = Console.ReadLine();
for ( ; guess != month; guess = Console.ReadLine())
{
    Console.WriteLine("type your guess");
    c++;
}
Console.WriteLine("you win , month is " + month);
Console.WriteLine(c + " suspect after you win");
```

جدول را به کمک هم گروهی خود کامل کنید.

| خروجی برنامه پس از اجرا | قطعه کد |
|---|--|
| ۱۰ ۱۲ ۱۴ ابتدا متغیر k با مقدار ۱۰ پر می شود و چون کمتر از ۱۵ است نمایش داده می شود. سپس یک واحد اضافه شده (۱۱) و در برگشت حلقه به سمت بالا یک واحد دیگر نیز به آن اضافه می شود (۱۲). بنابراین در هر بار گردش ۲ واحد به این متغیر اضافه شده است. | <pre>for (int k = 10; k <= 15; k++) { Console.WriteLine(k); k++;}</pre> |
| بی نهایت Iran چاپ می شود. این حلقه به خاطر نداشتن شمارنده در واقع به صورت یک حلقه متناوب و بی نهایت فعالیت می کند. از این حلقه در مسائلی که احتیاج به تکرار بدون پایان داریم استفاده می کنیم. | <pre>for (; ;) { Console.ForegroundColor = ConsoleColor.Cyan; Console.WriteLine("Iran");}</pre> |
| ۵ بار Iran زیر هم به رنگ فیروزه ای چاپ می شود. ابتدا متغیر i با مقدار ۱ پر می شود و چون کمتر از ۱۰ است یک بار واژه Iran نمایش داده می شود. سپس یک واحد اضافه شده (۲) و در برگشت حلقه به سمت بالا یک واحد دیگر نیز به آن اضافه می شود (۳) بنابراین در هر بار گردش ۲ واحد به این متغیر اضافه شده در واقع متغیر i مقادیر ۱ و ۳ و ۵ و ۷ و ۹ را در خود جای می دهد که به تعداد آنها (۵ بار) واژه Iran نمایش داده می شود. | <pre>int i = 1; for (; i<=10; i++) { Console.ForegroundColor = ConsoleColor.Cyan; Console.WriteLine("Iran"); i++;}</pre> |

برای قطعه کد زیر پس از اجرا و بررسی نتیجه، یک مسئله کاربردی بنویسید.
پاسخ: هنرجو می تواند یک یا چند مسئله برای این برنامه مثال بزند. مانند:
- نمایش مضارب عدد سه کوچکتر از ۱۰۰
- بازی هپ، برای مضارب ۳ طراحی شده است.

هنرستان دخترانه پروین اعتصامی هر سال به مناسبت میلاد حضرت معصومه (س) و روز دختر به هنرجویان، که نام آنها معصومه است هدیه می‌دهد. برنامه‌ای طراحی کنید که با دریافت نام هنرجویان یک هنرستان تعداد هنرجویانی که هم نام حضرت معصومه هستند را نمایش دهد. تعداد کل هنرجویان هنرستان در ابتدای برنامه از کاربر دریافت شود.

پاسخ:

```
string name;
byte counter = 0;
byte totalStudent;
Console.WriteLine("number of students");
totalStudent = byte.Parse(Console.ReadLine());
Console.Clear();
Console.WriteLine("name of student");
for (int i = 1; i <= totalStudent; ++i)
{
    name = Console.ReadLine();
    if (name == "masoomeh")
        ++ counter;
    Console.WriteLine("tedad ==> {0}", counter);
}
```

کارگاه ۶ – به کارگیری حلقه while

- قطعه کد زیر را در متد Main() وارد کنید و برنامه را کامل کنید.
ابتدا تعریف متغیرها:

```
float sum=0,avg;
int count=0;
Console.Write("Enter mark {0} =", count+1);
float mark=float.Parse ( Console.ReadLine());
while(mark >= 0)
{
    sum = sum + mark;
    count++;
    Console.Write("Enter mark {0} =", count+1);
    mark = float.Parse(Console.ReadLine());
}
```

متغیر ccount برای شمارش تعداد هنرجویانی که نمره ادبیات آنها وارد شده به کار گرفته شده است. (شمارش تعداد هنرجویان)

- دستورات لازم برای محاسبه و نمایش میانگین را به برنامه اضافه کنید.

پاسخ:

```
avg = sum / count;  
Console.WriteLine("avg = {0} ", avg);
```

فعالیت کارگاهی

ص ۷۱

- قطعه کدهای زیر را Trace کرده، اولین و آخرین مقدار چاپ شده را در جدول زیر بنویسید.

| آخرین مقدار در خروجی | اولین مقدار در خروجی | دستور |
|--|----------------------|--|
| . | ۹۸ | int i = 99; while (i >= 1) Console.WriteLine(--i); |
| ابتدا متغیر i با ۹۹ مقداردهی شده است، شرط برقرار است اما هنگام نمایش مقدار i یکی از آن کم می‌شود و عدد ۹۸ نمایش داده می‌شود. آخر حلقه نیز مقدار i یک می‌شود و به همان علت عدد ۰ نمایش داده می‌شود. | | |
| ۱۰۰۱ | ۱۰۱ | int i = 100; while (i <= 1000) Console.WriteLine(++i); |
| ابتدا مقدار متغیر i ۱۰۰ می‌شود و چون کمتر از ۱۰۰۰ است حلقه اجرا شده و ابتدا مقدار ۱۰۱ نمایش داده می‌شود و در آخر حلقه که مقدار ۱۰۰۰ می‌آید یکی اضافه شده و ۱۰۰۱ به نمایش در خواهد آمد. | | |

کارگاه ۷ – یافتن بزرگ‌ترین مقدار

پاسخ به فعالیت‌ها

- متغیرهای استفاده‌شده در قطعه کد زیر را اعلان کرده، قطعه کد زیر را تکمیل و برنامه را اجرا کنید.

پاسخ:

```
int hour, max;  
Console.ForegroundColor = ConsoleColor.Blue;  
Console.WriteLine("\nEnter 0 for Exit");  
Console.Write("enter hour: ");  
hour = int.Parse(Console.ReadLine());
```

تکمیل کارگاه

ص ۷۳

```

max = hour;
while (hour > 0)
{
if (hour > max)
max = hour;
Console.WriteLine("\nEnter 0 for Exit");
Console.Write("enter hour: ");
hour = int.Parse(Console.ReadLine());
}
Console.WriteLine("max= {0}", max);

```

شرط داخل حلقه while بزرگتر از صفر است و با وارد کردن یکی از اعداد منفی یا صفر شرط خروج از حلقه برقرار و از برنامه خارج می‌شود.

– برنامه را طوری تغییر دهید که کمترین و بیشترین ساعت کاری را نمایش بدهد.

پاسخ:

برای به دست آوردن مقدار بزرگتر و کوچکتر از میان تعدادی عدد راه‌حل ارائه شده این است که مقدار اول را هم بزرگتر و هم کوچکتر در نظر بگیریم و باقیمانده مقادیر را با آنها مقایسه کنیم. بنابراین در کد نوشته شده مقدار اولین ساعت خوانده شده و به عنوان بزرگترین و کوچکترین مقدار در متغیرهای min و max ذخیره شده است و پس از آن تا زمانی که مقدار وارد شده بزرگتر از صفر است، مقادیر خوانده شده با همین دو متغیر مقایسه شده و در صورت برقراری شروط نوشته شده مقادیر این متغیرها تغییر می‌کند.

```

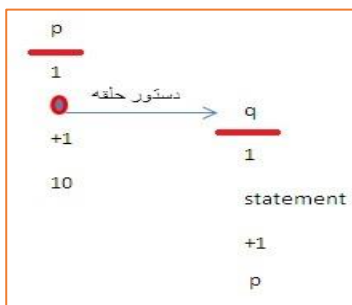
int hour, min, max;
Console.ForegroundColor = ConsoleColor.Blue;
Console.WriteLine("\nEnter 0 for Exit");
Console.Write("enter hour: ");
hour = int.Parse(Console.ReadLine());
max = hour;
min = hour;
while (hour > 0)
{
if (hour > max)
max = hour;
else if (hour < min)
min = hour;
Console.WriteLine("\nEnter 0 for Exit");
Console.Write("enter hour: ");
hour = int.Parse(Console.ReadLine());
}
Console.WriteLine("max= {0}", max);
Console.WriteLine("min= {0}", min);

```

کارگاه ۸ – کاربرد حلقه‌های متداخل

در صورت استفاده از یک حلقه بعنوان دستور حلقه دیگر، حلقه تودرتو ایجاد خواهد شد. به تکه برنامه زیر نگاه کنید.

```
for (int i = 0; i <= 10; ++i)
    for (int j = 0; j <= 10; ++j)
        Console.WriteLine("*");
```



در برنامه بالا در بدنه حلقه for یک حلقه دیگر وجود دارد. البته چنین تعریفی در خصوص حلقه نامعین while نیز صادق است. شکل روبه‌رو مفهوم حلقه تودرتو را برای حلقه معین for نشان می‌دهد. در این شکل می‌بینید که به جای دستور حلقه p یک حلقه دیگر یعنی حلقه q اجرا می‌شود.

به‌طور کلی حلقه‌های تودرتو (بدون توجه به معین بودن یا نامعین بودن) را می‌توان به دو دسته کلی تقسیم کرد:

(۱) حلقه‌های تودرتوی مستقل

تمام قسمت‌های حلقه (مقدار اولیه - مقدار نهایی - گام حلقه - دستور حلقه) مستقل، مشخص و معلوم هستند.

```
for(int i=0;i<=10;++i)
    for(int j=0;j<=10;++j)
        Console.WriteLine("*");
```

در این نوع حلقه‌ها به‌راحتی می‌توان تعداد تکرار دستورات را محاسبه کرد.

۱ + مقدار اولیه - مقدار نهایی = تعداد تکرار حلقه

کافی است از روش محاسبه تعداد تکرار حلقه‌های مستقل تکی تعداد تکرار حلقه‌ها را به صورت مجزا حساب کرده سپس عددهای به‌دست‌آمده را در یکدیگر ضرب کنیم.

تعداد کل تکرارها = تعداد تکرار حلقه اول * تعداد تکرار حلقه دوم * * *

(۲) حلقه‌های تودرتوی غیرمستقل (وابسته)

در این نوع حلقه‌ها، حلقه‌های داخلی به حلقه‌های بیرونی وابسته هستند در واقع یکی از قسمت‌های حلقه معلوم و مشخص نیست.

```
for (int p = 0; p <= 10; ++p)
    for (int q = 0; q <= p; ++q)
        Console.WriteLine("*");
```

مشاهده می‌کنیم که حلقه دوم یعنی حلقه q در بخش نهایی به حلقه p وابسته است. تأثیری که این وابستگی دارد این است که دیگر نمی‌توان مقدار گردش حلقه دوم را به صورت صحیح و با استفاده از فرمول محاسبه کرد.

```
for (int p=1 ; p<=5 ; ++p)
    for(int q=1 ; q<=p;++q)
        Console.WriteLine("{0}",q);
```

بنابراین برای محاسبه تعداد گردش یا به دست آوردن خروجی این‌گونه حلقه‌ها باید از جدول درستی که قبلاً توضیح دادیم بهره ببریم.
مثال: خروجی تکه برنامه زیر چیست؟

```
for (int p = 1; p <= 5; ++p)
{
    for (int q = 1; q <= p; ++q)
    {
        Console.Write("{0}", q);
    }
    Console.WriteLine();
}
```

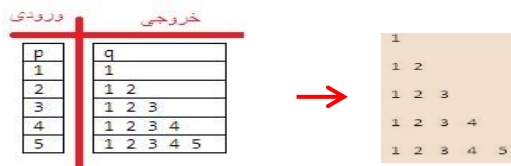
برای به دست آوردن خروجی این برنامه یک جدول درستی با همان توضیحات قبلی ترسیم می‌کنیم. می‌بینید که تعداد گردش حلقه دوم یعنی حلقه q به حلقه p وابسته شده است. برای مثال هنگامی که p مقدار ۳ را دریافت کرده حلقه q نیز تا شماره ۳ حرکت کرده است. در شکل خروجی نهایی این تکه برنامه را مشاهده می‌کنیم.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۷۵

برای تنظیم نمایش جدول ضرب به جای جانگهدار از متد `SetCursorPosition` استفاده کنید.

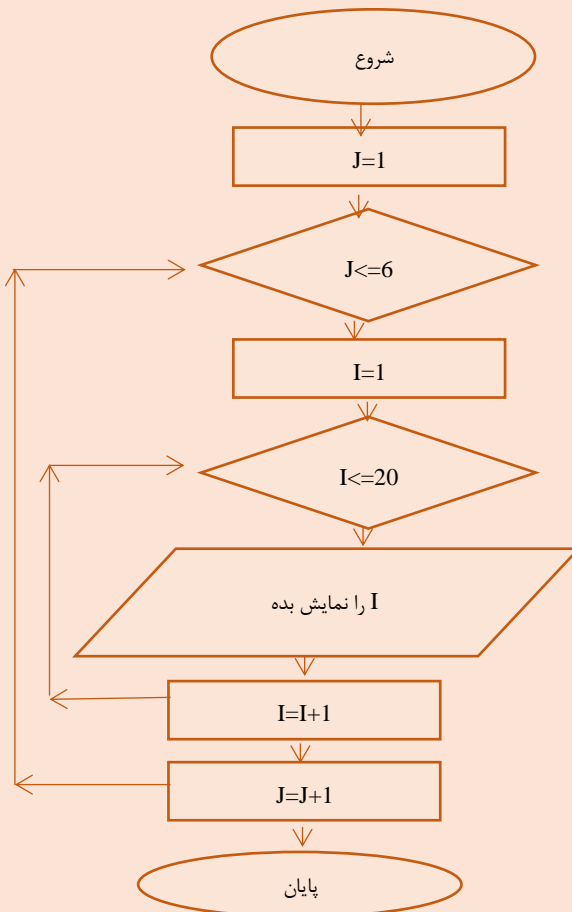
```
Console.ForegroundColor = ConsoleColor.Green;
for (int i = 1; i <= 5; i++)
{
    Console.SetCursorPosition(3, i + 3);
    for (int j = 1; j <= 5; j++)
```



```
Console.Write("{0,4}", i * j);
Console.WriteLine();}
```

سیستم نوبت‌دهی مطب دندان‌پزشکی را طوری تغییر دهید که برای یک هفته که شامل شش روز کاری است نوبت‌دهی انجام دهد.

```
for (int j = 1; j <= 6; j++)
for (int i = 1; i <= 20; i++)
    Console.WriteLine(i);
```



کارگاه ۹ – توسعه حلقه متداخل

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۷۵

- جدول روبه‌رو را تکمیل کنید. برنامه را اجرا کنید آیا با خروجی‌های جدول شما مطابقت دارد؟

| تعداد اسکناس‌های ۲ هزار تومانی | تعداد اسکناس‌های ۵ هزار تومانی |
|--------------------------------|--------------------------------|
| ۰ | ۱۰ |
| ۵ | ۸ |
| ۱۰ | ۶ |
| ۱۵ | ۴ |
| ۲۰ | ۲ |
| ۲۵ | ۰ |

پاسخ:

بله- دقیقاً همان خروجی‌هایی را تولید خواهد کرد که در جدول به دست آورده بودیم. البته خودپرداز بر اساس موجودی اسکناس‌ها پرداخت را انجام خواهد داد. اما در اینجا اولین مقادیر X و Y به دست آمده را استخراج خواهد کرد یعنی همان ردیف اول جدولی که شما پر کرده‌اید.

مقدار نهایی X مقدار ۲۵ در نظر گرفته شده است، زیرا بر اساس رابطه $۲۵ * ۲۰۰۰ = ۵۰۰۰۰$ خواهد شد و مقدار نهایی Y مقدار ۱۰ در نظر گرفته شده است، زیرا بر اساس رابطه $۵۰۰۰۰ = ۵۰۰۰ * ۱۰$ خواهد شد.

- برنامه چند حالت را با مقادیر مختلف (X, Y) آزمایش می‌کند تا به جواب برسد؟

پاسخ:

برای به دست آوردن جواب این سؤال ابتدا باید تعداد تکرار حلقه X و سپس تعداد تکرار حلقه Y را به دست آورده و سپس این مقادیر را در یکدیگر ضرب کنیم.

$$X \text{ تعداد تکرار حلقه} = ۱ + ۱ - ۲۵ = ۲۵$$

$$Y \text{ تعداد تکرار حلقه} = ۱ + ۱ - ۱۰ = ۱۰$$

و در نهایت تعداد کل تکرارها برابر است با $۲۵ * ۱۰ = ۲۵۰$

- برنامه را با فرض اینکه خودپرداز اسکناس ۱۰ هزار تومانی هم دارد، بنویسید.

```
for( int x=0 ; x <= 25 ; x++ )
for( int y=0 ; y <= 10 ; y++ )
for ( int z=0 ; z <= 5 ; z++ )
if ( 2000*x + 5000*y + 10000*z == 50000 )
Console.WriteLine("X={0},Y={1},Z={2}",x,y,z) ;
```

- برنامه را طوری تغییر دهید که وجه نقد درخواستی کاربر را از ورودی دریافت کند.

```
int p= int.Parse(Console.ReadLine());  
for( int x=1 ; x <= p/2 ; x++ )  
    for( int y=1 ; y <= p/5 ; y++ )  
        if ( 2000*x + 5000*y == p)  
            Console.WriteLine ( "X={0},Y={1}",x,y) ;
```

واحد یادگیری ۴

شایستگی کار با آرایه

مقدمات تدریس

الف) مفاهیم کلیدی

| مفاهیم کلیدی | | |
|--------------|---------------|------------------|
| رشته | عنصر | آرایه |
| کلاس آرایه | جست و جوی خطی | پیمایش آرایه |
| | کاراکتر | جست و جوی دودویی |

| طرح درس روزانه (هفتگی) پیشنهادی | | | | |
|--|---|--|--|-------------|
| کلاس: یازدهم | | درس: کار با آرایه | | |
| پیام جلسه (هدف کلی): هنرجو بتواند مسئله آرایه‌ای را تشخیص داده و آن را با آرایه انجام دهد. | | | | |
| اهداف یادگیری | فعالیت‌ها | | زمان | فیزیکی |
| فعالیت | طبقه هدف: حیطه عاطفی / شناختی / روانی - حرکتی | کار هنرآموز | کار هنرجو | مدت (دقیقه) |
| ارزشیابی رفتار ورودی | سنجش میزان آگاهی هنرجویان از کد نویسی و آشنایی با مفهوم آرایه | تعریف آرایه - اعلان آرایه - معرفی شماره عناصر آرایه | مشارکت در پاسخگویی و تعامل | ۳۰ |
| ایجاد انگیزه | نمایش فیلم | با استفاده از آرایه یک مسئله را حل کند. | یک روندنما برای آرایه ترسیم کند. | ۳۰ |
| ارائه مفاهیم کلیدی (توضیح هنرآموز) | توضیح کامل مفاهیم کلیدی (دانشی) و ایجاد علاقه و انگیزه در هنرجویان (بینشی) | مسئله‌ای را با استفاده از آرایه حل کند. روش‌های جست‌وجو را بیان کرده و کد نویسی لازم را انجام دهد. | هنرجویان مطالب پایه را می‌آموزند. | ۸۰ |
| فعالیت کارگاهی (تمرین هنرجویان) | هنرجو باید توانایی کار با آرایه را داشته باشد و کارگاه عملی و فعالیت کارگاهی را انجام دهد. | هنرآموز توضیح کلی در مورد فعالیت کارگاهی به هنرجو داده و از آنها می‌خواهد به صورت گروهی به انجام آن بپردازند. | با توجه به فعالیت‌های کارگاهی کتاب، هنرجویان در گروه‌های خود، گزینه‌های دیگر را با آزمون و خطا می‌آموزند. | ۸۰ |
| ارزیابی فعالیت‌ها ارائه تمرین | طرح یک مسئله که با استفاده از حلقه حل می‌شود. | از هنرجویان خواسته می‌شود در گروه‌های تعیین شده به انجام فعالیت‌های کارگاهی بپردازند. قبل از شروع به کار هنرجویان، ملاک‌های ارزیابی تمرین‌ها به آنها داده می‌شود. | هنرجویان در گروه‌های خود سعی می‌کنند با دقت و سرعت و با خلاقیت به انجام فعالیت کارگاهی بپردازند. | ۵۰ |
| ارائه نکات تکمیلی (جمع‌بندی) | هنرجو باید مسئله را شناخته و آرایه مناسب را انتخاب کرده و روندنمای مربوط به آن را ترسیم کرده و به کد نویسی بپردازد. | با توجه به مفاهیم بیان شده، بخشی از فعالیت‌های کارگاهی را برای هنرجویان شرح دهد. | توجه و دقت و پرسش و پاسخ انجام فعالیت‌های گروهی و ارائه راه‌حل‌های جدید برای مسائل. | ۷۰ |
| ارزشیابی شایستگی (ارزشیابی پایانی) | هنرجو باید بتواند یک مسئله را شناخته و آرایه مناسب برای حل آن انتخاب کرده، کد نویسی را عملی انجام دهد. | این ارزشیابی در دو شاخه انفرادی و گروهی انجام می‌شود: از هنرجویان خواسته می‌شود در گروه‌های تعیین شده به انجام پروژه این بخش از واحد کار بپردازند. نمره‌ای به کار گروهی هنرجویان داده می‌شود. آزمون برای هنرجویان به صورت انفرادی برگزار می‌شود. نمره گروهی در نمره انفرادی افراد تأثیرگذار است. قبل از شروع به کار، ملاک‌های ارزیابی پروژه به آنها داده می‌شود. | هنرجویان در گروه‌های خود سعی می‌کنند. در مدت‌زمان تعیین شده با نهایت دقت و سرعت با ایجاد خلاقیت به انجام پروژه بپردازند. | ۸۰ |
| ابزارهای مورد نیاز ویدئو پروژکتور، رایانه، تخته آموزشی، برنامه VS | | | | |

ج) ورود به بحث

بارها اتفاق افتاده است که شما احتیاج به ذخیره داده‌هایی برای استفاده در حال و آینده داشته‌اید. از هنجروی خود بپرسید که آیا تا به حال به فهرستی از داده برخورد کرده است؟ آیا در اطراف خود داده‌های هم‌نوعی را دیده است؟ شاید بزرگ‌ترین چالش این فصل درک مفهوم پلاک و اندیس خانه‌های آرایه است. بهتر است خیابان یا کوچه‌ای که هنجروی شما در آن زندگی می‌کند را مثال بزنید. مثال‌های متنوعی را برای هنجرویان خود مطرح کنید تا در حین تدریس این مشکل را به حداقل برسانید. مثلاً به آن‌ها بگویید فهرستی از اسامی دوستان هم‌محلله‌ای خود تهیه کنند.

به‌منظور تعیین سطح دانش‌آموزان می‌توانید سؤالاتی از این دست طرح کنید:

سؤال ۱: به نظر شما چگونه می‌توانیم قد دانش‌آموزان یک کلاس را ثبت کنیم؟

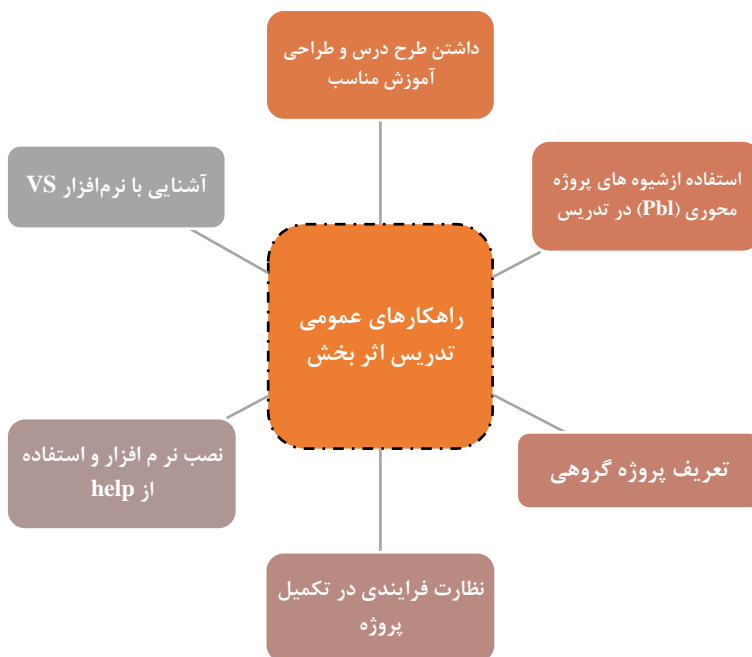
سؤال ۲: کدام یک از شما قبلاً تعدادی اسم را در کنار هم نوشته‌اید؟ مثال بزنید.

سؤال ۳: آیا تا به حال در یک کوچه به دنبال یک شخص که فقط نام خانوادگی او را می‌دانید گشته‌اید؟

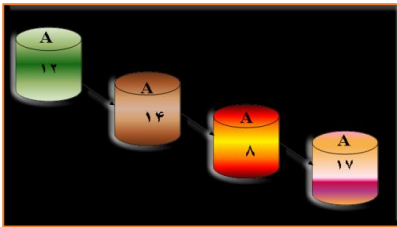
سؤال ۴: آیا تا به حال فهرستی از اسامی و داده‌ها را دیده‌اید؟ مثال بزنید.

سؤال ۵: آیا تا به حال در بین تعدادی از اسامی به دنبال اسم خاصی گشته‌اید؟

سؤال ۶: چند نمونه از کاربردهای فهرست را نام ببرید.



در برنامه‌نویسی یاد گرفتیم که یک متغیر را تعریف کرده و سپس آن را مقداردهی کنیم. اما مشکلی که پیش می‌آید این است که مقدار دوم باعث از بین رفتن مقدار اول و به همین ترتیب مقدار سوم باعث از بین رفتن مقدار دوم می‌شود و این روال همچنان ادامه دارد. (شکل زیر را ببینید)



در این شکل مقدار نهایی متغیر A عدد ۱۷ خواهد بود و مقادیر قبل از ۱۷ همگی از بین خواهند رفت. حال این سؤال پیش می‌آید که اگر بخواهیم داده‌های قبلی این متغیر را نگهداری کنیم چه کار باید انجام دهیم؟

مثال: برنامه‌ای بنویسید که ۱۰۰ عدد را به عنوان دمای شهر بخواند، ابتدا اعداد مثبت (دمای بالای صفر) و سپس اعداد منفی (دمای زیر صفر) را نمایش دهد.

در صورتی که از روش‌های قبلی استفاده کنید برای ذخیره این اعداد به صورت جداگانه به ۱۰۰ متغیر احتیاج خواهید داشت. شاید بگویید از حلقه‌ها استفاده می‌کنیم. آیا می‌توانید داده‌ها را نگهداری کنید؟ چه مشکلاتی برای حل این‌گونه مسائل خواهید داشت؟ باید نوع داده‌ای داشته باشیم که ۱۰۰ عدد را در خود نگهداری کند. یعنی نوع داده‌ای که ۱۰۰ فضای خالی و هم‌نوع و هم‌اندازه در اختیار شما بگذارد. به این‌گونه متغیرها آرایه می‌گویند.

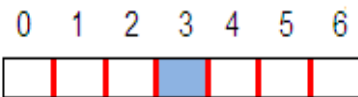
فرض کنید یک تکه مستطیل به نام P دارید و آن را به قسمت‌های مساوی تقسیم می‌کنیم. (شکل زیر)



مستطیل مشکی‌رنگ را با خطوط قرمز به قسمت‌های مساوی تقسیم کرده‌ایم. راه‌حل دسترسی به خانه رنگی‌شده شماره‌گذاری تمام خانه‌ها است.

حال به شکل جدید نگاه کنید. در تمام زبان‌های برنامه‌نویسی رایج شماره خانه‌های آرایه که به آنها اندیس آرایه می‌گویند از صفر شروع می‌شود. پس برای دسترسی به خانه شماره ۴ از آرایه p باید اندیس ۳

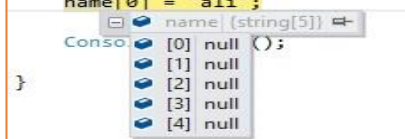
را در نظر بگیریم ($p[3]$).



می‌توانیم برای انجام عملیات بر روی آرایه از حلقه‌های معین و نامعین (که عملیات شمارش را انجام می‌دهند) استفاده کنیم.

```
static void Main(string[] args)
{
    string[] name;
    name = new string[5];
    name[0] = "ali";
    Console.WriteLine(name);
}

```



به تکه برنامه زیر دقت کنید. آرایه computer دارای ۵ خانه هست که هر کدام از این خانه‌ها می‌تواند یک‌رشته را در خود جای دهد، پس نوع آرایه رشته‌ای است.

```
string[] computer = new string[5];
long[] price = new long[5];

```

چون خانه‌های آرایه در کنار هم قرار دارند، بنابراین می‌توان با استفاده از حلقه‌ها بر روی آنها عملیات خواندن، نوشتن یا پردازش را انجام داد.

```
for (int i = 0; i <= 4; i++)
{
    computer[i] = Console.ReadLine();
    price[i] = int.Parse(Console.ReadLine());
}

```

یا

```
for (byte i = 0; i <= 6; ++i)
{
    Console.WriteLine(week[i]);
    ++i;
}

```

یا

```
for (int i = 0; i <= price.Length; i++)
    sumPrice += price[i];

```

شیوه و الگوی پیشنهادی

از هنرجویان خود بخواهید که اسامی دو نفر از همکلاسی‌های خود را یادداشت کنند. حالا دوباره بگویید دو نفر دیگر را یادداشت کنند، این کار را برای ۱۰ هنرجو انجام دهید. از آنها بپرسید راهکار جدیدی برای این کار سراغ دارند؟ حتماً می‌گویند نام ۱۰ نفر را با هم وارد کنیم. حالا از آنها بخواهید که به هر کدام از این اسامی یک شماره اختصاص بدهند. به هنرجویان بگویید نفر شماره ۳ و ۵ را نام ببرند.

از هنرجویان بخواهید در بین اسامی نوشته‌شده نام علی را پیدا کنند و شماره خانه‌اش را بنویسند. این بار از آنها بخواهید تا اسامی را مرتب کنند و بار دیگر اسامی را از انتها به ابتدا یادداشت کنند.

مشکلات متداول در فرآیند یاددهی - یادگیری

شاید بتوان گفت بزرگ‌ترین مشکل درک مفهوم اندیس خانه‌های آرایه است، چراکه اندیس آرایه از شماره صفر شروع می‌شود. درک هنرجویان بر اساس پلاک خانه‌هایی است که در آنها زندگی می‌کنند که از شماره ۱ شروع می‌شوند و این می‌تواند یک چالش برای بیان آرایه باشد.

پاسخ به فعالیت‌ها

روندنمایی رسم کنید که اسامی پنج هنرجو را دریافت کرده سپس اسامی را از آخر به اول نمایش دهد. برنامه این الگوریتم را به زبان سی شارپ بنویسید.

فعالیت کارگاهی

ص ۷۹

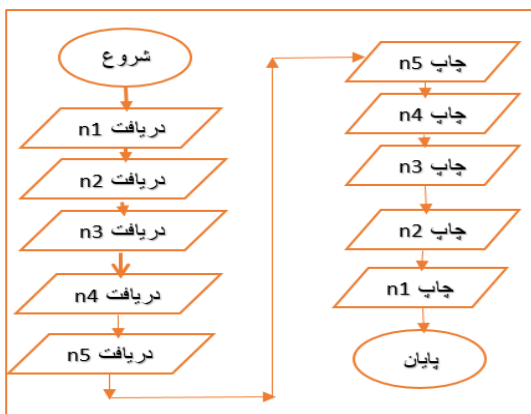
پاسخ:

- ۰ شروع
- ۱ خواندن n1,n2,n3,n4,n5
- ۲ نمایش n5,n4,n3,n2,n1
- ۳ پایان

در این فعالیت مشاهده می‌کنیم که به تعداد اسامی باید دستورات ورودی و خروجی بکار ببریم. نکته این سؤال در این است که از حلقه for نیز نمی‌توان استفاده کرد. در چنین مسئله‌هایی بهترین روش استفاده از آرایه است.

```
string n1, n2, n3, n4, n5;  
n1 = Console.ReadLine();  
n2 = Console.ReadLine();  
n3 = Console.ReadLine();  
n4 = Console.ReadLine();  
n5 = Console.ReadLine();
```

```
Console.WriteLine("{0}{1}{2}{3}{4}", n5, n4, n3, n2, n1);
```



فعالیت کارگاهی
ص ۸۰

یک پروژه جدید ایجاد کنید و دستور اعلان آرایه و دستور ایجاد آرایه name را بنویسید. بررسی کنید محتوای هر عنصر آرایه چیست؟

```
static void Main(string[] args)
{
    string[] name;
    name = new string[5];
    name[0] = "ali";
    Console.WriteLine(name);
}
```

فعالیت کارگاهی
ص ۸۱

- آرایه‌ای به نام salary برای نگهداری حقوق ۳۰۰ کارمند اعلان کنید.
- آرایه‌ای به نام vowels برای نگهداری حروف صدادار انگلیسی اعلان کنید.
- آرایه‌ای به نام lamp برای نگهداری وضعیت روشن و خاموش بودن پنج لامپ اعلان کنید.
- آرایه‌ای با نام دلخواه برای نگهداری اسامی ماه‌های سال اعلان کنید.
- آرایه‌ای با نام دلخواه برای نگهداری معدل هنرجویان کلاس اعلان کنید.
پاسخ:

- 1) long [] salary = new long [300];
- 2) char [] vowels = new char [5];
- 3) byte [] lamp = new byte [5];
- 4) string [] months = new string[12];
- 5) float [] average = new float [30];

کنجکاوی
ص ۸۱

چرا برای مراجعه به خانه سوم آرایه، از اندیس ۲ استفاده شده است؟
پاسخ: چون اندیس‌های خانه‌های آرایه از صفر شروع می‌شوند.

کارگاه ۱ – تعریف و مقداردهی آرایه

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۸۲

- برای ذخیره کردن اسامی قطعات دیگر، چه تغییری در کد باید بدهیم؟
پاسخ:

ابتدا باید تعداد خانه‌های آرایه را زیاد کنیم و سپس آنها مقداردهی کنیم.
- برای نگهداری قیمت قطعات رایانه، آرایه `price` را متناظر با آرایه نام قطعات اعلان کنید.

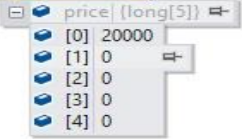
پاسخ:

```
string[] computer = new string[5];  
long[] price = new long[5];
```

- برنامه را با کلید F10 تا رسیدن به دستور مقداردهی `price[2]` اجرا کنید و محتوای آرایه `price` را مشاهده کنید.

پاسخ:

```
string[] computer = new string[5];  
long[] price = new long [5];  
  
computer[0] = "cpu";  
computer[1] = "main";  
computer[2] = "ram";  
  
price[0] = 20000;  
price[1] = 15000;
```

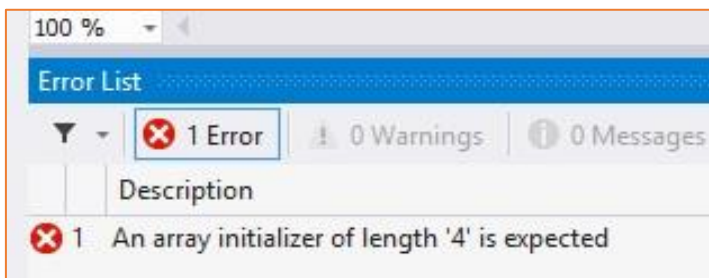


- دستور زیر را جایگزین کد مرحله ۹ کنید. به جای عدد ۴ عدد ۵ را قرار دهید. خطای رخ داده چیست؟

```
string[] computer= new string[4]{"CPU", "MainBoard",  
"RAM", "HDD", "DVD"};
```

پاسخ:

در صورتی که تعداد خانه‌های آرایه نسبت به مقادیر وارد شده کمتر باشد برنامه خطا نشان خواهد داد.



- دستوری برای تغییر مقدار عنصر شماره ۳ آرایه computer به "keyboard" به کد اضافه کنید.

پاسخ:

```
computer[2] = "keyboard";
```

کنجکاوی

ص ۸۵

آیا می‌توان با یک متد `writeline()` تمام عناصر آرایه را نمایش داد؟ حاصل اجرای دستور زیر چیست؟

```
Console.WriteLine(price);
```

پاسخ: خیر، در صورتی که از این دستور استفاده کنید متد `ToString` برای شی مورد نظر صدا زده شده و پیش فرض عملکرد متد مذکور اجرا می‌شود. به همین جهت مقدار `System.Int64[]` در خروجی چاپ می‌شود.

```
byte countMark;  
byte countStudent;  
float mark;  
double sum = 0;  
double average = 0;  
double minAverage = 0;  
double maxAverage = 0;
```

```
maxAverage = 0;  
minAverage = 20;
```

```
for (countStudent = 1; countStudent <= 2; countStudent++)
```

```

}
Console.WriteLine("    === student {0}    === \n" ,
countStudent );
sum = 0;

for (countMark = 1; countMark <= 5; countMark++)
{
Console.Write("please enter your mark {0} ==> ",
countMark);
mark = Single.Parse(Console.ReadLine());
if ((mark >= 0) && (mark <= 20))
sum = sum + mark;
{
average = sum / 5;
Console.WriteLine("average student {0} is ==> {1}",
countStudent, average);
if (average > maxAverage)
maxAverage = average;
if (average < minAverage)
minAverage = average;
Console.WriteLine("\n ===== \n");
{
Console.WriteLine("Max average in your class is ==> {0}
", maxAverage);
Console.WriteLine("Min average in your class is ==> {0}
", minAverage);
}
}
}
}

```

کارگاه ۲ – نمایش عناصر آرایه

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۸۵

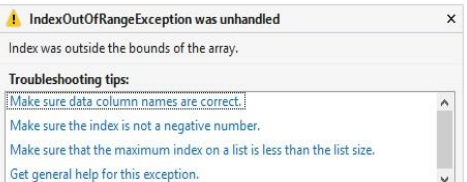
- برنامه را با عدد ۷ اجرا کنید.

پاسخ:

چون عدد ۷ از آخرین اندیس این آرایه بیشتر است با خطایی از نوع `System.IndexOutOfRangeException` مواجه می‌شود که نشان می‌دهد آرایه دارای عنصری با این اندیس نیست.

```
string[] week = { "sat", "sun", "mon", "tus", "wen", "ter", "fri" };
byte num = byte.Parse(Console.ReadLine());
Console.WriteLine(week[num]);

Console.ReadKey();
```



- اگر بخواهیم به جای اعداد ۰ تا ۶، اعداد ۱ تا ۷ دریافت شود، چه تغییری باید در برنامه ایجاد کنیم؟

پاسخ:

```
Console.WriteLine(week[--num]);
```

- دستوراتی به برنامه اضافه کنید که اگر شماره وارد شده در محدوده مجاز نبود، پیام مناسب نمایش داده شود.

پاسخ:

```
if (num < 1 || num > 7)
    Console.WriteLine("please enter correct number !!!");
else
    Console.WriteLine(week[--num]);
```

فعالیت کارگاهی
ص ۸۶

برنامه‌ای بنویسید که اسمی روزهای زوج آرایه `week` را نمایش دهد.

پاسخ:

```
for (int i = 0; i <= 6; i += 2)
{
    Console.WriteLine(week[i]);
}
```

کارگاه ۳ - دریافت عناصر آرایه از کاربر

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۸۷

- تعداد قطعات رایانه را از ورودی دریافت کنید.
پاسخ:

```
int count;
count = int.Parse(Console.ReadLine());
long[] price = new long[count];
string[] computer = new string[count];
```

```
for (int i = 0; i < count; i++)
{
    computer[i] = Console.ReadLine();
}
```

- برنامه را تغییر دهید تا علاوه بر عناصر آرایه computer، قیمت قطعات را هم دریافت کند.

پاسخ:

```
for (int i = 0; i < count; i++)
{
    computer[i] = Console.ReadLine();
    price[i] = long.Parse(Console.ReadLine());
}
```

تکمیل کارگاه
ص ۸۷

- برنامه را طوری تکمیل کنید که نام، قیمت قطعات و قیمت کل را نمایش دهد.
پاسخ:

```
int count;
long sumprice = 0;
count = int.Parse(Console.ReadLine());
long[] price = new long[count];
string[] computer = new string[count];
for (int i = 0; i < count; i++)
{
    computer[i] = Console.ReadLine();
    price[i] = long.Parse(Console.ReadLine());
}
for (int i = 0; i < price.Length; i++)
    sumprice += price[i];
for (int i = 0; i < count; i++)
    Console.WriteLine(" {0} {1} ", computer[i], price[i]);
Console.WriteLine(sumprice);
```

کارگاه ۴ - پیمایش نویسه‌های رشته

پاسخ به فعالیت‌ها

تکمیل کارگاه

ص ۸۸

- دستورات لازم را برای دریافت یک رشته از کاربر بنویسید.

پاسخ:

```
string mystr;  
mystr = Console.ReadLine();
```

- دستورات زیر را به کد اضافه کنید. خروجی آن چیست؟

```
for (int i = 0; i < mystr.Length; i++)  
Console.WriteLine(mystr[i]);
```

پاسخ: خروجی نمایش کاراکترهای رشته ورودی به صورت ستونی است.

- کد زیر را جایگزین مرحله ۳ کنید.

```
foreach (char c in mystr)  
Console.WriteLine(c);
```

پاسخ: دقیقاً همان رشته را تولید می‌کند.

- برای نمایش معکوس رشته، کد زیر را تکمیل کنید.

```
for (int i = mystr.Length - 1; i >= 0; i--)  
Console.WriteLine(mystr[i]);
```

- آیا می‌توان این کد را با دستور foreach نوشت؟

پاسخ: خیر

کارگاه ۵ - ویرایش عناصر آرایه

پاسخ به فعالیت‌ها

تکمیل کارگاه

ص ۸۹

- با اعلان آرایه و متغیرهای مناسب، کد زیر را برای دریافت محصولات تکمیل کنید.

پاسخ:

ابتدای کد اضافه شود:

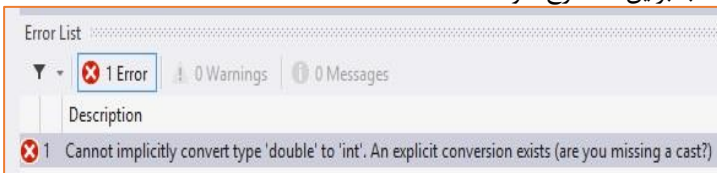
```
int i;  
int temp;  
int[] priceList = new int[50];
```

- دستورات محاسبه مالیات ارزش افزوده را بنویسید. در این کد نقش متغیر i و j چیست؟

پاسخ:

متغیر i شماره محصولات وارد شده را نگه‌داری می‌کند. متغیر j به تعداد محصولات وارد شده که درمکان i ذخیره شده است امکان محاسبه مالیات بر ارزش افزوده و ذخیره در همان خانه متناظر را ایجاد می‌کند.

- در قطعه کد ۳ تبدیل صریح int را حذف کنید. چه خطایی رخ می‌دهد؟ چرا؟
پاسخ: چون محاسبه ارزش افزوده مقدار اعشاری (double) تولید می‌کند و فضای حافظه اشغالی توسط متغیر اعشاری بیشتر از فضای اشغالی متغیر صحیح (int) است بنابراین خطا رخ خواهد داد.



برنامه‌ای بنویسید که هزینه روزانه یک ماه شما را دریافت کرده، در یک آرایه ذخیره کند. سپس هزینه ماهانه و میانگین هزینه روزانه شما را محاسبه کرده، نمایش دهد.

پاسخ:

```
int temp;
double totalPrice;
int[] cost = new int[30];
int i;
for ( i = 0; i < 30; i++)
{
    Console.WriteLine("Enter the cost of {0} th day : ", i+1);
    temp = int.Parse(Console.ReadLine());
    if (temp > 0)
        cost[i] = temp;
    else
        break;
}
totalPrice = 0;
for (int j = 0; j < i; j++)
    totalPrice = totalPrice + cost[j];
Console.WriteLine("sum cost of month ==> {0}",
totalPrice);
Console.WriteLine("average cost of day ==> {0} ",
(totalPrice / i));
```

فعالیت منزل

ص ۹۰

شرح برنامه:

متغیر temp برای دریافت مقدار هزینه روزانه تعریف شده است، بنابراین اگر این هزینه صفر یا منفی باشد حلقه (i) ایجاد شده با دریافت چنین عددی خاتمه پیدا خواهد کرد و در حلقه دوم مجموع هزینه‌ها و میانگین آنها، به تعداد روزها محاسبه خواهد شد (j) که در حلقه بالا برای آنها هزینه روزانه دریافت شده است. بنابراین می‌بینید که متغیر j با متغیر i مقایسه شده است. در متغیر sum_COST مجموع هزینه‌ها حساب شده است. برای اینکه میانگین هزینه‌ها به صورت اعشاری نمایش داده شود از تبدیل مستقیم (double)sum_COST / i استفاده شده است.

جست‌وجو در آرایه

در روش جست‌وجوی خطی، جست‌وجوی اطلاعات معمولاً در فهرست نامرتب صورت می‌گیرد. در فهرست نامرتب، برای یافتن یک قلم اطلاعات خاص، تنها راه حل این است که عناصر فهرست را از اولین عنصر تا آخرین عنصر (یا بالعکس) با اطلاعات موردنظر مقایسه کنیم. اگر عنصری از فهرست، با اطلاعات موردنظر برابر باشد یا به پایان فهرست برسیم، عمل جست‌وجو خاتمه می‌یابد. فرض کنید فهرست زیر (آرایه X) موجود است :

| X | X[0] | X[1] | X[2] | X[3] |
|---|------|------|------|------|
| | ۱۵ | ۵ | ۲۰ | ۳۵ |

در این فهرست می‌خواهیم عدد ۲۰ را به طور خطی جست‌وجو کنیم. مقدار ۲۰ را به خانه اول مقایسه می‌کنیم. اگر برابر نبود باید به خانه بعدی برویم و تا زمانی که محتوای خانه‌ای ۲۰ نباشد جست‌وجو را ادامه دارد. این جست‌وجو دو حالت دارد. حالت اول: جست‌وجوی موفق، یعنی مقدار جست‌وجو شده در فهرست باشد. حالت دوم: جست‌وجوی ناموفق، یعنی مقدار جست‌وجو شده در فهرست نباشد. تعداد جست‌وجوها وابسته به بودن و یا نبودن عنصر مورد نظر است. در صورتی که عنصر موجود باشد به شماره محل قرارگیری عنصر شمارش صورت گرفته است. (در خانه اول یک جست‌وجو و در خانه دهم، ۱۰ جست‌وجو) ولی اگر عنصر مورد نظر موجود نباشد، باید تمام خانه‌ها مورد جست‌وجو قرار بگیرند که در این صورت تعداد جست‌وجوها به اندازه طول آرایه خواهد بود.

پاسخ به فعالیت‌ها

فعالیت گروهی
ص ۹۱

یک آرایه نامرتب شامل اسامی ۱۰۰ هنرجو داریم. برای پیدا کردن نام یک هنرجو در این آرایه به روش جست‌وجوی خطی چند مقایسه نیاز است؟ جدول را کامل کنید.

پاسخ:

| مکان عنصر موردنظر | اول آرایه | وسط آرایه | آخر آرایه | در آرایه نباشد |
|-------------------|-----------|-----------|-----------|----------------|
| تعداد مقایسه | ۱ | ۵۰ | ۱۰۰ | ۱۰۰ |

- در هریک از موارد زیر تعیین کنید از روش جست‌وجوی خطی یا دودویی استفاده می‌شود؟ انتخاب روش بر اساس چه معیاری است؟
- ۱) یافتن کارنامه تحصیلی یک هنرجو در بین سایر کارنامه‌های مرتب‌شده بر اساس نام هنرجویان
 - ۲) پیدا کردن یک کتاب بر اساس نام آن کتاب در کتابخانه
 - ۳) پیدا کردن یک کتاب بر اساس کد کتاب در کتابخانه
 - ۴) پیدا کردن نام هنرجویی که در کلاس بالاترین معدل را دارد، اگر اسامی بر اساس معدل مرتب شده باشند.

پاسخ:

| معیار انتخاب | نوع جست‌وجو | ردیف |
|---|-------------|------|
| فهرست کارنامه‌ها بر اساس نام هنرجویان مرتب شده است. | دودویی | ۱ |
| فهرست مرتب شده نیست. | خطی | ۲ |
| فهرست مرتب شده نیست. | خطی | ۳ |
| فهرست هنرجویان بر اساس معدل آنها مرتب شده است. | دودویی | ۴ |

کارگاه ۶ – پیاده‌سازی جست‌وجوی خطی

پاسخ به فعالیت‌ها

– برای جست‌وجو در آرایه car کدهای زیر را بنویسید.
پاسخ: این بخش برای آموزش جست‌وجوی خطی بیان شده است. در روش جست‌وجوی خطی مقدار مورد جست‌وجو که در اینجا واژه pride است با محتوای تک‌تک خانه‌های آرایه مقایسه می‌شود. دو حالت رخ خواهد داد، در حالت اول محتوای خانه آرایه نام دیگری به‌غیر از pride است، مثلاً mazda که در این حالت جست‌وجو از خانه بعدی ادامه خواهد یافت و در حالت دوم مقدار خانه با واژه pride برابری خواهد کرد که در این صورت نام ماشین و محل خانه یافت‌شده به نمایش در خواهد آمد.

– برنامه را طوری تغییر دهید تا قیمت خودروی مورد جست‌وجو در خروجی نمایش داده شود.

پاسخ:

```
Console.WriteLine("{0} found in {1} and price {2}",
item, i, price[i]);
```

– مقدار item را برابر Hyundai قرار دهید. خروجی چیست؟

پاسخ:

```
String item = "Hyundai";
```

خروجی مانند قسمت قبل است با این تفاوت که به جای Hyundai ماشین pride جست‌وجو و اطلاعات آن ماشین نمایش داده می‌شود.

- برای تشخیص یافتن و یا عدم یافتن عنصر، کد را به صورت زیر تغییر دهید. نقش متغیر found در این کد چیست؟

پاسخ:

متغیر found نقش یک پرچم را بازی می‌کند که در صورتی که مقدار آن true شود اطلاعات ماشین مورد نظر نمایش داده می‌شود و در غیر این صورت یک پیام مناسب برای اطلاع کاربر نمایش می‌دهد. شرط false در انتهای حلقه بررسی می‌شود. برنامه را طوری تغییر دهید که نام خودرو را از ورودی دریافت کند.

پاسخ:

```
item = Console.ReadLine();
```

فعالیت منزل

ص ۹۳

برنامه‌ای بنویسید که نام و نمره هنرجویان را از ورودی دریافت کرده، در آرایه مناسب ذخیره کند. سپس نام هنرجویانی را نمایش دهد که نمره ۲۰ گرفته‌اند.

پاسخ:

```
int length = 10;
string[] name = new string[length];
double[] mark = new double[length];
for (int i = 0; i < length; i++)
{
    name[i] = Console.ReadLine();
    mark[i] = double.Parse(Console.ReadLine());
}
for (int i = 0; i < length; i++)
    if (mark[i] == 20)
        Console.WriteLine("mark {0} is 20. ", name[i]);
```

فعالیت گروهی

ص ۹۳

- هنرجویی یک عدد بین ۱ تا ۱۰۰ انتخاب کرده، آن را یادداشت می‌کند. برنده شخصی است که با کمترین تعداد، عدد مورد نظر را حدس بزند. شما چه الگوریتمی برای برنده شدن پیشنهاد می‌کنید؟ بهتر است اولین حدس چه عددی باشد؟

پاسخ:

بهتر است اولین حدس عدد ۵۰ باشد. و بهترین الگوریتم استفاده از عدد وسط یعنی ۵۰ است. حالا با توجه به بزرگ‌تر بودن یا کوچک‌تر بودن در نیمه اول اعداد (۱-۴۹) و نیمه دوم اعداد (۵۱-۱۰۰) باز هم باید حد وسط را ادامه دهیم.

- هنرجویی عدد ۵۹ را یادداشت کرده است. در روش جست و جوی دودویی حداکثر با چند مقایسه عدد مورد نظر حدس زده خواهد شد. جدول زیر را کامل کنید.
پاسخ :

برای یافتن عدد ۵۹ با پنجمین مقایسه عدد به دست آمده است.

| مقایسه | نتیجه مقایسه | عدد مورد نظر | عدد حدس زده شده (وسط آرایه) | آخرین عدد آرایه | اولین عدد آرایه |
|--------|-----------------|--------------|-----------------------------|-----------------|-----------------|
| ۱ | عدد بزرگتر است. | ۵۹ | ۵۰ | ۱۰۰ | ۱ |
| ۲ | عدد کوچکتر است. | ۵۹ | ۷۵ | ۱۰۰ | ۵۱ |
| ۳ | عدد کوچکتر است. | ۵۹ | ۶۲ | ۷۴ | ۵۱ |
| ۴ | عدد کوچکتر است. | ۵۹ | ۵۶ | ۶۱ | ۵۱ |
| ۵ | عدد درست است. | ۵۹ | ۵۹ | ۶۱ | ۵۷ |

کارگاه ۷ - متدهای کلاس Array

پاسخ به فعالیت‌ها

- مقدار متغیر `found` پس از اجرای برنامه چیست؟ عملکرد متد `IndexOf` چیست؟

پاسخ:

مقدار `found` پس از اجرای برنامه ۲ است و عملکرد متد `IndexOf`: اندیس اولین وقوع یک مقدار را در آرایه تعریف شده برمی گرداند.

`VariableName = Array.IndexOf (array_name, value);`

- اگر `X=60` قرار دهیم خروجی چه می‌شود؟

پاسخ :

به ازای مقدار `x=60` برنامه اندیس ۱- را نمایش می‌دهد.
کد زیر را به برنامه اضافه کنید.

```
int found = Array.LastIndexOf(a, x);
```

```
Console.WriteLine(found);
```

مقدار متغیر `found` پس از اجرای این دستور چیست؟ تفاوت عملکرد متد `LastIndexOf` با متد `IndexOf` چیست؟

پاسخ:

تکمیل کارگاه
ص ۹۴

مقدار found پس از اجرای برنامه ۶ است.
عملکرد متد LastIndexOf: اندیس آخرین وقوع یک مقدار را در آرایه تعریف شده برمی گرداند.
تفاوت متد IndexOf و LastIndexOf در این است که IndexOf اولین وقوع و LastIndexOf آخرین وقوع را برمی گرداند.
- با دستور foreach عناصر آرایه a را در خروجی نمایش دهید. عملکرد متد Sort چیست؟
پاسخ:

```
foreach (int temp in a)
    Console.WriteLine(temp);
```

عملکرد متد Sort: عناصر موجود در آرایه تعریف شده را مرتب می کند.
- کد زیر را به برنامه اضافه کرده، برنامه را اجرا کنید. مقدار found چیست؟

```
int x = 10;
int found = Array.BinarySearch(a, x);
Console.WriteLine(found);
```

پاسخ: مقدار found پس از اجرای برنامه 1 است.
- کد زیر را به برنامه اضافه کنید و دوباره با دستور foreach عناصر آرایه a را در خروجی نمایش دهید.

```
Array.Reverse(a);
```

عملکرد متد Reverse(): عناصر موجود در آرایه تعریف شده را معکوس می کند.
- کد زیر را اضافه کرده، برنامه را اجرا کنید. مقدار متغیر found چیست؟

```
found = Array.BinarySearch(a, x);
Console.WriteLine(found);
```

پاسخ: مقدار found پس از اجرای برنامه 1- است.

الف) نمونه ارزشیابی پایانی

ارزشیابی واحد یادگیری ۳

راهنمای پاسخ به سوالات: هنرجوی گرامی برای پاسخگویی به سوالات به ترتیب زیر عمل کنید.

- در بخش الف به سوالات ۱ و ۲ و ۳ پاسخ دهید.
- در بخش ب به سوالات ۴ و ۵ پاسخ دهید.
- در بخش ج به سوالات ۶ و ۷ پاسخ دهید.
- با مراجعه به جدول ارزش‌دهی عملکرد می‌توانید سطح شایستگی خود را بسنجید.
- ✓ مسائل :

الف- علی برای انجام کار بانکی پدرش به بانک تجارت رفته است. بر اساس قانون بانک‌ها، علی از دستگاه نوبت‌دهی شماره‌ای را دریافت می‌کند. شماره علی ۲۱۶ است. آخرین شماره اعلام شده در لحظه ورود علی ۱۷۳ بود. (در هر لحظه ۱ شماره از دستگاه بلندگوی بانک اعلام می‌شود).

ب - در یک فروشگاه مواد غذایی، ماکارونی ۵ شرکت مختلف به فروش می‌رسد. روز پنج‌شنبه هر هفته ۱۰ درصد تخفیف به این ماکارونی‌ها داده می‌شود. بهار روز پنج‌شنبه برای خرید به این فروشگاه مراجعه کرده است. برنامه‌ای بنویسید که به بهار کمک کند تا بیشترین تخفیف را به دست آورده و آن محصول را خریداری کند.

ج- سینا و مینا به ساعت‌فروشی خیابان فردوسی رفته‌اند و از فروشنده در مورد نحوه کارکرد ساعت کامپیوتری که دیده‌اند سوال کرده‌اند. فروشنده به آنها گفته است که سه نوع شمارش صورت می‌گیرد. ساعت، دقیقه، ثانیه. برنامه‌ای بنویسید که به فروشنده در تشریح شمارنده‌های ساعت کمک کند و با دیدن خروجی برنامه سینا و مینا با نحوه کارکرد ساعت دیجیتال آشنا شوند.

| ردیف | سوالات |
|------|--|
| ۱ | ورودی و خروجی، داده‌ها و اطلاعات مسئله را بنویسید. |
| ۲ | راه حل انتظار علی برای اعلام شماره‌اش را بنویسید. |
| ۳ | در صورتی که هر بار ۳ شماره اعلام شود، راه حل خود را برای اعلام شماره علی تغییر دهید. |
| ۴ | اجزاء ساختار حلقه را معین کنید. |
| ۵ | برنامه موردنظر را در محیط زبان برنامه نویسی VS بنویسید. |
| ۶ | تعداد حلقه‌های برنامه را تعیین کنید. |
| ۷ | برنامه مورد نظر را بنویسید. برنامه را محله به محله اجرا کرده و خطاهای احتمالی را برطرف کنید. |

جدول ارزش‌دهی عملکرد

| ایجاد برنامه با حلقه‌های متداخل | | ایجاد برنامه با حلقه (مرحله بحرانی) | | | حل مسئله تکرار | | | مراحل (حداقل نمره) | |
|-------------------------------------|--------------|--|------------------------------|---------------|-------------------------|---------------------------|-------------------------|------------------------------|--|
| اجرای گام به گام برنامه - رفع خطاها | نوشتن برنامه | تعیین تعداد حلقه‌های یک مسئله | خطایابی و کنترل اجرای برنامه | نوشتن کد حلقه | تعیین اجزای ساختار حلقه | توسعه راه حل در صورت نیاز | ارائه راه حل برای مسئله | تعیین داده‌ها، ورودی و خروجی | شاخص‌ها |
| ۶ | ۲ | ۱ | ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | سوال // حداقل نمره |
| | | | | | | | | ✓ | ۱ داده‌ها و اطلاعات برنامه را تعیین کنید. |
| | | | | | | | | ✓ | ۲ ورودی و خروجی برنامه را مشخص کنید. |
| | | | | | | ✓ | ✓ | | ۳ راه حل خود را برای حل مسئله بنویسید. راه حل را بر اساس نیاز توسعه دهید. |
| | | | | | ✓ | | | | ۴ اجزای حلقه را معین کنید. |
| | | | ✓ | ✓ | | | | | ۵ کد برنامه را بنویسید. خطاهای احتمالی را برطرف کنید. |
| | | ✓ | | | | | | | ۶ تعداد حلقه‌های برنامه را تعیین کنید. |
| ✓ | ✓ | | | | | | | | ۷ کد برنامه را بنویسید و برنامه را گام به گام اجرا و خطاهای احتمالی را برطرف کنید. |

| فهرست نمره واحد یادگیری ۳ براساس ارزش دهی ۱-۲-۳ (مخصوص هنرآموز) | | | | | | | | | | |
|---|--|-------------------------------|--------------|---------------------------------------|------------------------------|---------------|---------------------------------|---------------------------|-------------------------|----------------------------------|
| نمره نهایی | ایجاد برنامه با حلقه‌های متداخل (۳ مرحله بحرانی) | | | ایجاد برنامه با حلقه (۳ مرحله بحرانی) | | | حل مسئله تکرار (۳ مرحله بحرانی) | | | مراحل و حداقل نمره |
| | | تعمین تعداد حلقه‌های یک مسئله | نوشتن برنامه | خطاها | خطا پای و کنترل اجرای برنامه | نوشتن کد حلقه | تعمین اجرای ساختار حلقه | توسعه راه حل در صورت نیاز | ارائه راه حل برای مسئله | |
| | ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | نام و نام خانوادگی // حداقل نمره |
| | | | | | | | | | | ۱ |
| | | | | | | | | | | ۲ |

ارزشیابی واحد یادگیری ۴

راهنمای پاسخ به سوالات: هنرجوی گرامی برای پاسخگویی به سوالات به ترتیب زیر عمل کنید.

- در بخش الف به سوالات ۱ و ۲ و ۳ پاسخ دهید.
- در بخش ب به سوالات ۴ و ۵ پاسخ دهید.
- در بخش ج به سوالات ۶ و ۷ پاسخ دهید.
- با مراجعه به جدول ارزش‌دهی عملکرد می‌توانید سطح شایستگی خود را بسنجید.
- ✓ مسائل :

الف- مادر علی برای نگهداری خرید حبوبات از علی می‌خواهد که جدولی از اسامی حبوبات برای او بنویسد و در آشپزخانه بچسباند.

ب - هنرآموز کلاس موسیقی اسامی هنرجویان و نام موسیقی مورد علاقه آنها را در یک فهرست ثبت کرده است.

برنامه‌ای بنویسید که این فهرست را بازسازی کرده و عملیات خواسته‌شده را بر اساس شماره سوال ۴ و ۵ به انجام برساند.

ج- هنرآموز کلاس موسیقی برای دسته‌بندی هنرجویان بر اساس نوع دستگاهی که یاد می‌گیرند احتیاج به مرتب کردن اسامی بر اساس نوع موسیقی دارد. همچنین در برخی از روزها هنرآموز برای ارسال گزارش نوع دستگاه خاصی احتیاج به جست‌وجوی داده‌ها خواهد داشت.

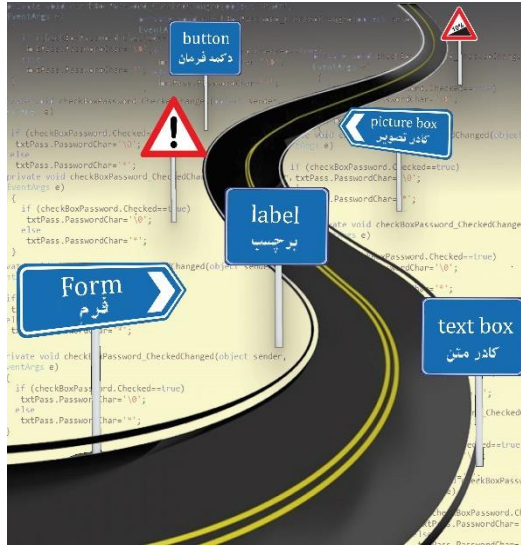
| ردیف | سوالات |
|------|--|
| ۱ | به نظر شما علی چه نوع آرایه و چه اندازه‌ایی برای این کار تعریف کرده است. |
| ۲ | آرایه علی را تعریف کنید. |
| ۳ | حبوبات آشپزخانه مادر علی را برایش وارد کنید. |
| ۴ | عناصر آرایه اسامی و آرایه نام موسیقی مورد علاقه را نمایش بدهد. |
| ۵ | بابک به هنرآموز خود می‌گوید که نام موسیقی مورد علاقه‌اش را از گیتار به سنتور تغییر دهد. کدهای لازم را بنویسید. |
| ۶ | کد برنامه لازم را برای مرتب کردن آرایه‌های کلاس موسیقی بنویسید. |
| ۷ | کد برنامه لازم را برای جست‌وجوی هنرجویانی که نوع دستگاه خاصی را آموزش می‌بینند بنویسید. |

جدول ارزش دهی عملکرد

| جست و جو در آرایه | | دسترسی به عناصر آرایه | | | اعلان آرایه (مرحله بحرانی) | | | مراحل (حداقل نمره) | | |
|--|--------------------|---|----------------------------------|----------------|---|------------------------------|----------------------------------|--------------------|--|------|
| اجرای گام به گام برنامه - رفع خطاها | کد برنامه جست و جو | کد برنامه مرتب سازی و معکوس کردن آرایه | خطا بائی و کنترل اجرای برنامه | نوشتن کد آرایه | تعیین ساختار آرایه و نمایش عناصر آرایه | رفع خطاهای احتمالی برنامه | تعریف آرایه و مقدار دهی اولیه | تعیین نوع آرایه | شاخص ها | |
| ۶ | ۶ | ۱ | ۶ | ۶ | ۱ | ۶ | ۶ | ۱ | سوال // حداقل نمره | ردیف |
| | | | | | | | | ✓ | به نظر شما علی چه نوع آرایه و چه اندازه‌هایی برای این کار تعریف کرده است؟ | ۱ |
| | | | | | | | ✓ | ✓ | آرایه علی را تعریف کنید. | ۲ |
| | | | | | | ✓ | ✓ | ✓ | حبوبات آشپزخانه مادر علی را برایش وارد کنید. | ۳ |
| | | | | | ✓ | | | | عناصر آرایه اسامی و آرایه نام موسیقی مورد علاقه را نمایش بدهد. | ۴ |
| | | | ✓ | ✓ | | | | | بابک به هنرآموز خود می‌گوید که نام موسیقی مورد علاقه‌اش را از گیتار به سنتور تغییر دهد. کدهای لازم را بنویسید. | ۵ |
| | | ✓ | | | | | | | کد برنامه لازم را برای مرتب کردن آرایه‌های کلاس موسیقی بنویسید. | ۶ |
| ✓ | ✓ | | | | | | | | کد برنامه لازم را برای جست‌وجوی هنرجویانی که نوع دستگاه خاصی را آموزش می‌بینند بنویسید. | ۷ |

فهرست نمره واحد یادگیری ۴ براساس ارزش‌دهی ۱-۲-۳ (مخصوص هنرآموز)

| نمره پایه‌ی | جست‌وجو در آرایه | | | دسترسی به عناصر آرایه | | | اعلان آرایه (مرحله بحرانی) | | | مراحل و حداقل نمره | |
|-------------|------------------|---|---|-----------------------|---|---|----------------------------|---|---|--------------------|----------------------------------|
| | | ۲ | ۱ | ۱ | ۲ | ۱ | ۱ | ۲ | ۱ | ۱ | نام و نام خانوادگی // حداقل نمره |
| | ۲ | ۱ | ۱ | ۲ | ۱ | ۱ | ۲ | ۱ | ۱ | | ۱ |
| | | | | | | | | | | | ۲ |



پودمان سوم

واحد یادگیری ۵ و ۶

طراحی واسط گرافیکی

واحد یادگیری ۵

شایستگی ایجاد واسط گرافیکی کاربر

مقدمات تدریس

الف) مفاهیم کلیدی

| مفاهیم کلیدی | | | |
|-------------------------|--------------------------------|--------------------|------------------------------------|
| جعبه ابزار (Toolbox) | کنترل فعال | کنترل (Control) | فرم (Form) |
| متد (Method) | رویداد (Event) | کشیدن (Stretch) | ویژگی (Property) |
| فیلتر (Filter) | کادر محاوره‌ای (Dialog Box) | منبع (Resource) | واکنش به رویداد (Event Handler) |
| | | | پروژه ویندوزی |

ب) تجهیزات لازم

مشخصات سخت‌افزاری برای نصب نرم‌افزار Visual Studio Express 2012

- پردازنده حداقل ۱/۶ گیگاهرتز
- حافظه RAM حداقل ۱ گیگابایت
- حداقل فضای موجود در دیسک سخت ۴ گیگابایت
- کارت ویدئویی متناسب با DirectX9 و قابل اجرا در رزولوشن ۱۰۲۴ × ۷۶۸ و بالاتر

ج) بودجه بندی

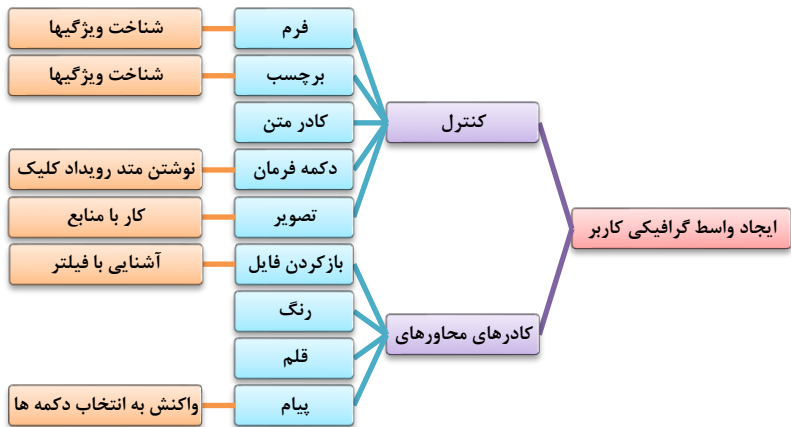
| جلسه | واحد یادگیری | کارگاه (موضوع) | شماره صفحات | اهداف توانمندسازی | فعالیت های تکمیلی |
|------|--------------|---|-------------|---|--|
| ۱۵ | ۵ | آشنایی با پروژه های ویندوزی | ۱۰۵ تا ۱۰۱ | آشنایی با پروژه ویندوزی - تعریف واسط گرافیکی کاربر - شناخت کنترل - ایجاد پروژه ویندوزی ساده - تنظیم مشخصات فرم | آشنایی بیشتر با IDE و .Net Framework و صفحه آغازین VS و پنجره Save و Open Project |
| ۱۶ | ۵ | کنترل های برجسب (Label)، کادر متن (Text Box) و دکمه فرمان (Button) و واکنش برنامه به رویدادها (Event Handler) | ۱۱۳ تا ۱۰۶ | آشنایی با کنترل های برجسب، کادر متن و دکمه فرمان و توانایی افزودن آن ها به فرم - شناخت ویژگی های هر کدام - آشنایی با تحقق و تشخیص و پردازش رویداد - توانایی نوشتن کد رویداد Click | آشنایی با LinkLabel و MaskedTextBox |
| ۱۷ | ۵ | منابع (Resources) کادر کنترل تصویر (Picture Box) | ۱۲۰ تا ۱۱۴ | توانایی کار با پوشه منابع - شناخت کادر کنترل تصویر و خصوصیات آن - توانایی استفاده از کادر تصویر | افزودن پرونده های موجود مثل تصویر و افزودن رشته و ایجاد پرونده هایی مثل Icon و TextFile در پوشه منابع |
| ۱۸ | ۵ | کادرهای محاوره ای (Dialog Box) کادر پیام (Message Box) | ۱۲۶ تا ۱۲۰ | توانایی استفاده از کادر محاوره ای در برنامه (انتخاب رنگ، انتخاب قلم، باز کردن پرونده) - معرفی کادر محاوره ای پیام و استفاده از آن در برنامه - به کار بردن دکمه های پنجره پیام و تصمیم گیری بر اساس نوع پاسخ کاربر | آشنایی با SaveFileDialog و FolderBrowserDialog و ویژگی های هر کدام |
| | ۶ | کامپوننت زمان سنج (Timer) | ۱۳۲ تا ۱۲۹ | آشنایی با تایمر و استفاده از آن و کد نویسی رویداد آن | ساخت یک ساعت دیجیتال ساخت کرنومتر حرکت یک جمله در پایین فرم |
| ۱۹ | ۶ | دکمه رادیویی (Radio Button) و کادر انتخاب (Check Box) و دسته بندی کنترل ها | ۱۳۵ تا ۱۳۳ | آشنایی با دکمه رادیویی و کادر انتخاب و به کارگیری آن ها - آشنایی با کنترل های گروه بندی (Group Box) و پنل (Panel) و به کارگیری آن ها | آشنایی با FlowLayoutPanel و TableLayoutPanel و SplitContainer |
| ۲۰ | ۶ | تولید اعداد تصادفی و کنترل سربرگ (Tab Control) | ۱۴۱ تا ۱۳۶ | آشنایی با نحوه تولید اعداد تصادفی - توانایی ایجاد پروژه ویندوزی با استفاده از کنترل های گروه بندی، دکمه های رادیویی، کادرهای انتخاب | شناخت کامل کلاس Random و متدهای آن و استفاده از آن ها |
| ۲۱ | ۶ | پروژه های چند فرم، و کادر لیست (List Box) و کادر ترکیبی (Combo Box) | ۱۵۰ تا ۱۴۲ | آشنایی با پروژه های چند فرم، و ایجاد آن - انتقال اطلاعات بین فرم ها - آشنایی با کادر لیست و کادر ترکیبی و استفاده از آن ها در پروژه | آشنایی با فرم های آماده و استفاده و الگو گیری از آن ها - آشنایی با شیوه های انتخاب کادر لیست و کادر ترکیبی، و استفاده از داده های انتخاب شده |

| طرح درس روزانه (هفتگی) پیشنهادی | | | |
|--|--|--|-------------|
| پایه : یازدهم | | درس: آشنایی با پروژه‌های ویندوزی | |
| پیام جلسه (هدف کلی): چرا نیاز به ساخت پروژه‌های ویندوزی داریم و چطور می‌توان آن را ساخت. | | | |
| اهداف یادگیری | فعالیت‌ها | زمان فیزیکی | |
| فعالیت | طبقه هدف: حیطه عاطفی / شناختی / روانی - حرکتی | کار هنرآموز | مدت (دقیقه) |
| ارزشیابی رفتار ورودی | سنجش میزان آگاهی هنرجویان از پروژه‌های ویندوزی | معطوف کردن توجه هنرجویان به تفاوت پروژه کنسولی و پروژه فرم ویندوزی و طرح سؤال: چه مواقعی لازم است پروژه کنسولی بنویسیم؟ | ۳۰ |
| ایجاد انگیزه | ایجاد توجه و تمرکز برای ورود به بحث آموزش ایجاد پروژه‌های ویندوزی | برنامه خوش آمدگویی به محیط C# با قلم و رنگ و ظواهر جذاب | ۴۰ |
| توضیح هنرآموز | توضیح کامل مفاهیم کلیدی (دانشی) و ایجاد علاقه و انگیزه در هنرجویان (بینشی) | توضیح مختصر در مورد شیوه ورود و ایجاد و اجرا و ذخیره پروژه | ۴۰ |
| تمرین هنرجویان | هنرجو باید بخش‌های مختلف Visual Studio را مشاهده و از آن برای ساخت یک پروژه ساده بهره گیرد. | هنرآموز توضیح کلی در مورد فعالیت کارگاهی به هنرجو داده و از آن‌ها می‌خواهد به‌صورت گروهی به حل آن‌ها بپردازند. | ۳۰ |
| ارائه تمرین | ایجاد یک پروژه ساده خوش آمدگویی با تنظیم ویژگی‌های فرم از قبیل قلم و رنگ و تصویر زمینه و ... | از هنرجویان خواسته می‌شود در گروه‌های تعیین شده به انجام فعالیت کارگاهی بپردازند. قبلاً ملاک‌های ارزیابی تمرین‌ها داده می‌شود. | ۹۰ |
| جمع‌بندی | هنرجو باید با مفاهیم پروژه فرم ویندوزی و نحوه ساخت آن کاملاً آشنا شده باشد. | مطالب مرور و در مورد درس جلسه آینده (کنترل‌های برجسب و کادر متن و دکمه) مختصراً به هنرجو اطلاعاتی ارائه می‌شود. | ۲۰ |
| ارزشیابی پایانی | هنرجو بتواند یک پروژه فرم ویندوزی بسازد | این ارزشیابی به‌صورت گروهی انجام می‌شود. | ۹۰ |
| تعیین تکلیف | انجام پروژه نسبتاً کامل‌تری با استفاده از ویژگی‌های دیگر فرم به‌صورت گروهی | توضیح خواسته‌های هنرآموز از هنرجویان به‌صورت شفاهی یا تایپ‌شده | ۲۰ |
| | | کار هنرجویان | |
| | | مشارکت در پاسخگویی و تعامل | |
| | | ایجاد پروژه مشابه | |
| | | هنرجویان ضمن توجه به صفحه‌نمایش، مطالب را می‌آموزند. | |
| | | با توجه به فعالیت‌های کارگاهی کتاب هنرجویان در گروه‌های خود، گزینه‌های دیگر را با آزمون و خطا می‌آموزند. | |
| | | هنرجویان در گروه‌های خود سعی می‌کنند با دقت، سرعت و خلاقیت به انجام فعالیت کارگاهی بپردازند. | |
| | | گوش‌دادن فعال و پرسش و پاسخ، انجام فعالیت‌های گروهی و برنامه‌ریزی به‌منظور انجام پژوهش | |
| | | هنرجویان به سوالات مطروحه پاسخ می‌دهند. | |
| | | تقسیم‌بندی وظایف کار پروژه بین اعضای گروه و تحویل در زمان‌بندی تعیین‌شده | |

د) ورود به بحث

هدف از این واحد شایستگی، چگونگی کار با اجزای IDE (محیط توسعه یکپارچه)، برنامه‌نویسی در پروژه ویندوزی و کنترل‌های اولیه و ایجاد واسط کاربری ساده است. با استفاده از اجزای محیط IDE برنامه‌نویسی و براساس دانش کسب شده، یک پروژه ساده ویندوزی ایجاد کرده، با درج کنترل‌های اولیه در فرم، واسط کاربری مناسب ایجاد و کد رویدادهای آن را بنویسد.

نمودار زیر نقشه مفهومی و ذهنی این واحد یادگیری را نمایش می‌دهد و شما می‌توانید با داشتن یک دورنما از مطالب و محتوای آموزشی، تدریس خود را به شکل بهتری مدیریت کنید.



با توجه به اینکه هنرجویان قبلاً با محیط Visual Studio کار کرده‌اند، و از طرفی جذابیت واسط گرافیکی، به‌احتمال زیاد، پروژه‌های از نوع Windows Form Application ایجاد کرده‌اند. می‌توانید با طرح سؤالاتی، میزان آشنایی آنان را سنجیده و انگیزه لازم را ایجاد کنید. مثلاً:

- تفاوت پروژه‌های فرم ویندوزی با پروژه‌های کنسول چیست؟
- کدام‌یک از شما پروژه فرم ویندوزی نوشته‌اید؟ چه برنامه‌ای؟
- چرا کاربران تمایل بیشتری برای کار با برنامه‌های فرم دارند؟

- چه ضرورتی برای نوشتن پروژه کنسولی وجود دارد؟
- از چه روشی برای دریافت نام کاربری و گذرواژه در برنامه استفاده کنیم؟
- برای تغییر شکل ظاهری برنامه، چه کارهایی می‌توان انجام داد؟
- بازی‌های ویندوزی چگونه طراحی می‌شوند؟
- سپس به برخی از دلایل برنامه‌نویسی کنسولی اشاره کنید. از قبیل:
 - یادگیری و درک عمیق مفاهیم برنامه‌نویسی.
 - عدم وجود سیستم‌عامل ویندوز روی همه سیستم‌های کامپیوتری مثل دستگاه‌های POS.
 - عدم امکان اجرای برنامه‌های گرافیکی روی برخی از سیستم‌ها مثل کنسول دستگاه CNC.

تدریس

واسط گرافیکی کاربر

واسط گرافیکی کاربر یا Graphical User Interface (به اختصار GUI که معمولاً GOO-ee یا Goopy یا جی‌یو‌ای تلفظ می‌شود)، نوعی واسط است که به کاربران اجازه می‌دهد، از طریق آیکن‌های گرافیکی و نشانگرهای بصری، با دستگاه‌های الکترونیکی تعامل و ارتباط برقرار کنند.

GUI ها، در مقابل نوعی دیگری از واسط‌های کاربری به نام Command-Line Interface یا CLI (که در آن‌ها دستورات و داده‌ها از طریق صفحه کلید در یک محیط متنی تایپ می‌شود)، معمولاً از ترکیبی از پنجره‌ها، آیکن‌ها، منوها و دستگاه‌های اشاره‌گر نظیر ماوس استفاده می‌کنند. به این ترتیب با ظهور GUI، مشکلات ناشی از تایپ نادرست دستورات، زمان‌بر بودن و نیاز به دقت بیشتر در CLI ها برطرف شد؛ به علاوه به جذابیت و رنگ و لعاب محیط‌های کاربری رایانه‌ها نیز افزوده شد.

امروزه استفاده از GUI ها نه تنها در رایانه‌های شخصی، بلکه در گوشی‌های هوشمند، تبلت‌ها، پخش‌کننده‌های موسیقی، کتاب‌خوان‌ها و ... نیز گسترش پیدا کرده است.

در اولین فرصت می‌توانید شیوه برگرداندن چینش محیط IDE را به فرم پیش‌فرض شرکت مایکروسافت، آموزش دهید و خیال هنرجو را راحت کنید و از او بخواهید، چیدمان محیط را به فرم خوشایند خود، تغییر دهد و نگران به‌هم‌ریختگی، نباشد.

Window → Reset Window Layout

برای مقایسه بهتر برنامه‌های کنسولی و برنامه‌های ویندوزی، مثال زیر را در نظر بگیرید:
فرض کنید در منزل هستید و قرار است برای شما مهمان بیاید. اگر بخواهید به‌موقع درب منزل را برای وی باز کنید می‌توانید به یکی از روش‌های زیر عمل کنید:



۱) می‌توانید کارهای دیگر خود را رها کنید و در جلوی در منزل خود، منتظر شوید تا وی بیاید.

۲) هرچند لحظه یک‌بار، به سمت در منزل بروید و بیرون را نگاه کنید که آیا مهمان شما آمده است.

۳) می‌توانید در داخل منزل مشغول انجام کارهای خود باشید، هر وقت که مهمان آمد، زنگ زد و شما را از ورودش مطلع کرد، به استقبال وی بروید و در را برایش باز کنید.

در برنامه‌های کنسولی، در دستوراتی مانند دستور دریافت داده، اجرای برنامه متوقف می‌شود و تا زمانی که کلید Enter زده شود دستورات بعدی اجرا نمی‌شوند. پس از اینکه کاربر اطلاعات را وارد کرد و کلید Enter را زد، خطوط بعدی برنامه به ترتیب اجرا می‌شوند. در مقایسه با مثال بالا می‌توان گفت در برنامه‌های کنسولی از روشی مانند روش ۱ و یا روش ۲ استفاده می‌شود. در برنامه‌های ویندوزی، از روش ۳ استفاده می‌شود.

کدام یک از زبان‌های برنامه‌نویسی بیشترین متقاضی را دارد؟

همانگونه که در جدول پایین می‌بینید، زبان‌های جاوا، جاوا اسکریپت و خانواده C (C, C#, Objective-C) بالاترین رتبه را در بین ۱۰ تا از محبوب‌ترین یا عمومی‌ترین زبان‌های برنامه‌نویسی در سال‌های ۲۰۱۶ و ۲۰۱۷ دارند. این زبان‌ها بستر بسیاری از برنامه‌های کاربردی موردنیاز هستند.

| Rank | TIOBE | IEEE | Stack Overflow | PYPL Index |
|------|----------------------|------------|----------------|-------------|
| 1 | Java | C | JavaScript | Java |
| 2 | C | Java | SQL | Python |
| 3 | C++ | Python | Java | PHP |
| 4 | C# | C++ | C# | C# |
| 5 | Python | R | PHP | JavaScript |
| 6 | PHP | C# | Python | C |
| 7 | JavaScript | PHP | C++ | C++ |
| 8 | Visual Basic NET | JavaScript | Angular JS | Objective-C |
| 9 | Delphi/Object Pascal | Ruby | Node.js | R |
| 10 | Perl | Go | C | Swift |

پاسخ به فعالیت‌ها

اجزاء IDE پروژه‌های ویندوزی

| ردیف | نام پنجره | کاربرد |
|------|-------------------|--|
| ۱ | Toolbox | جعبه ابزار متشکل از کنترل‌های گوناگون برای طراحی واسط گرافیکی کاربر. |
| ۲ | Solution Explorer | فهرستی از پرونده‌های تشکیل دهنده پروژه را نمایش می‌دهد. |
| ۳ | Menu Bar | نوار منو شامل دستورات کار با محیط VS و اجرا و خطایابی برنامه‌ها و ... |
| ۴ | Form | یک پنجره یا کادر محاوره‌ای است که رابط کاربری برنامه را تشکیل می‌دهد. همچنین، فرم کنترلی است که کنترل‌های دیگر را روی آن می‌چینیم. |
| ۵ | Properties | لیست ویژگی‌ها (Properties) و رویدادهای (Events) کنترل انتخاب شده. |
| ۶ | Tool Bar | نوار ابزار شامل ابزارهایی برای اجرای دستورات پرکاربردتر مثل Save و Start |

فعالیت کارگاهی
ص ۱۰۲

ویژگی‌های فرم

ویژگی Anchor اشیاء: اگر بخواهیم با تغییر اندازه فرم، اندازه کنترل هم تغییر کند؛ یا به عبارتی فاصله لبه‌های کنترل از لبه‌های فرم ثابت بماند، از ویژگی Anchor استفاده می‌کنیم. این ویژگی می‌تواند مقادیر Top و Left و Right و Bottom و یا ترکیبی از آن‌ها را بپذیرد. هر یک از مقادیر باعث می‌شوند که فاصله کنترل از سمت مربوطه ثابت بماند. مثلاً اگر مقدار ویژگی Anchor کنترل جعبه تصویر برابر Top, Bottom, Left, Right باشد، فاصله کنترل از چهار طرف فرم، ثابت باقی می‌ماند، یعنی با تغییر اندازه فرم، اندازه تصویر هم تغییر می‌کند.

ویژگی AcceptButton فرم: اگر ویژگی AcceptButton فرم را برابر یک دکمه خاص قرار دهید، هنگام فشردن شدن Enter، متد رویداد کلیک دکمه مذکور اجرا خواهد شد.

ویژگی CancelButton فرم: اگر ویژگی CancelButton فرم را برابر یک دکمه خاص قرار دهید، هنگام فشردن دکمه ESC، متد رویداد کلیک دکمه مذکور اجرا خواهد شد.

ویژگی AutoSize فرم: اگر ویژگی AutoSize فرم را برابر true قرار دهید، بسته به محتوایش، تغییر اندازه پیدا خواهد کرد یا به عبارتی اندازه فرم در جهت راست و پایین کم یا زیاد خواهد شد. البته اگر مقدار AutoSizeMode فرم برابر GrowOnly باشد، فرم فقط بزرگ می‌شود و اگر برابر GrowAndShrink باشد، بزرگ و کوچک خواهد شد.

ویژگی HelpButton فرم: با انتساب مقدار true به این ویژگی، در صورتی که ویژگی MinimizeBox و MaximizeBox فرم false باشد، دکمه Help که با [?] مشخص می‌شود، در نوار عنوان نمایش داده می‌شود. در این حال اگر یک کنترل helpProvider بانام HP به فرم اضافه کنید و در ویژگی HelpNamespace نام یک فایل از نوع .chm، col، htm و یا html را قرار دهید، می‌توانید از آن استفاده کنید. سه ویژگی HelpKeyword on HP و HelpNavigator on HP و HelpString on HP که به فرم و کنترل‌های روی آن افزوده می‌شود، برای همین منظور است.

برای ساخت فایل chm می‌توانید از نرم‌افزار WinCHM Pro که به صورت رایگان و به راحتی قابل دانلود است، استفاده کنید.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۱۰۳

از پنجره ویژگی‌های فرم، ویژگی `BackgroundImage` را انتخاب کنید. با کلیک روی نماد و دکمه `Import` تصویر موردنظر را انتخاب کنید. مقدار ویژگی `BackgroundImageLayout` را روی گزینه `Stretch` قرار دهید. مقادیر دیگر این ویژگی را بررسی کرده، نتیجه را در جدول ۲ بنویسید.

| مقدار | نتیجه |
|----------------|--|
| Tile | تصویر زمینه به صورت کاشی‌کاری، کل فرم را پر می‌کند. |
| Center | تصویر زمینه با اندازه اصلی، در وسط فرم نمایش داده می‌شود. |
| Stretch | تصویر زمینه در جهت طولی و عرضی به اندازه فرم، کشیده یا جمع می‌شود. |
| Zoom | تصویر به نسبت یکسان در جهت طولی و عرض تغییر اندازه می‌دهد تا طول یا عرض فرم را پر کند. |

- ویژگی `FormBorderStyle` فرم را با مقادیر مختلف آزمایش کنید. مقادیر این ویژگی را بررسی کرده، نتیجه را در جدول ۳ بنویسید.

| مقدار | نتیجه |
|--------------------------|--|
| FixedSingle | حاشیه تک‌خطی، که اجازه تغییر اندازه فرم را نمی‌دهد. |
| Fixed3D | حاشیه سه‌بعدی |
| FixedDialog | حاشیه نازک به سبک کادر محاوره‌ای (آیکن فرم در نوار عنوان ظاهر نمی‌شود. آیکن فرم منویی شامل <code>Restore</code> و <code>Move</code> و <code>Size</code> و <code>Min</code> و <code>Max</code> و <code>Close</code> را در اختیار می‌گذارد.) |
| Sizable | فرم با حاشیه قابل تغییر اندازه |
| FixedToolWindow | حاشیه غیرقابل تغییر اندازه که در نوار عنوان فقط دکمه <code>Close</code> دارد. اگر <code>ShowInTaskbar=false</code> باشد، هنگام استفاده از <code>Alt+Tab</code> ، فرم در لیست دیده نخواهد شد. |
| SizableToolWindow | حاشیه قابل تغییر اندازه که در نوار عنوان فقط دکمه <code>Close</code> دارد. اگر <code>ShowInTaskbar=false</code> باشد، هنگام استفاده از <code>Alt+Tab</code> ، فرم در لیست دیده نخواهد شد. |

با کمک هم گروهی خود جدول ۴ را کامل کنید.

| عملکرد | ویژگی های فرم |
|---|----------------------|
| نوع قلم و اندازه آن | Font |
| اندازه طول و عرض فرم | Size (Width, Height) |
| با مقدار True در فاز طراحی پروژه، امکان تغییر اندازه و جابجایی را نمی دهد. | Locked |
| عنوان فرم | Text |
| موقعیت فرم در اولین نمایش مثلاً: وسط صفحه نمایش (CenterScreen) | StartPosition |
| آیکن فرم در گوشه سمت چپ عنوان و هنگام Minimize | Icon |
| میزان کدری یا تاری فرم، عکس شفافیت (Transparency) | Opacity |
| با مقدار False در نوار عنوان فقط عنوان نمایش داده می شود و آیکن و دکمه های Close و Minimize و Maximize محو می شوند. | ControlBox |

کارگاه ۳- استفاده از کادر متن و دکمه در واسط کاربری H

کنترل های برجسته (Label)، کادر متن (TextBox) و دکمه فرمان (Button):

فارسی زبانان برای اینکه نیاز نباشد ویژگی RightToLeft و Font را برای تک تک کنترل ها تنظیم کنند، می توانند این ویژگی ها را برای Form تنظیم کنند. همه اشیاء، این ویژگی را از والد (Parent) خود به ارث می برند.

مقدار Autosize برجسته، True در نظر گرفته شده است. اگر می خواهید اندازه آن را هنگام طراحی تغییر دهید، مقدار آن را False کنید.

نوع و اندازه قلم و رنگ عنوان فرم را نمی توان تغییر داد. برای این منظور می توانید:

۱) ویژگی FormBorderStyle فرم را None کنید یا ControlBox=False و مقدار Text را خالی کنید.

۲) یک برجسب با مقادیر Dock=Top و AutoSize=False و TextAlign=MiddleCenter و رنگ و متن و قلم دلخواه روی فرم قرار دهید.

از کادر متن می‌توان برای نمایش و ورود اطلاعات استفاده کرد. در صورتی که برای دریافت رمز استفاده می‌کنید، می‌توانید ویژگی PasswordChar آن را * قرار دهید. در حالت کد نویسی برای خنثی کردن حالت رمز، از این دستور استفاده می‌شود: `textBox1.PasswordChar = '\0';`

یادآوری: کاراکتر null به صورت '\0' تعریف می‌شود.

برای دریافت یا نمایش مقادیری از نوع شماره تلفن، تاریخ، ساعت و کد پستی و ... می‌توانید از نوع TextBox به نام MaskedTextBox استفاده کنید.

برای متنی که در آن قلم‌های مختلف با رنگ و اندازه‌های متفاوت و امکانات بسیار زیاد دیگر، می‌توانید از RichTextBox استفاده کنید.

اگر ویژگی FlatStyle دکمه را Flat و ویژگی FlatAppearance.BorderSize را صفر قرار دهید، دکمه شبیه برجسب خواهد شد که در بعضی از طراحی‌ها استفاده می‌شود.

پاسخ به فعالیت‌ها

ویژگی Text، متنی که زمان اجرا در کادر متن وارد می‌شود را نگهداری می‌کند. مقدار پیش‌فرض این ویژگی چیست؟ مقدار ویژگی Text کادر متن را تغییر دهید و با اجرای برنامه نتیجه را بررسی کنید. آیا کادر متن فقط برای دریافت اطلاعات به کار می‌رود؟
پاسخ: پیش‌فرض، رشته خالی است (""). از کادر متن برای نمایش اطلاعات هم می‌توان استفاده کرد.

کنجکاوی
ص ۱۰۷

آیا می‌توان بیش از یک نویسه را در ویژگی PasswordChar قرار داد؟ نوع داده این ویژگی چیست؟
پاسخ: خیر. char

کنجکاوی
ص ۱۰۷

کنجکاوی
ص ۱۰۸

آیا می‌توان یک ویژگی چند کنترل را هم‌زمان تغییر داد؟

پاسخ: بلی. برخی از ویژگی‌های مشترک و هم‌نوع کنترل‌های انتخاب شده مثل Font و BackColor و ForeColor و Text. برخی از ویژگی‌های مشترک مثل Name را چون باید منحصر به فرد باشد، نمی‌توان هم‌زمان تغییر داد.

فعالیت گروهی
ص ۱۰۸

با کمک هم گروهی خود، جدول ویژگی‌های کنترل‌ها را تکمیل کنید.

| نام ویژگی | عملکرد | کنترل |
|-------------|--|---------------------------|
| Name | نام کنترل | همه کنترل‌ها |
| BackColor | رنگ زمینه | TextBox-Button-Form-Label |
| ForeColor | رنگ قلم | TextBox-Button-Form-Label |
| Text | متن یا عنوان | TextBox-Button-Form-Label |
| Enabled | فعال | TextBox-Button-Form-Label |
| Visible | قابل رؤیت | TextBox-Button-Label |
| AutoSize | تغییر اندازه خودکار | Button-Form-Label |
| TextAlign | تراز بندی متن | TextBox-Button-Label |
| Size | اندازه | TextBox-Button-Form-Label |
| Location | موقعیت | TextBox-Button-Form-Label |
| Cursor | مکان‌نما | TextBox-Button-Form-Label |
| Dock | پهلوی گرفتن (چسباندن کنترل به یکی از لبه‌ها) | TextBox-Button-Label |
| RightToLeft | راست به چپ | TextBox-Button-Form-Label |

واکنش برنامه به رویدادها

پاسخ به فعالیت‌ها

فعالیت کارگاهی
ص ۱۱۵

هنگام اضافه کردن پرونده‌های تصویر به پروژه (شکل ۱۴) گزینه New Image را انتخاب کنید، چه اتفاقی می‌افتد؟ کاربرد این گزینه در چه مواقعی است؟

پاسخ: می‌توان پرونده‌های تصویری از نوع PNG و BMP و GIF و JPEG و TIFF و آیکن، توسط محیط طراحی گرافیکی (Graphics Designers) و

یا ویرایشگر منبع (Resource Editor) ایجاد و به منابع پروژه افزود. اگر بخواهیم خودمان تصویر را تولید کنیم و مخصوصاً اگر نرم‌افزار ایجاد تصویر، نداشته باشیم، این ابزار کاراست.

با استفاده از گزینه های Local resource و Project resource file تصویر دلخواه را در کادر تصویر قرار دهید و تفاوت این دو گزینه را بنویسید. پاسخ: هنگام استفاده از Local resource با استفاده از دکمه Import می‌توان تصویر را به کادر تصویر اختصاص داد و صرفاً در همان فرم قابل استفاده است و توسط دکمه Clear می‌توان تصویر اختصاص داده شده را حذف نمود. اما با دکمه Project resource file ابتدا فایل منبع به پوشه resource پروژه کپی می‌شود و آنگاه امکان اختصاص آن به کادر تصویر، فراهم است و در همه فرم‌های پروژه قابل استفاده است.

فعالیت گروهی
ص ۱۱۶

چرا یک نسخه از پرونده‌های تصویری را در پوشه Debug قرار دادیم؟ پاسخ: زیرا پرونده اجرایی پروژه در این پوشه است و مثلاً اگر پرونده تصویر Test.Jpg در این پوشه باشد، آنگاه:

```
picSample.ImageLocation = "Test.Jpg";
```

نیاز به نوشتن مسیر پرونده نیست. در این صورت از آدرس‌دهی نسبی استفاده شده است. اگر پوشه پروژه به مسیر دیگری منتقل شود و یا تغییر نام داده شود، مشکلی پیش نخواهد آمد.

کنجکاوی
ص ۱۱۹

کادرهای محاوره‌ای

پاسخ به فعالیت‌ها

اگر در کادر محاوره‌ای پرونده انتخاب نشود، چه چیزی در کادر تصویر نمایش داده خواهد شد؟

پاسخ: در ویژوال استودیو ۲۰۱۵، اگر پرونده‌ای انتخاب نکنیم، با کلیک روی دکمه Open، اتفاقی نمی‌افتد و منتظر انتخاب پرونده می‌ماند ولی اگر دکمه Cancel را کلیک کنیم، تصویر نمایش داده خواهد شد. اگر پرونده غیر تصویری هم انتخاب کنیم، باز هم همان تصویر نمایش داده خواهد شد.

سوال
ص ۱۲۲

فعالیت گروهی
ص ۱۲۴

با استفاده از راهنمای مایکروسافت و به کمک هم گروهی خود اعضای داده‌های شمارشی `MessageBoxIcon` و `MessageBoxButtons` و `MessageBoxDefaultButton` را تعیین کنید.

| اعضای داده شمارشی | داده شمارشی |
|--|--------------------------------------|
| Asterisk, Error, Exclamation, Hand, Information, None, Question, Stop, Warning | <code>MessageBoxIcon</code> |
| <code>Button1</code> , <code>Button2</code> , <code>Button3</code> | <code>MessageBoxDefaultButton</code> |
| <code>AbortRetryIgnore</code> , <code>OK</code> , <code>OKCancel</code> , <code>RetryCancel</code> , <code>YesNo</code> , <code>YesNoCancel</code> | <code>MessageBoxButtons</code> |
| <code>Abort</code> , <code>Cancel</code> , <code>Ignore</code> , <code>No</code> , <code>None</code> , <code>OK</code> , <code>Retry</code> , <code>Yes</code> | <code>DialogResult</code> |

فعالیت کارگاهی
ص ۱۲۵

با کمک هنرآموز خود تغییراتی در دستورات کارگاه ۹ ایجاد کنید که دو دکمه `Ok` و `Cancel` در کادر پیام نمایش داده شود. دکمه پیش فرض را دکمه `Cancel` تعیین کنید تصویر نماد کادر محاوره‌ای را تغییر دهید.

پاسخ:

```
MessageBox.Show("لطفاً یک پرونده تصویری را انتخاب کنید")  
MessageBoxButtons.OKCancel,  
MessageBoxIcon.Error, MessageBoxDefaultButton.Button2);
```

واحد یادگیری ۶

شایستگی کار با کنترل‌های پیشرفته

مقدمات تدریسی

الف) مفاهیم کلیدی

| مفاهیم کلیدی | | | |
|------------------------------|---------------------------|--------------------------------|----------------------------|
| دسته‌بندی کنترل‌ها | کادر علامت (Check Box) | دکمه رادیویی (Radio Button) | زمان‌سنج (Timer) |
| کنترل سربرگ (Tab Control) | اعداد تصادفی (Random) | پنل (Panel) | کادر گروهی (Group Box) |
| کادر فهرست (List Box) | فرم شروع | پروژه چند فرمی | صفحه سربرگ (Tab Page) |
| | | | کادر ترکیبی (Combo Box) |

ب) ورود به بحث

هدف از این واحد شایستگی چگونگی استفاده از کنترل‌های پیشرفته در پروژه‌های ویندوزی و برنامه‌های چند فرمی است. نمودار زیر نقشه مفهومی و ذهنی این واحد یادگیری را نمایش می‌دهد و شما می‌توانید با داشتن یک دورنما از مطالب و محتوای آموزشی، تدریس خود را به شکل بهتری مدیریت کنید.



تدریس

با استفاده از دانش کنترل‌های پیشرفته، کنترل‌های موردنیاز را به فرم اضافه و گروه‌بندی کرده، پروژه چند فرمی ایجاد کنید و برای رویدادهای آن‌ها برنامه بنویسید.

کامپوننت زمان سنج

مهم‌ترین ویژگی این کامپوننت Interval است که فاصله زمان تکرار دستور برحسب میلی‌ثانیه را تنظیم می‌کند. برای به کار انداختن زمان‌سنج لازم است ویژگی Enabled آن را True کرد.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۱۲۹

برای توقف خودرو متد رویداد کلیک دکمه btnStop را بنویسید.

```
timer1.Enabled = false;
```

- امکان کم‌وزیاد کردن سرعت خودرو را به برنامه اضافه کنید.

پاسخ: برای کم کردن سرعت خودرو می‌توان دکمه btnDecrease را با عنوان کاهش سرعت روی فرم گذاشت و با افزودن یک مقدار ثابت به ویژگی Interval کامپوننت زمان‌سنج، کد زیر را برای آن نوشت:

```
timer1.Interval += 200;
```

برای زیاد کردن سرعت خودرو می‌توان دکمه `btnIncrease` را با عنوان افزایش سرعت روی فرم گذاشت و با کاهش یک مقدار ثابت از ویژگی `Interval` کامپوننت `timer1`، کد زیر را برای آن نوشت

```
if (timer1.Interval > 200) timer1.Interval -= 200;
```

- برنامه را طوری تغییر دهید که در صورت برخورد خودرو به فرم هم، خودرو متوقف شود.

پاسخ: کد `timer1_Tick` اگر به صورت زیر نوشته شود، در صورت برخورد خودرو به فرم، باعث توقف آن می‌شود

```
if (picCar.Left + picCar.Width <= this.Width - 5) picCar.Left += 5;
```

و یا دقیق‌تر به صورت زیر

```
if (picCar.Left + picCar.Width <= this.ClientSize.Width - 5) picCar.Left += 5;
```

- برای حرکت کردن خودرو به سمت عقب نیز برنامه بنویسید.

پاسخ: برای حرکت دادن خودرو به سمت عقب، می‌توان یک متغیر `x` را به عنوان متغیر `timer1_Tick` استفاده نمود و برای عقب راندن خودرو، مقدار `x` را به `-5` تغییر داد. همچنین کد فوق را به صورت زیر تغییر داد.

```
if (picCar.Left + picCar.Width <= this.ClientSize.Width - x && picCar.Left > -x)
```

```
picCar.Left += x;
```

کنجکاوی
ص ۱۳۰

عملکرد متد timer1_Tick چیست؟ برنامه را اجرا کنید. چرا خودرو حرکت نمی‌کند؟

پاسخ: رویداد Tick تنها رویداد کامپوننت زمان‌سنج است. این رویداد در فاصله زمانی مشخص شده در ویژگی Interval رخ می‌دهد و دستورات متد این رویداد اجرا می‌شوند. مثلاً اگر Interval=2000 باشد، هر ۲ ثانیه یک‌بار رویداد Tick اتفاق افتاده و دستورات آن اجرا می‌گردند.

برای حرکت خودرو، بایستی ویژگی Enabled آن را True کرد. واضح است که مقدار ویژگی Interval، هم باید یک عدد بزرگتر از صفر باشد.

فعالیت کارگامی
ص ۱۳۱

ساعت سیستم را در برنامه خود نمایش دهید.



```
private void frmClock_Load(object sender, EventArgs e)
{
    timerClock.Interval = 1000;
    timerClock.Enabled = true;
}
private void timerClock_Tick(object sender, EventArgs e)
{
    lblClock.Text = DateTime.Now.ToLongTimeString();
}
```

کنترل‌های دکمه رادیویی (RadioButton) و کادر علامت (CheckBox)

تفاوت عمده این دو کنترل انتخاب، در این است که در یک مجموعه (گروه) دکمه رادیویی، باید یکی را انتخاب کرد (در هر لحظه فقط یک کانال رادیو را می‌توان انتخاب و پخش کرد و با انتخاب هریک دیگری از حالت انتخاب خارج می‌شود). ولی در یک مجموعه کادر انتخاب، می‌توان هیچ‌کدام یا هر تعداد یا همه را انتخاب کرد.

در حالت عادی، تمام دکمه‌های رادیویی روی فرم، یک مجموعه را تشکیل می‌دهند. می‌توان از کنترل‌های دسته‌بندی یا کنترل‌های گروهی (Containers) برای ایجاد مجموعه‌ها استفاده کرد.

Containers: GroupBox, Panel, TabControl, FlowLayoutPanel, SplitContainers, TableLayoutPanel.

زمانی که یک کنترل در داخل پنل یا کادر گروه‌بندی قرار می‌گیرد، آن کنترل به فرزند پنل یا کادر گروهی تبدیل می‌شود و این ارتباط بین والد و فرزند اجازه می‌دهد که مقادیر ویژگی‌های عمومی مثل `AutoSize` و `Font` و `ForeColor` و `Enabled`، به فرزندان به ارث برسد. برای مثال اگر مقدار ویژگی `Enabled` پنل را برابر با `False` قرار دهیم، نه تنها خود پنل بلکه تمامی کنترل‌های داخل آن نیز غیرفعال می‌شوند.

دسته‌بندی کنترل‌ها

از کلاس `Random` برای تولید اعداد تصادفی استفاده می‌شود.

```
Random rndGenerator = new Random();  
for (int i = 0; i < 5; i++)  
    label1.Text += rndGenerator.Next(10) + " ";
```

شیء `rndGenerator`، مولد اعداد تصادفی است. مبنای محاسبه اعداد تصادفی، زمان است. بنابراین هر بار که اجرا شود، اعداد تولید شده، با اعداد تولیدی در اجرای قبلی متفاوت خواهد بود. اگر از `new Random(n)` استفاده شود که در آن `n` یک عدد صحیح است، مبنای اولیه محاسبه عدد تصادفی، `n` است. در این صورت، هر بار اجرا، باعث تولید همان سری اعداد تولیدشده در مرحله قبل خواهد شد.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۱۳۷

جدول را به کمک راهنمای مایکروسافت و با توجه به شکل‌های مختلف متد Next کامل کنید.

[https://msdn.microsoft.com/en-us/library/system.random\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.random(v=vs.110).aspx)

| شرح | شکل |
|---|------------------------|
| عدد صحیح تصادفی بزرگ‌تر یا مساوی صفر | Next() |
| عدد صحیح تصادفی بزرگ‌تر یا مساوی صفر و کوچک‌تر از max | Next(int max) |
| عدد صحیح تصادفی بزرگ‌تر یا مساوی min و کوچک‌تر از max | Next(int min, int max) |

فعالیت گروهی
ص ۱۳۸

برنامه Tour را تغییر دهید تا به جای انتخاب کادر علامت با کلیک روی هر برجسب بخشی از تصویر ظاهر شود.
پاسخ: برای هر برجسب، در رویداد Click آن، ویژگی Visible آن را false کنید.
با کمک هم گروهی خود و هنرآموز جدول زیر را کامل کنید.

| نام کنترل | کاربرد | ویژگی‌های مهم | رویدادهای مهم |
|-------------------------------|---|-------------------------------|------------------|
| دکمه رادیویی (RadioButton) | برای انتخاب یکی از حالات ممکن در یک گروه | Checked CheckAlign Text | CheckedChanged() |
| کادر علامت (CheckBox) | برای انتخاب یا عدم انتخاب یک حالت | Checked CheckState Text | CheckedChanged() |
| کادر گروهی groupBox | جهت ایجاد یک گروه یا مجموعه از کنترل‌ها | Text Dock | |
| پنل (panel) | جهت ایجاد یک گروه یا مجموعه از کنترل‌ها بدون عنوان (Text) | Dock RightToLeft | |

تکمیل کارگاه
ص ۱۴۰

ویژگی RightToLeftLayout را True قرار دهید. چه تغییری اتفاق می‌افتد؟

پاسخ: این ویژگی وابسته به ویژگی RightToLeft است. اگر RightToLeft=true باشد، آنگاه ویژگی RightToLeft تمام اشیاء روی کنترل سربرج، true خواهد شد ولی اثری روی ظاهر خودکنترل سربرج نخواهد داشت. در این حال، اگر RightToLeftLayout=true شود، ظاهرکنترل سربرج، راست به چپ خواهد شد.

کارگاه ۶ - ایجاد پروژه‌های چند فرمی

در ویژوال استودیو فرم‌های از پیش طراحی شده زیادی وجود دارد که باعث تسریع در طراحی برنامه‌ها می‌شوند. یکی از این فرم‌ها، پنجره About است که با اضافه کردن آن به برنامه می‌توانید کادر About را نمایش دهید. Class، User Control، Bitmap File، Html Page و MDI Parent Form، نمونه‌های دیگری از این فرم‌ها هستند. این گونه فرم‌ها برای الگو گرفتن هم مفید هستند.

اشیای روی یک فرم به صورت خصوصی (Private) در نظر گرفته می‌شوند و خارج از فرم قابل دسترس نیستند. اختصاص مقدار Public به ویژگی Modifiers هر شیء، آن را خارج از فرم، قابل دسترس می‌کند.

در برنامه‌های چند فرمی، وقتی فرمی را Show می‌کنید، در حافظه اصلی قرار می‌گیرد و هنگامی که Close شود، از حافظه خارج می‌شود؛ ولی در برنامه‌هایی که از سربرگ استفاده می‌کنند، به محض اجرای برنامه، همه چیز در حافظه قرار می‌گیرد و تا هنگام باز بودن برنامه، در حافظه می‌مانند.

گاهی در شرایط خاصی مثلاً با ورود کد کاربری و کلمه عبور درست، امکان باز شدن یک فرم فراهم می‌شود.

در برنامه‌های چند فرمی، همزمان می‌توان محتوای چند فرم را در کنار هم دید.

پاسخ به فعالیت‌ها

مقدار اولیه فیلد succeeded و userName چیست؟

پاسخ: null و false

تکمیل کارگاه
ص ۱۴۲

عملکرد دستور (); new LoginForm frm = LoginForm چیست؟

پاسخ: LoginForm یک کلاس است. برای استفاده از آن، باید یک شیء ایجاد کرد (نمونه‌گیری Instance) و با استفاده از new آن شیء را ساخت.

کنجکاوی
ص ۱۴۵

- برای نمایش فرم می‌توان از متدهای Show یا ShowDialog استفاده کرد. به کمک هنرآموز خود تفاوت این دو متد را بررسی کنید.
پاسخ: در صورت استفاده از متد Show در حالی که فرم باز است، می‌توان روی فرم‌های دیگر هم رفت یا به عبارتی به آن‌ها سوئیچ کرد؛ ولی با استفاده از متد ShowDialog، تا هنگامی که فرم بسته نشود، امکان رفتن به فرم‌های دیگر

فعالیت کارگاهی
ص ۱۴۵

برنامه، وجود ندارد و اجرای ادامه برنامه تا بسته شدن فرم، به تعویق می‌افتد و فرم اصطلاحاً به صورت modal نمایش داده می‌شود.
 - آیا می‌توان در این برنامه به جای متد ShowDialog از متد Show استفاده کرد؟ چرا؟
 پاسخ: خیر. اگر از متد Show استفاده کنیم، پس از نمایش فرم لاگین، ادامه کد اجرا می‌شود و چون حاصل frm.succeeded برابر false است، this.Close() اجرا و فرم بسته و برنامه خاتمه خواهد یافت.

پژوهش
ص ۱۴۵

استفاده از سطح دسترسی public چه معایبی دارد؟ چه راهکارهایی برای انتقال اطلاعات بین فرم‌ها وجود دارد؟

پاسخ: فیلد با سطح دسترسی public، قابل خواندن و نوشتن توسط دیگران است. همچنین کنترلی بر مقداره‌ی آن نیست. از طرفی، اگر private تعریف شود، توسط دیگران، غیرقابل دسترس است. برای خواندن و نوشتن یک فیلد private می‌توان از دو متد استفاده کرد.

```
class Circle
{
    private float fltRadius;
    public void setRadius(float r)
    {
        if (r < 0) fltRadius = 0;
        else fltRadius = r;
    }
    public float getRadius()
    {
        return fltRadius;
    }
}
```

به جای فیلد private و دو متد فوق، می‌توان از property استفاده کرد:

```
class Circle
{
    private float fltRadius;
    public float Radius
    {
        get { return fltRadius; }
        set
        {
            if (value < 0) fltRadius = 0;
            else fltRadius = value;
        }
    }
}
```

در این صورت، آن خصوصیت قابل خواندن و نوشتن است. می توان با حذف متد `set`، خصوصیت را `ReadOnly` و با حذف متد `get`، خصوصیت را `WriteOnly` تعریف کرد. همچنین بر روی مقدار ورودی هم می توان کنترل داشت. فیلم مربوط به این پژوهش را می توانید در لوح نوری همراه کتاب مشاهده کنید.

کنترل های کادر فهرست (ListBox) و کادر ترکیبی (ComboBox)

کادر ترکیبی (Combo Box)، ترکیبی از کادر متن و کادر فهرست است.

It is a combination of a List Box and a single-line editable Text Box.

پاسخ به فعالیت ها

- با استفاده از پنجره ویژگی ها، تغییری در ویژگی های کادر ترکیبی ایجاد کنید که اسامی هنرجویان را به شکل مرتب نمایش دهد.

پاسخ: قرار دادن `true` در ویژگی `Sorted`. البته به صورت رشته ای مرتب می کند. مثلاً اگر اعداد ۱۰ و ۳۰ و ۲۰ و ۴ و ۲ را در آن قرار دهیم، اعداد به صورت 10,2,20,30,4 نمایش داده خواهند شد.

- دستوراتی به پروژه اضافه کنید تا شماره عنصر انتخاب شده در فهرست را در یک کادر متن نمایش دهد.

```
textBox1.Text = comboBox1.SelectedIndex.ToString();
```

فعالیت کارگامی
ص ۱۴۸

الف) فعالیت‌های تکمیلی

- ۱) با جستجو در اینترنت، چند زبان سطح بالای متداول امروزی را پیدا کرده و آن‌ها را نام ببرید.
- ۲) برنامه‌ای شامل سه کادر فهرست بنویسید که در اولی اسامی فصول سال قرار داشته باشد. با کلیک روی هر فصل در کادر دومی اسامی ماه‌های آن فصل نمایش داده شود. با کلیک روی هر ماه، شماره روزهای آن ماه در کادر سومی نمایش داده شود.
- ۳) فرم زیر را برای محاسبه معدل هنرجویان طراحی کنید. دکمه حذف زمانی فعال می‌شود که داده‌ای از فهرست مربوطه، انتخاب شده باشد. در صورتی که دکمه افزودن را کلیک کنید، داده مربوطه به فهرست اضافه و از کادر متن پاک می‌شود. اگر مقادیر نام درس و تعداد واحد و نمره آن درس را وارد کرده باشید، می‌توانید با کلیک روی دکمه "افزودن درس و واحد و نمره"، کار هر سه دکمه افزودن را یکجا انجام دهید. با کلیک روی دکمه "اعلام معدل"، ضمن محاسبه و اعلام معدل در یک پیام، از کاربر در مورد خالی کردن فهرست نمرات سؤال کنید؛ اگر پاسخ "بلی" داد، فهرست نمرات را برای محاسبه معدل هنرجوی بعدی خالی کنید.

| محاسبه معدل | | | |
|--------------------------------|----------------------|----------------------|---------------------|
| افزودن درس و واحد و نمره | نمره درس؛ | تعداد واحد؛ | نام درس؛ |
| | افزودن نمره به فهرست | افزودن واحد به فهرست | افزودن درس به فهرست |
| | | | |
| خالی کردن فهرست‌ها | | | |
| اعلام معدل | حذف نمره از فهرست | حذف واحد از فهرست | حذف درس از فهرست |

۴) هنگام مبادلات بانکی از طریق اینترنت، برای بالا بردن امنیت، تأکید می‌شود که از صفحه‌کلید مجازی استفاده کنید. برنامه‌ای بنویسید که فرم زیر را شبیه‌سازی کند. می‌توانید با استفاده از یک Panel حاوی کلیدهای ارقام و دکمه‌های "X" و "پاک کردن"، صفحه‌کلید مجازی را بسازید. هر دفعه که برنامه را اجرا می‌کنیم، باید چیدمان کلیدهای ارقام عوض شود.

شماره کارت: ۶۰۳۷-۹۹۷۲- [] []

....

پاک کردن X

CVV۲

سال

 /

ماه

رمز دوم کارت

CVV۲

تاریخ انقضا

اطلاعات کارت مقصد-دسته ای

اطلاعات کارت مقصد-تکی

1

به

شماره کارت مقصد

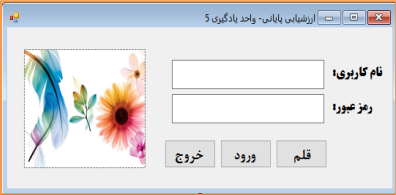
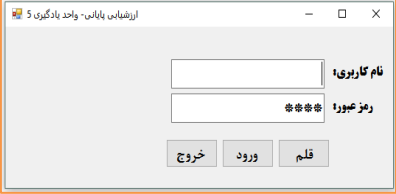
مبلغ کل

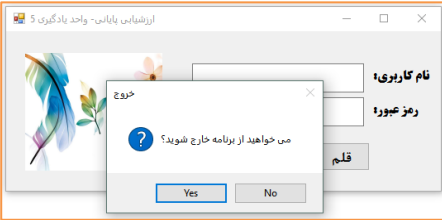
ریال

(ب) نمونه ارزشیابی پایانی

ارزشیابی پایانی واحد یادگیری ۵:

| استفاده از کادرهای محاوره‌ای | | ایجاد کد برای واکنش به رویدادها | ایجاد واسط کاربری بدون واکنش به رویدادها | ایجاد پروژه ویندوزی | مراحل کاری |
|---|--|--|--|---|--|
| ۲ | | ۲ | ۲ | ۱ | بحرانی |
| مدیریت برنامه بر اساس پاسخ کاربر به کادر محاوره‌ای و پیام نمایش کادر محاوره‌ای زمان اجرا را مشخصات تعیین شده) | | تنظیم و ویژگی های کنترل کادر تصویر- نوشتن کد تغییر مقدار ویژگی های کنترل با کدنویسی- افزودن تصویر به منابع پروژه- بستن فرم | مقداردهی یک ویژگی چند کنترل به صورت همزمان: تنظیم ویژگی های برچسب و کادر متن | ایجاد پروژه ویندوزی فعال و غیر فعال کردن اجزای محیط IDE تنظیم ویژگی های فرم | استاندارد (شاخص ها) |
| ۱ | | ۱ | ۱ | ۱ | سوالات/نمره شایستگی |
| درج کادر محاوره‌ای و تنظیم ویژگی های آن | | درج کنترل کادر تصویر- ایجاد متد رویداد- فعال کردن محیط کدنویسی | درج کنترل های برچسب- کادر متن و دکمه در فرم | ✓ | ۱ یک پروژه ویندوزی با نام خانوادگی خود در پوشه D:\Exam ایجاد نمایید. |
| ۲ | | ۲ | ۲ | ✓ | ۲ جعبه ابزار (ToolBox) را در صفحه ثابت کنید. |

| سوالات/نمره شایستگی | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---|---|--|
| ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | <p>فرم زیر را طراحی کنید. ویژگی های فرم را مانند شکل داده شده، تنظیم کنید:</p> <p>- رنگ زمینه فرم آبی</p> <p>- جهت نمایش فرم از راست به چپ</p>  |
| | | | | | ✓ | | | ✓ | ✓ | | | <p>- نوع قلم دکمه ها: B Nazanin با اندازه ۱۴ و سبک Bold</p> <p>- نوع قلم برجسب ها: B Titr با اندازه ۱۲</p> <p>- هنگام تایپ رمز عبور، به جای کاراکتر تایپ شده، کاراکتر "*" نمایش داده شود.</p> <p>- در ابتدای اجرای برنامه، تصویر دیده نشود.</p>  |
| | | | ✓ | | | | | | | | | <p>تصویر (دلخواه) را بگونه ای در کادر تصویر قرار دهید که به پوشه منابع اضافه شود.</p> |
| ✓ | ✓ | ✓ | | | | | | | | | | <p>با کلیک بر روی دکمه "قلم" کاربر بتواند رنگ قلم و نوع قلم کادرهای متن را به دلخواه تغییر دهد.</p> |

| سوالات /نمره شایستگی | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---|---|--|---|
| ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | ۳ | ۲ | ۱ | <p>با کلیک بر روی دکمه "ورود"، در صورت صحیح بودن نام کاربری و رمز عبور، تصویر نمایش داده شود و متن داخل کادرهای متن پاک شود. در غیر اینصورت پیام مناسب نمایش داده شود.</p> <p>(نام کاربری: "Exam5" و رمز عبور: "123")</p>  | ۷ |
| ✓ | | | ✓ | | | | | | | | | <p>با کلیک روی دکمه "خروج" از کاربر سوال شود: "می خواهید از برنامه خارج شوید؟" و اگر کاربر بر روی دکمه YES کلیک کرد، از برنامه خارج شود.</p>  | ۸ |

ارزشیابی یابانی واحد یادگیری ۶:

| مراحل کاری | به کارگیری کامپوننت زمان سنج | به کارگیری کنترل های انتخاب | گروه بندی کنترل ها | به کارگیری چند فرم در پروژه | استفاده از کنترل های فهرست |
|---------------------|---|--|--|---------------------------------|--|
| بحرانی | ۲ | ۲ | ۲ | ۱ | ۲ |
| استاندارد (شاخص ها) | زمان سنج روی فرم | تعیین ویژگی های زمان سنج در پنجره ویژگی ها، فعال و غیرفعال | تعیین واکنش برنامه به وضعیت انتخاب یا عدم انتخاب | افزودن فرم جدید به پروژه | دسترسی به عناصر فهرست، عنصر جاری و تعداد عناصر |
| | کندوبوسی برای استفاده از زمان سنج | درج دکمه رادیویی و کادر انتخاب | درج کادر گروهی و پیل و سربرگ | نمایش و مخفی کردن فرم زمان اجرا | درج کادر فهرست و کادر ترکیبی |
| سوال/نمره شایستگی | ۱ | ۱ | ۱ | ۱ | ۱ |
| ۱ | با اجرای برنامه، فرم "ورود" نمایش داده شود. (شکل ۱) | ✓ | | | |
| ۲ | شرط لاگین (ورود به فرم بعدی): (نام کاربری "admin"، رمز عبور ۱۲۳ و انتخاب سطح دسترسی "مدیر سیستم") یا (سطح دسترسی "کاربر مهمان" با نام دلخواه و بدون رمز عبور) | ✓ | | ✓ | |

| سوالات/نمره شایستگی | | ۱ | ۲ | ۳ | ۱ | ۲ | ۳ | ۱ | ۲ | ۳ | ۱ | ۲ | ۳ | ۱ | ۲ | ۳ |
|---------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ۳ | با انتخاب گزینه "نمایش گذرواژه"، گذرواژه وارد شده در کادر متن (TextBox) به صورت عادی و در غیر اینصورت به شکل ستاره نمایش داده شود. (شکل ۲ و ۳) | | | | | | | | | ✓ | | | | | | |
| ۴ | در صورت لاگین: فرم "قرعه کشی" که دارای ۲ سربرگ است، ظاهر شود. در سربرگ "قرعه کشی" نام کاربری وارد شده در فرم "ورود" در یک برچسب (Label) نمایش داده شود. در صورت ورود با سطح دسترسی "کاربر مهمان" کنترل های موجود در سربرگ تنظیمات غیر فعال باشند. (شکل ۴ و ۵) | | | ✓ | | | ✓ | ✓ | ✓ | | | | | | | |
| ۵ | -در سربرگ "قرعه کشی" با کلیک بر روی دکمه "اضافه به فهرست" نام وارد شده در کادر متن به کادر فهرست (ListBox) اضافه شود. (حداکثر ۴ مورد) -دکمه حذف زمانی فعال شود، که داده ای از فهرست انتخاب شده باشد و با کلیک بر روی آن، گزینه انتخاب شده، از فهرست حذف شود. - با کلیک بر روی دکمه "انتخاب" به طور تصادفی، یکی از نام های موجود در فهرست، انتخاب و در برچسب نمایش داده شود. (شکل ۴) | ✓ | ✓ | ✓ | | | | | | | | | | | | |
| ۶ | در سربرگ "قرعه کشی" ساعت سیستم، در بالای صفحه نمایش داده شود. | | | | | | | | | | | | ✓ | | ✓ | |
| ۷ | در سربرگ "تنظیمات" امکان تغییر رنگ سربرگ "قرعه کشی" یا انتخاب رنگ از یک کنترل کادر ترکیبی (ComboBox) وجود داشته باشد. همچنین بتوان وضعیت نمایش ساعت را کنترل کرد. (شکل ۶) | | ✓ | ✓ | | | | | | | | | | | ✓ | |

| | | |
|--------------|--------------|--------------|
| | | |
| <p>شکل ۳</p> | <p>شکل ۲</p> | <p>شکل ۱</p> |
| | | |
| <p>شکل ۶</p> | <p>شکل ۵</p> | <p>شکل ۴</p> |

ج) برای مطالعه بیشتر

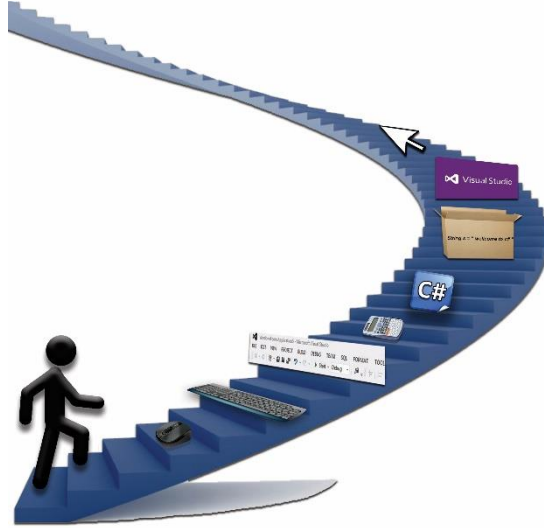
کنترل‌ها و ابزارهای شخص ثالث برای .Net.

علاوه بر کنترل‌هایی که در جعبه‌ابزار VS وجود دارد، کنترل‌های دیگر و ابزارهای پیشرفته‌ای نیز توسط شرکت‌های نرم‌افزاری که با شرکت مایکروسافت همکاری دارند، ساخته شده‌اند. به این نوع کنترل‌ها که قابلیت اضافه شدن به فرم‌های ویندوزی را دارند و توسط شرکت دیگری ساخته می‌شوند، کنترل‌های شخص ثالث نامیده می‌شوند. برخی از این ابزارها و کنترل‌ها در محیط ویژوال استودیو اضافه می‌شوند.



شرکت DevExpress از جمله شرکت‌هایی است که مجموعه کنترل‌ها و ابزارهای واسطه کاربری را برای تولیدکنندگان برنامه‌های کاربردی، از جمله برنامه‌نویسانی که از ویژوال استودیو استفاده می‌کنند، فراهم و به صورت تجاری عرضه کرده است. با استفاده از این کنترل‌ها، واسطه‌های کاربری زیباتری، بدون کدنویسی می‌توان ایجاد کرد. یکی از ابزارهایی که این شرکت به رایگان عرضه می‌کند، CodeRush Express است که می‌توانید آن را از آدرس زیر دانلود کنید. با نصب این ابزار در محیط VS، امکانات بیشتری در هنگام برنامه‌نویسی فراهم می‌شود. مثلاً منوی Refactoring با گزینه‌های متنوعی را در اختیار خواهید داشت.





پودمان چهارم

واحد یادگیری ۷ و ۸

توسعه واسط گرافیکی

واحد یادگیری ۷

شایستگی کار با ماوس و منو

مقدمات تدریسی

الف) مفاهیم کلیدی

| مفاهیم کلیدی | | | |
|--------------|------------------|----------|-------------|
| ماوس | دستگاه‌های ورودی | کنترل‌ها | واسط کاربری |
| | صفحه‌کلید | منو | رویداد |

ب) تجهیزات لازم

مشخصات سخت‌افزاری برای نصب نرم‌افزار Visual Studio Express 2012:

- پردازنده حداقل ۱/۶ گیگاهرتز
- حافظه RAM حداقل ۱ گیگابایت
- حداقل فضای موجود در دیسک سخت ۴ گیگابایت
- کارت ویدئویی متناسب با DirectX9 و قابل اجرا در رزولوشن 1024×768 و بالاتر

پیشنهاد می‌شود برای درک صحیح موضوع از صفحه‌کلیدهایی استفاده شود که دکمه‌های مربوط به NumPad از بقیه دکمه‌ها مستقل باشند.

ج) بودجه‌بندی

بودمان چهارم: توسعه واسط گرافیکی

| جلسه | واحد یادگیری | کارگاه (موضوع) | شماره صفحات | اهداف توانمندسازی | فعالیت‌های تکمیلی |
|------|--------------|-----------------|-------------|--|---|
| ۲۲ | ۷ | کارگاه ۱ تا ۳ | ۱۵۶-۱۶۲ | شناخت ماوس و دکمه‌ها شناخت رویدادهای مرتبط با ماوس و به‌کارگیری آن‌ها در توسعه واسط کاربری | واکنش به رویدادهای ماوس با نوشتن برنامه‌هایی که بتوانند رویدادهای ماوس را تشخیص داده و وضعیت مستقل از کلید و یا وابسته به کلید بودن آن را گزارش کنند. برنامه‌ریزی انجام یک فعالیت در صورت تحقق هر یک از رویدادهای زیر: الف) رویدادهای وابسته به کلید ب) رویدادهای مرتبط با حرکت ماوس ج) رویدادهای وابسته به زمان |
| ۲۳ | ۷ | کارگاه ۴ و ۵ | ۱۶۲-۱۶۶ | کار با آرگومان‌های رویدادهای ماوس استفاده از زمان‌سنج | برنامه‌ریزی انجام یک فعالیت در صورت تحقق هر یک از رویدادهای ماوس و به‌کارگیری آرگومان‌های رویداد برای نوشتن متدهای مشترک برای مدیریت رویدادها در کنترل‌های مشابه استفاده از زمان‌سنج در توسعه واسط کاربری |
| ۲۴ | ۷ | کارگاه ۶ تا ۸ | ۱۶۸-۱۷۳ | ایجاد و سفارشی کردن منو- به‌کارگیری رویدادهای منو در توسعه واسط کاربری | ایجاد، ویرایش و حذف گزینه‌های منو، تنظیم ویژگی‌های منو، برنامه‌ریزی رویداد برای گزینه‌های منو، کد نویسی برای گزینه استفاده از منوهای نواری، برنامه‌ریزی اجرای فعالیت‌ها در صورت کلیک روی گزینه‌های منو |
| ۲۵ | ۷ | کارگاه ۱۰ تا ۱۰ | ۱۷۴-۱۷۷ | طراحی منوی زمینه و انجام پروژه عملی با منو | ایجاد منوی زمینه، درج گزینه‌های منو، تنظیم ویژگی‌های منوی زمینه، تخصیص منوی زمینه به کنترل، اجرای یک پروژه عملی با امکان انجام فعالیت‌ها در صورت انتخاب گزینه‌های منو |
| | ۸ | کارگاه ۱ تا ۲ | ۱۸۱-۱۸۳ | آشنایی با رویدادهای صفحه‌کلید و ترتیب گزارش رویدادها در زمان فشردن دکمه‌ها | شناسایی رویدادهای صفحه‌کلید و ترتیب گزارش رویدادها برنامه‌ریزی انجام یک فعالیت در صورت تحقق رویدادهای صفحه‌کلید |
| ۲۶ | ۸ | کارگاه ۳ تا ۶ | ۱۸۴-۱۹۱ | کار با رویدادهای صفحه‌کلید، مدیریت رویدادها در زمان فشردن دکمه‌های ترکیبی، پردازش رویداد در فرم قبل از رسیدن رویداد به سایر کنترل‌های موجود در فرم | تعیین رویدادهای موردنیاز، ایجاد متد رویدادهای صفحه‌کلید، کد نویسی برای رویدادهای صفحه‌کلید، استفاده از آرگومان‌های رویداد صفحه‌کلید، تعیین کلیدهای ترکیبی فشرده‌شده. انجام یک فعالیت کارگاهی با موضوع به‌کارگیری رویدادهای صفحه‌کلید |
| ۲۷ | ۸ | کارگاه ۷ تا ۹ | ۱۹۱-۱۹۸ | کار با کلاس‌های آماده: استفاده از داده شمارشی Keys، کار با رویداد keyDown در کادر متن، کار با کلاس Char | ارائه تکلیف منزل و کارگاهی برای واکنش فرم به کلیدهای فشرده شده، انتقال فوکوس به سایر کنترل متناوب با کلید فشرده شده، تعیین کلیدهای فشرده شده با استفاده از مقادیر داده شمارشی آماده، فیلتر کردن ورودی در کادر متن |
| ۲۸ | ۸ | کارگاه ۱۰ تا ۱۲ | ۱۹۹-۲۰۲ | کار با متدهای کلاس‌های آماده کلاس String | ارائه تکلیف منزل و کارگاهی با موضوع اعمال پردازش روی رشته‌ها به کمک متدهای آماده کلاس String، نوشتن برنامه‌های ساده و پیچیده با متدهای رشته‌ای |

| طرح درس روزانه (هفتگی) پیشنهادی | | | | |
|---|--|--|---|---|
| درس: توسعه برنامه‌سازی و پایگاه داده | | | پایه: یازدهم | |
| پیام جلسه (هدف کلی): شناخت واسط کاربری، شناخت ماوس و رویدادهای مستقل و وابسته به کلید در ماوس | | | | |
| زمان | فعالیت‌ها | | اهداف یادگیری | |
| مدت (دقیقه) | کار هنرجو | کار هنرآموز | طبقه هدف: حیطه عاطفی / شناختی / روانی - حرکتی | فعالیت |
| ۱۰ | مشارکت در فرآیند تشخیص رویدادهای محیطی (حرکت ثانیه‌شمار، ضربان قلب، تنفس، شنیدن صدا و ...) و تهیه کردن فهرستی از رویدادهای در حال انجام که قادر به تشخیص آن‌ها است | معطوف کردن توجه هنرجویان به اتفاقاتی که در محیط در حال روی دادن است (حرکت ثانیه‌شمار، ضربان قلب، تنفس، شنیدن صدا و ...) | سنجش میزان آگاهی هنرجویان از مفاهیم رویدادها | ارزشیابی رفتار ورودی |
| ۲۰ | باید انجام فعالیت‌های مشخصی را بر اساس رویدادهایی که تشخیص می‌دهد، انجام دهد و نتیجه انجام کار را گزارش دهد. | هنرجویان را به دو/چند دسته تقسیم کنید و از آن‌ها بخواهید تا نسبت به اتفاقاتی که در محیط پیرامون آن‌ها می‌افتد واکنش‌هایی را از خود نشان دهند. برای مثال می‌توانید افزایش / کاهش یک واحد به عددی که از قبل روی تخته وجود دارد و نوشتن مجدد نتیجه خروجی در قسمت مشخصی از تخته کلاس را در نظر بگیرید. | درک رویداد و انجام واکنش‌های از قبل برنامه‌ریزی‌شده در صورت وقوع رویداد | ایجاد انگیزه |
| ۳۰ | مشارکت در پاسخگویی و تعامل با هنرآموز در فرآیند تدریس | مفاهیم کلیدی مطرح‌شده در این واحد یادگیری را برای هنرجو تشریح کند. | توضیح کامل مفاهیم کلیدی (دانشی) و ایجاد علاقه و انگیزه در هنرجویان (بینشی) | ارائه مفاهیم کلیدی (توضیح هنرآموز) |
| ۶۰ | هنرجو به صورت گروهی اقدام به انجام فعالیت کارگاه ۱ می‌کند. | هنرآموز توضیح کامل در خصوص فعالیت کارگاهی ۱ به هنرجو بدهد. | هنرجو باید رویدادهای ماوس را بشناسد و وقوع آن‌ها را در یک برنامه ساده آشکار کند | انجام فعالیت کارگاهی شماره ۱ (تمرین هنرجویان) |
| ۶۰ | هنرجویان ممتاز موظف به ارائه مفاهیم به هنرجویان ضعیف‌تر باشند. | رفتار هنرجو را در حین اجرای تکالیف زیر نظر داشته باشد و در صورت لزوم از هنرجویان مستعدتر برای آموزش مفاهیم به آن‌ها کمک بگیرد. | بررسی نقاط ضعف هنرجویان در درک مفاهیم کلیدی | نظارت بر عملکرد هنرجویان و ارزیابی فعالیت‌ها |

| طرح درس روزانه (هفتگی) پیشنهادی | | | |
|---|--|--|----|
| پایه: یازدهم | | درس: توسعه برنامه‌سازی و پایگاه داده | |
| پیام جلسه (هدف کلی): شناخت واسط کاربری، شناخت ماوس و رویدادهای مستقل و وابسته به کلید در ماوس | | | |
| اهداف یادگیری | فعالیت‌ها | زمان | |
| انجام فعالیت کارگاهی شماره ۲ (تمرین هنرجویان) | هنرجو باید رویدادهای ماوس را بشناسد و وقوع آن‌ها را در یک برنامه ساده آشکار کند. | هنرآموز توضیح کامل در خصوص فعالیت کارگاهی ۲ به هنرجو بدهد. | ۶۰ |
| نظارت بر عملکرد هنرجویان و ارزیابی فعالیت‌ها | بررسی نقاط ضعف هنرجویان در درک مفاهیم کلیدی | رفتار هنرجو را در حین اجرای تکالیف زیر نظر داشته باشد و در صورت لزوم از هنرجویان مستعدتر برای آموزش مفاهیم به آن‌ها کمک بگیرد. | ۶۰ |
| انجام فعالیت کارگاهی شماره ۳ (تمرین هنرجویان) | هنرجو باید رویدادهای ماوس را بشناسد و وقوع آن‌ها را در یک برنامه ساده آشکار کند. | هنرآموز توضیح کامل در خصوص فعالیت کارگاهی ۳ به هنرجو بدهد. | ۶۰ |
| نظارت بر عملکرد هنرجویان و ارزیابی فعالیت‌ها | بررسی نقاط ضعف هنرجویان در درک مفاهیم کلیدی | رفتار هنرجو را در حین اجرای تکالیف زیر نظر داشته باشد و در صورت لزوم از هنرجویان مستعدتر برای آموزش مفاهیم به آن‌ها کمک بگیرد. | ۶۰ |
| دریافت بازخورد از تدریس | هنرجو بتواند فعالیت منزل صفحه ۱۶۲ کتاب را انجام دهد. | هنرجو را در خصوص نحوه انجام فعالیت راهنمایی کند. انجام تکالیف به صورت گروهی باشد. | ۶۰ |
| ابزارهای موردنیاز ویدئو پروژکتور، رایانه، تخته آموزشی، دفتر یادداشت | | | |

طرح درس روزانه (هفتگی) پیشنهادی

| پایه: یازدهم | | درس: توسعه برنامه‌سازی و پایگاه داده | |
|---|--|--|--|
| پیام جلسه (هدف کلی): آشنایی با رویدادهای صفحه‌کلید و ترتیب گزارش رویدادها در زمان فشردن دکمه‌ها | | | |
| زمان | فعالیت‌ها | | اهداف یادگیری |
| | کار هنرجو | کار هنرآموز | فعالیت طبقه هدف: حیطه عاطفی/شناختی/روانی - حرکتی |
| ۱۰ مدت (دقیقه) | | | |
| ۱۰ | کلیه رویدادهای محیط کلاس را که قادر به تشخیص آن است را یادداشت کند و به هنرآموز گزارش دهد. | معطوف کردن توجه هنرجویان به اتفاقاتی که در محیط در حال روی دادن است (تغییر نور محیط، حرکت ثانیه‌شمار، ضربه قلب، تنفس، شنیدن صدا و تغییر دمای محیط) | سنجش میزان آگاهی هنرجویان از مفاهیم رویدادها |
| ۲۰ | فعالیت‌های مشخصی را بر اساس رویدادهایی که تشخیص می‌دهد، انجام دهد. | هنرجویان را به دو/چند دسته تقسیم کنید و از آن‌ها بخواهید تا نسبت به اتفاقاتی که در محیط پیرامون آن‌ها می‌افتد واکنش‌هایی را از خود نشان دهند. | درک رویداد و انجام عکس‌العمل از قبل برنامه‌ریزی شده در صورت بروز رویداد |
| ۲۰ | مشارکت در پاسخگویی و تعامل با هنرآموز در فرآیند تدریس | مفاهیم کلیدی مطرح شده در این واحد یادگیری را برای هنرجو تشریح کند. | توضیح کامل مفاهیم کلیدی (دانشی) و ایجاد علاقه و انگیزه در هنرجویان (بینشی) |
| ۲۰ | هنرجو به صورت گروهی اقدام به انجام فعالیت کارگاه ۱ می‌کند. | هنرآموز توضیح کامل در خصوص فعالیت کارگاهی ۱ به هنرجو بدهد. | هنرجو باید رویدادهای صفحه‌کلید را بشناسد و وقوع آن‌ها را در یک برنامه ساده آشکار کند. |
| ۱۰ | هنرجویان ممتاز موظف به ارائه مفاهیم به هنرجویان ضعیف‌تر باشند. | رفتار هنرجو را در حین اجرای تکالیف زیر نظر داشته باشد و از هنرجویان مستعدتر برای آموزش کمک بگیرد. | بررسی نقاط ضعف هنرجویان در درک مفاهیم کلیدی |
| ۱۰ | تقسیم‌بندی وظایف کار پروژه بین اعضای گروه و تحویل در زمان‌بندی تعیین شده | هنرجو را در خصوص نحوه ایجاد رنگ با استفاده از کلاس Random راهنمایی کند. | هنرجو بتواند برنامه‌ای بنویسد که با فشردن و رها شدن کلید رنگ زمینه فرم تصادفی تغییر کند. |
| | | | ابزارهای موردنیاز ویدئو پروژکتور، رایانه، تخته آموزشی، دفتر یادداشت |

د) ورود به بحث

قبل از شروع بحث لازم است ذهن هنرجو را به سمت مفاهیم زیر معطوف کنید.

واسط کاربری، کنترل‌ها، دستگاه‌های ورودی، صفحه کلید، ماوس، رویداد واسط کاربری ارتباط بین کاربر با نرم‌افزار را از طریق کنترل‌ها فراهم می‌کند، کنترل‌ها اشیایی هستند که امکان تعامل کاربر با نرم‌افزار را از طریق دستگاه‌های ورودی همچون صفحه کلید و ماوس فراهم می‌کنند، یک شی مجموعه‌ای از داده‌ها، امکانات کار با داده و در برخی موارد دارای قابلیت شناسایی/ درک رویدادها است.

در هر لحظه ممکن است هزاران رویداد در دنیای پیرامون ما روی دهد ولی هر موجودی که قابلیت شناسایی/درک این رویدادها را داشته باشد در زمان وقوع آن رفتاری متناسب با آن رویداد را از خود بروز می‌دهد. این رفتار باید از قبل برای آن موجود برنامه‌ریزی شده باشد.

برای درک بهتر موضوع می‌توانید مواردی شبیه این مثال‌ها را یادآوری کنید:

- ✓ زلزله یک رویداد طبیعی است، آیا رفتار بزرگ‌ترها و کودکان در مقابل لرزش‌های این رویداد طبیعی یکسان است؟
- ✓ هر حرکت ثانیه‌شمار ساعت دیواری یک رویداد است، آیا رفتار شما و سایر اعضای غیرمحصل خانواده در رسیدن عقربه‌های ساعت به ساعت ۷:۰۰ صبح یکسان است؟

شناخت و درک صحیح مفهوم رویداد توسط هنرجویان می‌تواند در نیل به اهداف این واحد یادگیری به شما کمک کند.

ایجاد انگیزه در هنرجویان

پیشنهاد می‌شود در شروع تدریس این واحد یادگیری هنرجویان را به دو/چند دسته تقسیم کنید و از آن‌ها بخواهید تا نسبت به اتفاقاتی که در محیط پیرامون آن‌ها می‌افتد واکنش‌هایی را از خود نشان دهند، می‌توانید از تخته کلاس به‌عنوان نمایشگر بهره‌گیرید و خروجی‌ها را روی آن منعکس کنید، برای درک بهتر این موضوع توسط هنرجویان باید انجام فعالیت‌های مشخصی را برای آن‌ها در نظر بگیرید برای مثال می‌توانید افزایش/ کاهش یک واحد به عددی که از قبل روی تخته وجود دارد و نوشتن مجدد نتیجه خروجی در قسمت مشخصی از تخته کلاس را در نظر بگیرید. کلاس را به دو گروه تقسیم کنید، حال انجام عمل افزایش مقدار را مقید به وقوع یک رویداد و انجام عمل کاهش مقدار را مقید به وقوع رویداد دیگری کنید، نظارت کنید تا با وقوع رویدادها، هر گروه کار خودش را به‌طور صحیح انجام دهد. این روش می‌تواند هنرجو را با نحوه‌ی اجرای برنامه توسط کنترل‌ها در محیط ویندوز آشنا کند. در ادامه کار ترتیبی اتخاذ کنید که گروهی که مسئول کاهش مقدار است وقوع رویداد را متوجه نشود، طبیعتاً این گروه با وجودی که رویداد اتفاق افتاده است ولی عکس‌العملی بروز

نخواهند داد، این موارد مشابه همان حالتی خواهد بود که شما در محیط ویندوز با غیرفعال کردن ویژگی Enabled یک کنترل ایجاد می‌کنید.

تدریس

عملکرد ماوس در ویندوز و برنامه‌های کاربردی

یک نرم‌افزار دارای یک سری قابلیت‌های کاربردی است که در راستای هدف اصلی تولید آن نرم‌افزار است. این قابلیت‌ها در بطن نرم‌افزار نهفته شده است. هر نرم‌افزار دارای یک بطن و یک پوسته است، پوسته‌ی نرم‌افزار همان واسط کاربری و اجزاء ملموس نرم‌افزار است که برای کاربر و استفاده‌کننده از نرم‌افزار قابل رؤیت است. نکته اساسی و مهم در تولید نرم‌افزار داشتن واسط گرافیکی کاربرپسند و متناسب با نیازهای کاربر است، برای این منظور در تولید واسط کاربری از مجموعه‌ای از منوها و عناصر گرافیکی استفاده می‌شود تا دسترسی راحت‌تری به قابلیت‌های نرم‌افزار را برای کاربر فراهم کند. منوها^۱ و عناصر گرافیکی^۲ به‌کاررفته در واسط کاربری قادرند تا رویدادهای ماوس و صفحه‌کلید را تشخیص داده و متناسب با این رویدادها واکنش‌های از پیش تعریف‌شده‌ای را از خود بروز دهند.

ماوس یکی از دستگاه‌های ورودی است که توسط دست کنترل می‌شود و قادر است حرکت دوبعدی را نسبت به یک سطح تشخیص دهد. این حرکت‌ها باعث جابه‌جایی نشانگر ماوس بر روی صفحه‌نمایش می‌شود.

به‌طور معمول ماوس دستگاه کوچکی است با یک یا چند دکمه که توسط یک دست قابل کنترل است و برای کار با اشیای موجود بر روی صفحات نمایش گرافیکی کاربرد دارد.

ماوس‌ها در گذشته بر اساس حرکت چرخ (گوی متحرک) و در حال حاضر بر اساس سیستم تابش نور، اقدام به تشخیص حرکت در فضای دوبعدی می‌کنند. ماوس‌ها بر اساس حرکت دست کاربر، موقعیت نشانگر ماوس را بر روی صفحه‌نمایش گرافیکی کنترل می‌کنند. حرکت نشانگر در یک فضای دوبعدی در صفحه‌نمایش انجام می‌شود، بنابراین ماوس باید قادر باشد مختصات X, Y را نسبت به حرکت دست کاربر تعیین کند. ماوس علاوه بر امکان تشخیص حرکت، دارای دکمه‌هایی برای انجام امور قابل برنامه‌ریزی است. در حال حاضر ماوس‌ها عمدتاً دارای سه دکمه هستند که عبارت‌اند

^۱ چنانچه یک نرم‌افزار دارای امکانات متعددی باشد، از منوها برای ایجاد قابلیت دسترسی به امکانات نرم‌افزار بهره می‌گیرد، درواقع یک منو فهرستی دسته‌بندی‌شده و قابل انتخاب است که امکان دسترسی به قابلیت‌های نرم‌افزار را برای کاربر استفاده‌کننده فراهم می‌کند.

^۲ عناصر گرافیکی یا آیکون‌ها امکان دسترسی راحت‌تر به امکانات داخلی یک نرم‌افزار را برای کاربر استفاده‌کننده فراهم می‌کنند.

از دکمه سمت چپ، دکمه سمت راست و دکمه وسط (دکمه اسکرول). ماوس قادر است همزمان با تشخیص حرکت، فشردن هر یک از این دکمه‌ها را نیز تشخیص داده و به سیستم‌عامل گزارش کند.

ما در برنامه‌نویسی قادر هستیم اجرای یک سری از امور و فعالیت‌ها را متناسب با تحقق هر یک از رویدادهای ماوس برنامه‌ریزی کنیم که از آن جمله می‌توان به برنامه‌ریزی اجرای یک فعالیت در صورت تحقق هر یک از رویدادهای زیر اشاره کرد.

- ✓ در صورت تحقق رویداد فشردن / رها کردن دکمه‌های ماوس
- ✓ در صورت تحقق رویداد حرکت ماوس
- ✓ در صورت تحقق ترکیبی از رویدادهای دو حالت قبلی

برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویداد فشردن / رها کردن

دکمه‌های ماوس

رویدادهای فشردن و رها کردن دکمه‌های ماوس می‌تواند شامل موارد زیر باشد:

- ✓ فشردن دکمه ماوس
- ✓ رها کردن دکمه ماوس
- ✓ فشردن و رها کردن دکمه سمت چپ (یک‌بار : کلیک، دو بار : دابل کلیک)

برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویداد حرکت ماوس

حرکت ماوس می‌تواند شامل موارد زیر باشد:

- ✓ ورود نشانگر ماوس به محدوده یک شی
- ✓ حرکت نشانگر ماوس در محدوده یک شی
- ✓ خروج نشانگر ماوس از محدوده یک شی

برنامه‌ریزی اجرای یک فعالیت در صورت تحقق ترکیبی از رویدادهای دو

حالت قبلی

ترکیب حرکت همزمان نشانگر ماوس درحالی‌که دکمه‌های آن نیز فشرده شده باشد، می‌تواند شامل موارد زیر باشد:

- ✓ فشردن ماوس بر روی یک شی و سپس حرکت ماوس و در نهایت رها کردن دکمه ماوس
- ✓ فشردن ماوس بر روی یک شی و سپس حرکت ماوس

مشکلات متداول در فرآیند یادگیری - یاددهی

در برنامه‌نویسی رویدادگرا، توانایی تشخیص صحیح رویدادهای قابل برنامه‌ریزی دارای اهمیت ویژه‌ای است و شاید بتوان گفت اصلی‌ترین چالش مدیریت برنامه بر اساس وقوع رویدادهای مختلف است، در پایان این واحد یادگیری هنرجو باید به درک

صحیحی از رویدادهای ماوس، نحوه مدیریت رویدادهای ماوس و ایجاد و مدیریت گزینه‌های منو رسیده باشد. همکاران گرامی باید دقت داشته باشند که حرکت ماوس یا فشردن دکمه‌های آن باعث وقوع چندین رویداد مختلف خواهد شد و آنچه اهمیت دارد دانستن ترتیب وقوع این رویدادها و انتخاب رویداد متناسب با عملی است که باید انجام پذیرد.

شیوه و الگوی پیشنهادی

برنامه‌نویسی خوب، بدون داشتن الگو و پیش‌زمینه قبلی نتیجه‌ای جز اتلاف وقت و استفاده نامناسب از تجهیزات کارگاه شما در پی نخواهد داشت. پیشنهاد می‌شود با توجه به محدودیت تجهیزات کارگاهی، صرفاً زمانی که هنرجو تکلیفی برای انجام دادن آماده کرده باشد، اجازه استفاده از تجهیزات به وی داده شود. برای اجرایی کردن این روش می‌توان در محیط کارگاه یک میز اجلاس برای بحث و تبادل نظر بین هنرجویان در نظر گرفت و در زمانی که هنرجو به نتیجه رسید، اجازه تست بر روی رایانه به وی داده شود. تشخیص این زمان به عهده هنرآموز است. چنانچه هنرجویان به طور مستمر به سیستم دسترسی داشته باشند، با سعی و خطا و عدم اطمینان اقدام به تکمیل برنامه‌ها می‌کنند و عملاً فرآیند یادگیری و تفکر برنامه‌نویسی در آن‌ها ضعیف می‌شود.

پاسخ به فعالیت‌ها

به کمک هم‌گروهی‌های خود کاربردهای دیگر ماوس در ویندوز و برنامه‌های کاربردی را پیدا کنید و در جدول زیر بنویسید.

فعالیت گروهی
ص ۱۵۵

| کاربرد ماوس در برنامه‌های کاربردی | کاربرد ماوس در ویندوز |
|---|--|
| انتخاب، جابه‌جایی، تغییر شکل و اندازه اشیا، کلیک کردن دکمه‌ها، انتخاب گزینه‌های منو | اجرای برنامه‌ها، کپی پرونده یا پوشه، جابه‌جایی آیکون‌ها، فعال کردن منوهای کمکی با کلیک راست روی آیکون‌ها |
| کاربرد ماوس در بازی‌ها | کاربرد ماوس در نرم‌افزار MS-Word |
| انتخاب، جابه‌جایی، تغییر شکل و اندازه اشیا، به کارگیری ماوس به جای دسته بازی | انتخاب بخشی از متن با ماوس، انتخاب واژه، جمله و پارامتن با یک یا چند بار کلیک روی متن، ترسیم اشکال |

شرح رویدادهای جدول ۱ را با توجه به شرح آن‌ها در VS بنویسید.

| شرح رویداد در VS | شرح رویداد | نام رویداد |
|---|---|-------------------------|
| Occurs when the control is clicked by the mouse. | زمانی رخ می‌دهد که کنترل با ماوس کلیک شود. | MouseClicked |
| Occurs when the control is double clicked by the mouse. | زمانی رخ می‌دهد که کنترل با ماوس دابل کلیک شود. | MouseDoubleClick |
| Occurs when the mouse pointer is over the control and a mouse button is pressed. | زمانی رخ می‌دهد که نشانگر ماوس بر روی کنترل باشد و دکمه ماوس فشرده شود. | MouseDown |
| Occurs when the mouse pointer is over the control and a mouse button is released. | زمانی رخ می‌دهد که نشانگر ماوس بر روی کنترل باشد و دکمه ماوس رها شود. | MouseUp |

با انجام عملیات زیر روی ماوس عملیات چه رویدادی رخ می‌دهد؟

| رویداد | عملیات |
|-----------|---------------------------|
| MouseDown | نگه‌داشتن دکمه راست یا چپ |
| MouseUp | رها کردن دکمه راست یا چپ |

کارگاه ۱ – واکنش به رویدادهای ماوس

ما در برنامه‌نویسی قادر هستیم اجرای یک سری از امور و فعالیت‌ها را متناسب با تحقق هر یک از رویدادهای ماوس برنامه‌ریزی کنیم. هدف از این کارگاه آموزش نحوه کار با رویدادهای وابسته به کلید در ماوس است. در این کارگاه هنرجو با برنامه‌ریزی رویدادهای `MouseClicked`, `MouseDown`, `MouseUp` آشنا می‌شود و برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویداد فشردن/رها کردن دکمه‌های ماوس را فرامی‌گیرد.

پاسخ به فعالیت‌ها

مرحله ۳: متد رویداد `MouseClicked` را برای کنترل `btnRun` ایجاد کنید.

```
private void btnRun_MouseClick(object sender,
MouseEventArgs e)
{
    MessageBox.Show("به سی شارپ خوش آمدید");
}
```

مرحله ۴: قابلیت خروج از برنامه را با دابل کلیک روی فرم ایجاد کنید.

```
private void Form1_MouseDoubleClick(object sender,
MouseEventArgs e)
{
    Application.Exit();
}
```

مرحله ۵: با ننگه داشتن دکمه ماوس رنگ دکمه btnRun چه تغییری می کند. رنگ دکمه قرمز می شود.

مرحله ۶: در رویداد MouseUp دکمه btnRun رنگ دکمه را به حالت اول برگردانید.

```
private void btnRun_MouseUp(object sender,
MouseEventArgs e)
{
    btnRun.BackColor = Color.Blue;
}
```

کارگاه ۲ – رویدادهای MouseUp و MouseDown

هدف از این کارگاه آموزش نحوه کار با رویدادهای وابسته به کلید در ماوس است. در این کارگاه هنرجو با برنامه ریزی رویدادهای MouseDown, MouseUp آشنا می شود و برنامه ریزی اجرای یک فعالیت در صورت تحقق رویداد فشردن/رها کردن دکمه های ماوس را فرامی گیرد. علاوه بر این موارد در این کارگاه یک زمان سنج نیز در نظر گرفته شده است که در زمان های مشخصی رویداد تیک آن فعال می شود و هنرجو قادر است با برنامه ریزی انجام یک فعالیت در زمان تحقق این رویداد، حرکت منظمی را برای تصویر ماشین بر روی صفحه نمایش شبیه سازی کند.

پاسخ به فعالیت ها

چرا در این برنامه نیاز به زمان سنج داریم؟

پاسخ: به منظور برنامه ریزی حرکت خودکار ماشین

کنجکاو
ص ۱۵۸

ویژگی Right چه مقداری را نگهداری می کند؟

پاسخ: فاصله لبه سمت راست شی را از لبه سمت چپ فرم نگهداری می کند.

برنامه پارک ماشین را اجرا کنید. با چه شرایطی، کاربر برنده می شود؟

پاسخ: اگر در زمان توقف، فاصله ماشین با مانع بین ۹ تا ۱ پیکسل باشد کاربر برنده است.

تکمیل کارگاه
ص ۱۵۸

پس از پایان یک بازی چگونه می‌توان بازی جدیدی را شروع کرد؟ برنامه آن را بنویسید.

پاسخ: یک دکمه بانام btnNew به فرم اضافه می‌کنیم و در رویداد MouseClick کد زیر را می‌نویسیم

```
private void btnNew_MouseClick(object sender, EventArgs e)
{
    picCar.Left = 20;
}
```

فعالیت منزل
ص ۱۵۹

بازی کارگاه ۲ را با شرایط زیر تغییر دهید:

- پس از رها کردن ماوس، ماشین به نسبت زمان نگه‌داشتن دکمه ماوس، حرکت کند. به‌طور مثال اگر کاربر یک ثانیه دکمه ماوس را نگه دارد، ماشین ۱۰۰ پیکسل حرکت کند.

پاسخ: زمان نگه‌داشتن دکمه ماوس به چندین صورت قابل محاسبه است که در زیر به دو روش معمول آن اشاره می‌کنیم.

روش اول: با افزودن یک متغیر اضافه به برنامه برای نگهداری زمان و سپس افزایش یک واحدی متغیر موردنظر به ازای هر بار اجرای رویداد تیک زمان‌سنج.

روش دوم: استفاده از مقدار جابه‌جایی ماشین، در این روش می‌توان با محاسبه اختلاف مقدار موجود در ویژگی Left کنترل استفاده‌شده برای ماشین، با مقدار اولیه آن و سپس تقسیم بر ۵ کردن عدد به‌دست‌آمده زمان را به دست آورد. (دقت کنید که ماشین در هر تیک به‌اندازه ۵ واحد حرکت می‌کند.)

کارگاه ۳ - رویدادهای MouseEnter و MouseLeave

هدف از این کارگاه آموزش نحوه کار با رویدادهای مستقل از کلید در ماوس است. در این کارگاه هنرجو با برنامه‌ریزی رویدادهای MouseEnter و MouseLeave آشنا می‌شود و برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویدادهای مرتبط با حرکت ماوس را فرامی‌گیرد.

پاسخ به فعالیت‌ها

چرا ویژگی Visible کادر تصویرها برابر با false است؟

پاسخ: چون در شروع اجرای برنامه تصویر واژه‌ها پنهان باشد.

در فعالیت ۵ از کارگاه ۳ پیشنهاد می‌شود به هنرجو توضیح داده شود بهتر است این دستور در قسمتی از برنامه قرار داده شود که یکبار تکرار شود. پیشنهاد می‌شود که مورد نظر عینا به متد Load فرم منتقل شود.

کنجکاو
ص ۱۶۱

در متد رویداد lblBook_MouseLeave دستوری بنویسید که picBook پنهان شود و رنگ lblBook مشکی شود.

```
private void lblBook_MouseLeave(object sender, EventArgs e)
{
    picBook.Visible = false;
    lblBook.ForeColor = Color.Black;
}
```

ستون شرح رویدادها در جدول ۲ را تکمیل کنید.

| شرح رویداد در VS | شرح رویداد | نام رویداد |
|---|--|------------|
| Occurs when the mouse enters the visible part of the control. | زمانی رخ می‌دهد که ماوس به بخشی از کنترل وارد شود. | MouseEnter |
| Occurs when the mouse pointer leaves the control. | زمانی رخ می‌دهد که نشانگر ماوس از محدوده کنترل خارج شود. | MouseLeave |
| Occurs when the mouse pointer is moved over the control. | زمانی رخ می‌دهد که نشانگر ماوس بر روی کنترل حرکت کند. | MouseMove |
| Occurs when the mouse pointer rests on the control. | زمانی رخ می‌دهد که نشانگر ماوس بر روی کنترل قرار گیرد. | MouseHover |

برنامه آموزش، کارگاه ۳ را برای آموزش واژگان فارسی، استفاده کنید. کاربر با ورود ماوس به روی تصویر، واژه فارسی معادل را ببیند.

برای جلوگیری از تکرار کدنویسی، کافه است ابتدا کادر تصویر picApple را انتخاب کنید، آنگاه با پایین نگاه‌داشتن دکمه کنترل بقیه کادرهای تصویر را انتخاب کنید و در نهایت از برگه Properties بر روی رویداد MouseLeave دابل کلیک کرده و در رویداد باز شده کد زیر را قرار دهید.

```
private void picApple_MouseLeave(object sender, EventArgs e)
{
    lblFarsiName.Text = "";
}
```

سپس برای هر کادر تصویر رویداد MouseEnter را به صورت زیر بنویسید.

```
private void picBook_MouseEnter(object sender,EventArgs
e)
{
    lblFarsiName.Text = "کتاب";
}
private void picPen_MouseEnter(object sender,EventArgs e)
{
    lblFarsiName.Text = "قلم";
}
private void picCar_MouseEnter(object sender,EventArgs e)
{
    lblFarsiName.Text = "ماشین";
}
private void picApple_MouseEnter(object sender,EventArgs
e)
{
    lblFarsiName.Text = "سیب";
}
private void picClock_MouseEnter(object sender,EventArgs
e)
{
    lblFarsiName.Text = "ساعت";
}
}
```

کارگاه ۴ – آرگومان‌های رویدادهای ماوس

هدف از این کارگاه آموزش نحوه کار با رویدادهای مستقل از کلید و وابسته به کلید در ماوس است. به طور خلاصه در جدول زیر ۸ رویداد معمول ماوس براساس اولویت گزارش در زمان وقوع و دسته‌بندی براساس استقلال از کلید و یا وابسته به کلید آورده شده است. (شماره‌ها ترتیب وقوع رویداد را مشخص می‌کند)

| رویدادهای مستقل از کلید | رویدادهای وابسته به کلید |
|-------------------------|--------------------------|
| 1.MouseEnter | 3.MouseDown |
| 2.MouseMove | 3.1.Click |
| 5. MouseHover | 3.2.MouseClick |
| 6. MouseLeave | 4.MouseUp |

در سه کارگاه قبلی هنرجو با برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویداد فشردن / رها کردن دکمه‌های ماوس و نیز برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویداد حرکت ماوس آشنا شده است. در این کارگاه هنرجو با برنامه‌ریزی اجرای

یک فعالیت در صورت تحقق ترکیبی از رویدادهای دو حالت قبلی آشنا می‌شود و برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویدادهای مرتبط با حرکت ماوس را فرامی‌گیرد.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۱۶۲

۴- قطعه کدی بنویسید که با رها کردن دکمه ماوس، رنگ دکمه‌ها مشکی شود.

```
Private void MousePanel_MouseUp(object sender,
MouseEventArgs e)
{
    picRight.BackColor= Color.Black;
    pic Middle.BackColor= Color.Black;
    picLeft.BackColor= Color.Black;
}
```

۶- کد مرحله ۵ را تغییر دهید تا به جای تصویر اشاره‌گر ماوس، شکل نمادین ماوس حرکت کند.

برای پاسخ به این مرحله به دو روش می‌توانید عمل کنید .

۱- بدون تغییر کد فقط تصویر موجود در `picCursor` را تغییر دهید و شکل نمادین ماوس را در آن بارگذاری کنید.

۲- با توجه به شرایط فشردن دکمه‌های ماوس ۴ تصویر مختلف داشته باشید و در رویداد `MouseDown` تصویر موردنظر را در `picCursor` بارگذاری کنید. در این وضعیت باید در حالت پیش‌فرض و شروع برنامه تصویر عادی ماوس در `picCursor` بارگذاری شده باشد.

فعالیت کارگاهی
ص ۱۶۴

به برنامه کارگاه ۴ امکان تعیین راست‌دست و چپ‌دست بودن کاربر را اضافه کرده، متناظر با آن دکمه‌های ماوس را جابه‌جا کنید.

پاسخ: کافی است یک کادر علامت به نام `chkLeftHand` به فرم اضافه کنید و کد زیر را در رویداد `MouseDown` پنل `MousePanel` بنویسید.

```
private void MousePanel_MouseDown(object sender,
MouseEventArgs e)
{
    MouseButton mb=e.Button;
    if(chkLeftHand.Checked)
    {
        if(mb==MouseButton.Left)
            mb=MouseButton.Right;
        else if(mb==MouseButton.Right)
            mb=MouseButton.Left;
    }
}
```

```

switch (mb)
{
    case MouseButton.Left:
        picLeft.BackColor = Color.White;
        break;
    case MouseButton.Middle:
        picMiddle.BackColor = Color.White;
        break;
    case MouseButton.Right:
        picRight.BackColor = Color.White;
        break;
}
}

```

تکمیل جدول
ص ۱۶۴

در VS با قرار دادن اشاره گر ماوس روی هر ویژگی، نوع و عملکرد آن به صورت راهنما در اختیار برنامه نویس قرار می گیرد. با توجه به این راهنما جدول ۳ را تکمیل کنید.

| نام ویژگی | نوع داده | شرح | شرح در VS |
|-----------|---------------------------|---|---|
| Button | داده شماری MouseButton | مشخص می کند که کدام دکمه ماوس فشار داده شده است. (راست-چپ-وسط) | Gets which mouse button was pressed. |
| Location | کلاس Point | موقعیت نشان گر ماوس را روی کنترل مشخص می کند. | Gets the location of the mouse during the generating mouse event. |
| X | Int | موقعیت افقی نشانگر ماوس را در زمان وقوع رویداد ماوس مشخص می کند. | Gets the x-coordinate of the mouse during the generating mouse event. |
| Y | Int | موقعیت عمودی نشانگر ماوس را در زمان وقوع رویداد ماوس مشخص می کند. | Gets the y-coordinate of the mouse during the generating mouse event. |
| Clicks | Int | تعداد دفعات فشردن و رها شدن دکمه ماوس را مشخص می کند. | Gets the number of times the button was pressed and released. |

برنامه‌ای بنویسید که با حرکت ماوس در فضای فرم واژه Computer به همراه نشان‌گر ماوس نمایش داده شود.

پاسخ: یک کنترل برجسب بانام lblSample به فرم اضافه کنید و مقدار ویژگی Text آن را Computer قرار دهید. و برای حرکت از رویداد Form1_MouseMove استفاده کنید.

```
private void Form1_MouseMove(object sender,
MouseEventArgs e)
{
    lblSample.Location = e.Location;
}
```

برنامه بالا را به گونه‌ای ویرایش کنید که به جای واژه Computer مختصات ماوس به صورت (X,Y) به همراه نشان‌گر ماوس نمایش داده شود.

```
private void Form1_MouseMove(object sender,
MouseEventArgs e)
{
    lblSample.Location = e.Location;
    lblSample.Text = "(" + e.X + ", " + e.Y + ")";
}
```

بررسی کنید چگونه می‌توان با برنامه‌نویسی، یک متد رویداد را برای یک کنترل تخصیص داد.

پاسخ: ابتدا یک کنترل دلخواه ایجاد کنید. سپس یک متد با پارامترهای مشابه رویداد موردنظرتان ایجاد کنید. و با استفاده از دستور زیر نسبت به تخصیص متد دلخواه به رویداد موردنظر اقدام کنید

;(نام متد.this += new EventHandler(نام رویداد.نام کنترل

مثال: تخصیص متد ShowMessage به رویداد کلیک دکمه btnSample
یک متد به نام ShowMessage تعریف کنید که عمل موردنظر شمارا انجام دهد.
در اینجا برای راحتی کار فقط یک پیام نمایش می‌دهیم.

```
private void ShowMessage(object sender,EventArgs e)
{
    MessageBox.Show("Welcome");}
```

در رویداد Form1_Load رویداد Click کنترل btnSample را به صورت زیر تعریف کنید.

```
private void Form1_Load(object sender, EventArgs e)
{
    btnSample.Click += new EventHandler(this.ShowMessage);
}
```


کارگاه ۵ - رویدادهای ماوس و زمان سنج

هدف از این کارگاه آموزش نحوه کار با رویداد Tick مربوط به زمان سنج و نیز رویدادهای مستقل از کلید و وابسته به کلید در ماوس است. در چهار کارگاه قبلی هنرجو با برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویداد فشردن/رها کردن دکمه‌های ماوس و نیز برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویداد حرکت ماوس آشنا شده است. در این کارگاه هنرجو با برنامه‌ریزی اجرای یک فعالیت در صورت تحقق ترکیبی از رویدادهای دو حالت قبلی با اعمال محدودیت زمانی آشنا می‌شود و برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویدادهای مرتبط با حرکت ماوس در یک بازه زمانی معین را فرامی‌گیرد.

پاسخ به فعالیت‌ها

کنجکاوی
ص ۱۶۶

آیا می‌توان در بازی اعصاب سنج از رویدادهای Form_MouseMove یا Form_MouseHover برای بررسی خطای کاربر استفاده کرد؟

پاسخ: از رویداد MouseHover نمی‌توان استفاده کرد چون برای وقوع این رویداد باید چند لحظه ماوس روی فرم قرار گیرد. از رویداد MouseMove می‌توان استفاده کرد ولی بهتر است از رویداد MouseEnter استفاده شود.

اگر به جای رویداد MouseEventArgs از رویداد Click استفاده کنیم چه مشکلی پیش می‌آید؟ امکان تقلب

پاسخ: با توجه به کد ارائه شده در فیلم، در رویداد MouseEventArgs کار انجام شده است که شامل مقداردهی متغیر شروع، صفر نمودن زمان و فعال نمودن تایمر است، هر چند این سه دستور را می‌توان در رویداد Click هم قرار داد ولی نکته مهم این است که با توجه به منطق برنامه، ماوس در ابتدای کار باید بر روی دکمه شروع باشد. در صورت استفاده از رویداد Click کاربر می‌تواند ماوس را خارج از قسمت زرد رنگ (و در محیط فرم) قرار دهد و (چون در اینجا دکمه شروع به عنوان کنترل فعال از قبل انتخاب شده است) با فشردن دکمه‌های Space و یا Enter برنامه شروع به کار می‌کند و کاربر می‌تواند بدون گذر از مسیر زرد رنگ، ماوس را وارد برچسب پایان نماید و برنده شود.

چه تفاوتی بین رویداد Click و MouseEventArgs وجود دارد؟

پاسخ: رویداد MouseEventArgs فقط با کلیک ماوس رخ می‌دهد ولی اگر فوکوس (Focus) روی دکمه باشد رویداد Click با فشردن کلیدهای Enter و Space هم رخ می‌دهد. ضمن اینکه رویداد دارای آرگومان MouseEventArgs است که اطلاعاتی راجع به موقعیت ماوس، دکمه فشرده شده ماوس و ... می‌دهد.

چه تفاوتی بین منوهای بالا از لحاظ شکل و قالب و گروه‌بندی وجود دارد؟
 پاسخ: برخی از منوها مثل منوی Notepad یک منوی ساده بدون گروه‌بندی است.
 منوی شکل سمت راست منویی در ویندوز ۱۰ است که بارنگ و قالب ویژه از دیگر
 منوها متمایز شده است.
 در کدام شکل منو با راست‌کلیک فعال می‌شود؟
 پاسخ: شکل سمت راست (منو با زمینه مشکی و خاکستری) با راست‌کلیک فعال
 می‌شود.

چند نمونه از این عملیات را در جدول زیر بیان کنید و در صورت داشتن کلید
 میانبر آن را بنویسید.

| کلید میان‌بر | نوع عملیات با منو |
|--------------|------------------------------|
| Ctrl+C | کپی کردن فایل یا پوشه |
| Ctrl+P | گزینه Print در محیط توسعه VS |
| F7 | نمایش کدهای برنامه در VS |
| Ctrl+A | انتخاب همه |

کارگاه ۶ – ایجاد منو با کنترل MenuStrip

هدف از این کارگاه آموزش ایجاد منو به‌عنوان واسطه گرافیکی کاربر برای دسترسی به
 قابلیت‌های درونی نرم‌افزار است.
پاسخ به فعالیت‌ها

مرحله ۲: کنترل MenuStrip در کجا قرار می‌گیرد؟ پنجره پایین فرم
مرحله ۳: گزینه‌های دیگر منو، را در کجا می‌نویسید؟ در کنار یا پایین گزینه
مرحله ۴: با راست‌کلیک روی گزینه‌ها، چه گزینه‌های دیگری مشاهده می‌کنید؟
 با توجه به تصویر Enabled, Set Image, Checked و ...
مرحله ۵: روی گزینه‌ی «ذخیره» راست‌کلیک کنید و از گزینه Insert ،
 MenuItem را انتخاب کنید. چه تغییری در منو رخ می‌دهد؟ می‌توان یک گزینه
 جدید اضافه کرد.
 اگر بخواهید گزینه‌ای را حذف کنید، چگونه اقدام می‌کنید؟ با راست‌کلیک و
 انتخاب گزینه Delete
مرحله ۶: پنجره ویژگی‌های کنترل MenuStrip1 را باز کنید. مقدار ویژگی
 RightToLeft را Yes کنید. چه تغییری در منو رخ می‌دهد؟ جهت نمایش منو
 را از راست به چپ می‌کند.

فعالیت منزل
ص ۱۷۰

منوی نرم‌افزار Notepad را در یک پروژه طراحی کنید. برای انجام این فعالیت، کنترل MenuStrip را از جعبه‌ابزار به فرم اضافه کنید. سپس بر اساس یکی از روش‌های زیر نسبت به تکمیل فعالیت اقدام کنید.

نرم‌افزار Notepad ویندوز را اجرا کنید، با توجه به نرم‌افزار گزینه‌ها را اضافه کنید. خطوط جداکننده را در زیر منوهای File و Edit و Help فراموش نکنید. برخی از گزینه‌های غیرفعال را با کلیک سمت راست و برداشتن تیک Enabled غیرفعال کنید.

بر روی کنترل MenuStrip راست‌کلیک کنید و گزینه Insert Standard Items را انتخاب کنید. منوی ایجادشده را به‌دلخواه ویرایش و سفارشی کنید.

کارگاه ۷ – سفارشی کردن منو

هدف از این کارگاه آموزش ایجاد یک منوی سفارشی و دلخواه به‌عنوان واسطه گرافیکی کاربر برای دسترسی به قابلیت‌های درونی نرم‌افزار و نحوه ایجاد منوهای کاربردی خاص یک نرم‌افزار است.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۱۷۰

برای چند گزینه منو به‌دلخواه، متن راهنما (ToolTip) قرار دهید. پاسخ: گزینه موردنظر را انتخاب کنید و ویژگی ToolTipText را با متن دلخواه مقداردهی کنید.

کنجکاوی
ص ۱۷۱

آیا می‌توان جهت نمایش متن گزینه‌های منو را تغییر داد؟ پاسخ: بله، از پنجره ویژگی‌های منو، ویژگی RightToLeft را برابر Yes قرار دهید. جهت نمایش منو از راست به چپ می‌شود.

فعالیت کارگاهی
ص ۱۷۱

جدول ۴ ویژگی‌های متداول گزینه منو است. با توجه به کارگاه ۷ این جدول را تکمیل کنید.

| ویژگی | شرح |
|--------------|----------------------------|
| Checked | تیک دار کردن گزینه |
| Enabled | فعال یا غیرفعال کردن گزینه |
| Visible | نمایش یا عدم نمایش گزینه |
| Image | تصویر در کنار گزینه |
| RightToLeft | جهت نمایش راست به چپ |
| ShortCutKeys | کلید میانبر گزینه |
| ToolTipText | تعیین متن راهنما |

بررسی کنید که چه رویدادهای دیگری برای منو استفاده می‌شود و کاربرد آن‌ها چیست؟

پاسخ: کنترل منو و گزینه‌های آن همانند کنترل‌های دیگر دارای رویدادهای دیگری نظیر رویدادهای ماوس و صفحه‌کلید و ... هستند که در صورت نیاز می‌توان برای این رویدادها کدنویسی کرد.

کارگاه ۸ – رویداد برای گزینه‌های منو

هدف از این کارگاه آموزش ایجاد برنامه‌ریزی اجرای یک فرآیند مشخص در صورت انتخاب هر یک از گزینه‌های منوی نرم‌افزار است.

پاسخ به فعالیت‌ها

تکمیل کارگاه

ص ۱۷۲

مرحله ۴: عملکرد گزینه پاک چیست؟ محتویات کادر متن را پاک می‌کند.

مرحله ۵: پس از اجرای برنامه، با کلیک روی گزینه «سفید» چه رخ می‌دهد؟

پاسخ: رنگ زمینه کادر متن سفید و گزینه «سفید» تیک‌دار و تیک گزینه «زرد» برداشته می‌شود.

برای گزینه «زرد» از زیر منوی «رنگ زمینه»، همانند مرحله قبل، کدی بنویسید.

```
private void mnuYellow_Click(object sender, EventArgs e)
{
    txtPaper.BackColor = Color.Yellow;
    mnuWhite.Checked = false;
    mnuYellow.Checked = true; }

```

مرحله ۶: برای گزینه «قرمز» و «آبی» از زیر منوی «رنگ زمینه»، همانند مرحله قبل، کدی بنویسید.

```
private void mnuRed_Click(object sender, EventArgs e) {
    txtPaper.ForeColor = Color.Red;
    mnuBlack.Checked = false;
    mnuRed.Checked = true;
    mnuBlue.Checked = false;
}

```

مرحله ۷: متد رویداد کلیک گزینه فونت را بنویسید.

پاسخ: روی گزینه «فونت» دوبار کلیک کنید. کد تغییر فونت کادر متن txtPaper را با استفاده از کادر محاوره‌ای fontDialog1 بنویسید.

```
private void mnuFont_Click(object sender, EventArgs e)
{
    if (fontDialog1.ShowDialog() == DialogResult.OK)
        txtPaper.Font = fontDialog1.Font;
}

```

مرحله ۸: برای گزینه «خروج از برنامه» کد مناسب، را بنویسید.

```
private void mnuExit_Click(object sender, EventArgs e)
{
    Close();
}
```

فعالیت کارگاهی

ص ۱۷۳

پس از افزودن کنترل `MenuStrip` به فرم، روی آن راست کلیک کنید و گزینه `Insert Standard Items` را انتخاب کنید. در فرم برنامه چه مشاهده می کنید؟ پاسخ: پس از انتخاب گزینه `Insert Standard Items` یک منوی استاندارد همانند منوی برنامه `notepad` ویندوز ایجاد می شود که می توان به دلخواه آن را ویرایش کرد.

آیا پس از اجرای برنامه و کلیک روی گزینه ها، عملیاتی انجام می شود؟ خیر

کارگاه ۹ - طراحی منوی زمینه

هدف از این کارگاه آموزش ایجاد منوهای کمکی نرم افزار است، این منوها کاربرد وسیعی در واسط کاربری دارند. و در بسیاری از مواقع با راست کلیک کردن روی کنترل مورد نظر، نمایش داده می شوند. در سه کارگاه قبلی هنرجو با نحوه ایجاد، سفارشی کردن و برنامه ریزی رویدادهای منو آشنا شده است در این کارگاه با نحوه ایجاد و برنامه ریزی منوهای کمکی نرم افزار که گزینه های آنها عمدتاً وابسته به کنترل های موجود روی واسط کاربری هستند، آشنا می شود و برنامه ریزی اجرای یک فعالیت در صورت تحقق رویدادهای مرتبط با منو را فرا می گیرد.

پاسخ به فعالیت ها

تکمیل کارگاه

ص ۱۷۴

برنامه را اجرا کنید و روی کنترل جعبه متن راست کلیک کنید. چه چیزی مشاهده می کنید؟ منوی ایجاد شده در کارگاه قبل برای این که با راست کلیک روی فرم هم همین منو نمایش داده شود، چگونه عمل می کنید؟ ویژگی `ContextMenuStrip` فرم را برابر `contextMenuStrip1` قرار دهید.

کنجکاو

ص ۱۷۴

چه تفاوتی بین کنترل `MenuStrip` و `ContextMenuStrip` وجود دارد؟ پاسخ: از کنترل `MenuStrip` برای ایجاد منوهای اصلی و ثابت (معمولاً) در بالای فرم استفاده می شود. کنترل `ContextMenuStrip` برای ایجاد منوهای کمکی، دلخواه مرتبط با کنترل های برنامه استفاده می شود. *: با تغییر ویژگی `Dock` به `None` می توان موقعیت `MenuStrip` را به دلخواه تعیین کرد.

کارگاه ۱۰ – پروژه با منو

هدف از این کارگاه استفاده از مطالب آموخته شده کارگاه‌های قبلی در طراحی واسط کاربر و ایجاد نرم‌افزار با منو است. هنرجو در این کارگاه با نحوه ایجاد و برنامه‌ریزی منوهای نرم‌افزار آشنا می‌شود و برنامه‌ریزی اجرای یک فعالیت در صورت تحقق رویدادهای مرتبط با منو را فرا می‌گیرد.

پاسخ به فعالیت‌ها

قطعه کدی به برنامه اضافه کنید تا با کلیک روی هر تصویر اطلاعاتی مربوط به آن در `ToolTip` نمایش داده شود

پاسخ: برای افزودن `ToolTip` به برنامه، کافی است از جعبه ابزار کامپوننت `ToolTip` را انتخاب کنید. در این صورت یک شی به نام `ToolTip1` به نرم‌افزار اضافه می‌شود. برای تعیین متن راهنما به عنوان `ToolTip` برای یک کنترل می‌توانید از متد `SetToolTip` شی کمک بگیرید. در این متد پارامتر اول نام شی و پارامتر دوم متنی است که قرار است به عنوان راهنما نمایش داده شود.

مثال:

```
ToolTip1.SetToolTip(txtUserName, "نام کاربری را وارد کنید");
```

راه حل دوم استفاده مستقیم از کلاس `ToolTip` به صورت زیر است.

```
new ToolTip().SetToolTip(btnMember, "فرم مشخصات اعضا");
```

در صورت استفاده از این کد بهتر است در متد لود فرم و یا جایی نوشته شود که یکبار فراخوانی شود تا از ایجاد شی‌های متعدد جلوگیری شود.

تکمیل کارگاه
ص ۱۷۴

عملکرد علامت @ قبل از آدرس فایل چیست؟

پاسخ: اگر در یک ثابت رشته‌ای علامت \ وجود داشته باشد، همانند یک کاراکتر کنترلی عمل می‌کند و باید به صورت \\ نوشته شود. اگر در ابتدای ثابت رشته‌ای علامت @ قرار دهید نیازی به تکرار \ نیست. اگر ثابت رشته‌ای طولانی باشد به طوری که در یک سطر جا نشود هم می‌توان با @ این مشکل را حل کرد.

کنجکاوی
ص ۱۷۶

به کمک کلاس `SoundPlayer` برای گزینه «توقف سرود»، دستور توقف سرود را بنویسید.

```
private void mnuStop_Click(object sender, EventArgs e)
{
    System.Media.SoundPlayer player;
    player = new System.Media.SoundPlayer();
    player.Stop();
}
```

تکمیل کارگاه
ص ۱۷۴

یک کادر محاوره ای ColorDialog و FontDialog به فرم اضافه کنید و...

```
private void mnuFont_Click(object sender, EventArgs e){
    if (fontDialog1.ShowDialog() == DialogResult.OK){
        lblTitle.Font = fontDialog1.Font;
        lblDescription.Font = fontDialog1.Font;
    }
}
private void mnuBackColor_Click(object sender,
    EventArgs e){
    if (colorDialog1.ShowDialog() == DialogResult.OK)
        this.BackColor = ColorDialog1.Color;
}

```

برای تغییر رنگ کادر با توجه به نامگذاری برجسب ها در برنامه از کدی مشابه کد بالا استفاده کنید.

فعالیت منزل
ص ۱۷۷

قابلیتی به برنامه اضافه کنید تا با کلیک روی هر تصویر، تصویر در اندازه بزرگتر نمایش داده شود.

یک متد به نام myPic_Click ایجاد کنید و دستورات زیر را بنویسید.

```
private void myPic_Click(object sender, EventArgs e)
{
    //PictureBox p = (PictureBox)sender;
    PictureBox p = sender as PictureBox;
    if (p!=null)
    {
        int tag = (int) p.Tag;
        double factor = (tag == 0) ? 2 : 0.5;
        p.Width = (int) (p.Width * factor);
        p.Height = (int)(p.Height * factor);
        p.Tag = (tag != 0) ? 0 : 1;
        p.BringToFront();
    }
}

```

در متد Form1_Load مربوط به فرم برنامه دستورات زیر را بنویسید.

```
private void Form1_Load(object sender, EventArgs e)
{
    Pic1.Click += new EventHandler(myPic_Click);
    (برای تمامی عکس ها، کد بالا تکرار می شود.)
    pic1.Tag = pic2.Tag = pic3.Tag= pic4.Tag=0;
    pic5.Tag = pic6.Tag = pic7.Tag= pic8.Tag=0; }

```

واحد یادگیری ۸

شایستگی کار با صفحه کلید

تدریس

عملکرد صفحه کلید در ویندوز

قبل از شروع بحث لازم است ذهن هنرجو را به سمت عملکرد صفحه کلید به عنوان یک دستگاه ورودی و رویدادهای مرتبط با آن معطوف کرد.

صفحه کلید یکی از دستگاه‌های ورودی است که کاربر را قادر می‌سازد تا داده‌های موردنیاز را در اختیار نرم‌افزار قرار دهد، برای این منظور در صفحه کلید تعدادی کلید با عملکردهای متفاوتی وجود دارد، که می‌توان به کلیدهای الفبایی/ الفبا-عددی، کلیدهای نشان‌گذاری (کاما و نقطه و ...)، کلیدهای ناوبری و کلیدهای خاص (مثل Enter و Del و ESC و ...) اشاره کرد. در صفحه کلید ما با سه دسته رویداد روبه‌رو هستیم که عبارت‌اند از: پایین رفتن کلید (KeyDown)، رها شدن/بالا آمدن کلید (KeyUp)، فشردن کلی (KeyPress).

نکته مهم در خصوص رویدادهای صفحه کلید این است که رویدادهای پایین رفتن و رها شدن کلید برای همه کلیدها گزارش می‌شود، ولی رویداد فشردن کلید مخصوص کلیدهای الفبایی/ الفبا-عددی، کلیدهای نشان‌گذاری (کاما و نقطه و ...) و کلیدهای خاص (مثل Enter و Del و ESC و ...) است. در این دسته از کلیدها رویداد پایین رفتن کلید (KeyDown) در شروع سیکل فشرده شدن یک دکمه، و رویداد رها شدن کلید (KeyUp) در انتهای سیکل گزارش می‌شود و مادامی که کاربر کلید موردنظر را تحت فشار نگه می‌دارد، به صورت پیاپی رویداد فشردن (KeyPress) گزارش می‌شود.

مثال: یکی از کلیدهای الفبایی را فشرده و برای چند لحظه رها نکنید، در این صورت سیستم‌عامل گزارشی به صورت زیر به برنامه ارسال می‌کند (از چپ به راست بخوانید): KeyDown, KeyPress, KeyPress, ..., KeyPress, KeyUp

مشکلات متداول در فرآیند یادگیری - یاددهی

در برنامه‌نویسی رویدادگرا، توانایی تشخیص صحیح رویدادهای قابل‌برنامه‌ریزی دارای اهمیت ویژه‌ای است و شاید بتوان گفت اصلی‌ترین چالش مدیریت برنامه بر اساس وقوع رویدادهای مختلف است، در پایان این واحد یادگیری هنرجو باید به درک صحیحی از رویدادهای صفحه کلید و نحوه مدیریت آن‌ها رسیده باشد. همکاران گرامی باید دقت داشته باشند که فشردن یک دکمه از صفحه کلید سیستم باعث وقوع چندین رویداد خواهد شد و آنچه اهمیت دارد دانستن ترتیب وقوع این رویدادها و انتخاب رویداد متناسب با عملی است که باید انجام پذیرد.

شیوه و الگوی پیشنهادی

در شروع کار با طرح چندین مسئله سعی کنید تا هنرجو بتواند ترتیب وقوع رویدادهای صفحه کلید را شناسایی کند. سپس از وی بخواهید تا بر اساس وقوع رویدادها، فعالیتی را انجام دهد.

پاسخ به فعالیت‌ها

فعالیت کارگاهی
ص ۱۸۰

با فشردن کلیدهای زیر چه نوع نویسه‌ای ایجاد می‌شود؟

| نوع نویسه | کلید |
|--------------|-------|
| غیر چاپ‌شدنی | Home |
| غیر چاپ‌شدنی | Space |
| غیر چاپ‌شدنی | Enter |
| چاپ‌شدنی | A |
| چاپ‌شدنی | 0 |
| چاپ‌شدنی | # |

کارگاه ۱ - رویدادهای صفحه کلید برای فرم

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۱۸۱

شرح رویدادهای صفحه کلید در جدول ۵ آمده است. جدول را تکمیل کنید.

| نام رویداد | شرح رویداد | شرح رویداد در VS |
|------------|--|---|
| KeyDown | زمانی رخ می‌دهد که کلیدی فشرده شود. | Occurs when a key is first pressed. |
| KeyPress | زمانی رخ می‌دهد که فوکوس روی کنترل باشد و کاربر کلیدی را فشار دهد. | Occurs when the control has focus and the user presses and release a key. |
| KeyUp | زمانی رخ می‌دهد که کلید فشرده شده رها شود. | Occurs when a key is released. |

تکمیل کارگاه
ص ۱۸۱

مرحله ۲: برنامه را اجرا کنید. کلیدی را به دلخواه فشار دهید. چه اتفاقی می‌افتد؟ پاسخ: رنگ زمینه فرم قرمز می‌شود.

مرحله ۳: در متد رویداد Form1_KeyUp کدی بنویسید که رنگ زمینه‌ی فرم را آبی کند.

```
private void Form1_KeyUp(object sender, KeyEventArgs e)
{
    this.BackColor = Color.Blue;
}
```

برنامه را اجرا کنید. کلیدی را به دلخواه فشار دهید. چه اتفاقی می‌افتد؟ با فشردن کلید رنگ زمینه فرم قرمز و بارها کردن کلید رنگ زمینه فرم آبی می‌شود.

کارگاه ۲ – ترتیب رخداد رویدادهای صفحه کلید

پاسخ به فعالیت‌ها

مرحله ۴: برنامه را اجرا کنید. کلیدهای زیر وارد کنید و نتیجه را در جدول زیر بنویسید.

تکمیل کارگاه
ص ۱۸۲

| نتیجه | کلید |
|-------------------------|-------|
| KeyDown_KeyUp_ | Alt |
| KeyDown_KeyUp_ | Home |
| KeyDown_KeyPress_KeyUp_ | Space |
| KeyDown_KeyPress_KeyUp_ | Enter |
| KeyDown_KeyPress_KeyUp_ | A |
| KeyDown_KeyPress_KeyUp_ | 0 |
| KeyDown_KeyPress_KeyUp_ | # |

نتیجه حاصل از ورود نویسه‌های چاپ‌شدنی و غیرچاپ‌شدنی چیست؟ همه کلیدها نسبت به رویدادهای KeyDown و KeyUP واکنش نشان دادند ولی فقط کلیدهای Space,Enter,A,0,# نسبت به رویداد KeyPress واکنش نشان دادند.

مرحله ۵: عملکرد متد رویداد Form_KeyDown چیست؟
فعال‌سازی زمان‌سنج و تنظیم مقدار isUp به true برای شروع حرکت رو به بالای توپ در زمان وقوع رویداد Tick زمان‌سنج.

مرحله ۷: عملکرد timer1 در این برنامه چیست؟

حرکت دادن توپ به بالا یا پایین بر اساس مقدار متغیر isUp برنامه را اجرا کنید. یک کلید دلخواه را نگه‌دارید. چه اتفاقی می‌افتد؟
پاسخ: توپ به سمت به بالا حرکت می‌کند

کلید را رها کنید. چه اتفاقی می‌افتد؟ توپ به سمت به پایین حرکت می‌کند
مرحله ۸: برنامه را طوری تغییر دهید که توپ هنگام برخورد به دیواره بالای فرم به پایین برگردد.

```
private void timer1_Tick(object sender, EventArgs e)
{
    picBall.Top += (isUp) ? -5 : 5;
    isUp = (picBall.Top > 0) ? isUp : false;
}
```

مرحله ۹: برنامه را طوری تغییر دهید که توپ پس از رها کردن کلید در جایگاه اولیه‌اش متوقف شود.

یک متغیر به نام startPosition در نظر بگیرید که در ابتدای حرکت موقعیت توپ را در آن ذخیره کنید. سپس در زمان رها کردن کلید موقعیت Top را به مقدار موجود در متغیر startPosition تغییر دهید.

```
public partial class Form1 : Form
{
    bool isUp;
    int startPosition;

    private void Form1_KeyDown(object sender, KeyEventArgs e)
    {
        isUp = true;
        timer1.Enabled = true;
        startPosition = picBall.Top;
    }

    private void Form1_KeyUp(object sender, KeyEventArgs e)
    {
        picBall.Top = startPosition;
    }
}
```

کارگاه ۴ - آرگومان e در رویداد KeyDown و KeyPress

پاسخ به فعالیت‌ها

آیا نوع آرگومان e در رویداد KeyDown و KeyPress یکسان است و ویژگی‌های یکسانی دارند؟

کنجکاوی
ص ۱۸۵

پاسخ: خیر، در رویداد KeyDown آرگومان e از نوع KeyEventArgs و در رویداد KeyPress از نوع KeyPressEventArgs است.

کد اسکی (ASCII code) چیست؟

پاسخ: یکی از شیوه‌های رایج کدگذاری حروف، ارقام و علامت‌ها استفاده از کدهای اسکی (ASCII) است. واژه اسکی (ASCII) مخفف و کوتاه شده عبارت American Standard Code for Information Interchange است که به معنای "استاندارد کدگذاری آمریکایی برای تبادل اطلاعات" است.

کنجکاوی
ص ۱۸۵

مرحله ۴: برنامه را اجرا کرده، کلیدهای زیر را فشار دهید و نتایج را در جدول زیر بنویسید.

تکمیل کارگاه
ص ۱۸۵

| کد اسکی | نویسه | نام کلید |
|-----------|--------------------|----------------|
| 66 | B | B |
| 66 | b | B |
| 189 و 187 | OemMinus و Oemplus | + و - |
| 109 و 107 | Subtract و Add | + و - بخش عددی |
| 16 | ShiftKey | Shift |

آیا در این برنامه، نتیجه برای حروف کوچک و بزرگ یکسان است؟ بله
آیا رویداد KeyDown و KeyUp برای تشخیص حروف الفبای کوچک و بزرگ مناسب است؟ خیر

کدام ویژگی آرگومان e کد اسکی نویسه را مشخص می‌کند؟ KeyValue
مرحله ۵: ویژگی keyData را برای مرحله ۳، آزمایش کنید.

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    lblCharacter.Text = e.KeyData.ToString();
    lblCode.Text = e.KeyValue.ToString();
}
```

برنامه را اجرا کرده، کلیدهای جدول مرحله ۴ را فشار دهید و نتایج را در جدول دیگری بنویسید.

| نام کلید | نویسه | کد اسکی |
|----------------|--------------------|-----------|
| B | B | 66 |
| B | b | 66 |
| + و - | OemMinus و Oemplus | 189 و 187 |
| + و - بخش عددی | Subtract و Add | 109 و 107 |
| Shift | ShiftKey | 16 |

برای شناسایی کلید فشرده شده در رویداد KeyPress از چه ویژگی استفاده می‌کنید؟ نوع داده‌ی آن چیست؟
 پاسخ: ویژگی KeyChar از آرگومان e، نوع داده KeyChar از نوع char است.
 مرحله ۷: آیا رویداد KeyPress برای تشخیص حروف الفبای کوچک و بزرگ مناسب است؟ بله
 آیا کد کلیدهای Shift و جهت‌دار نمایش داده می‌شود؟ خیر

فعالیت منزل
ص ۱۸۷

برنامه‌ای بنویسید که با فشردن کلیدهای فارسی، کد حروف فارسی نمایش داده شود.

پاسخ: برنامه‌ای همانند کارگاه ۴ ایجاد کنید. و متد رویداد Form1_KeyPress را همانند کد زیر ایجاد کنید. و برنامه را اجرا کنید و صفحه‌کلید را در حالت فارسی قرار دهید.

```
private void Form1_KeyPress(object sender,
KeyPressEventArgs e)
{
    lblCharacter.Text = e.KeyChar.ToString();
    int code = (int)e.KeyChar;
    lblCode.Text = code.ToString();
}
```

کدهای حروف فارسی به صورت یونیکد در برچسب lblCode قرار می‌گیرد.

کارگاه ۵ - واکنش برنامه به فشردن کلیدهای ترکیبی

پاسخ به فعالیت‌ها

متغیر Ratio چه مقداری می‌گیرد؟

پاسخ: نسبت عرض به طول

کنجکاوی
ص ۱۸۹

اعداد 187 و 189 کد اسکی چه کلیدهایی هستند؟

پاسخ: کلید + و - ردیف بالای صفحه کلید

عملکرد متد رویداد Form1_KeyDown چیست؟

پاسخ: با گرفتن کلید CTRL و کلید +، پنج واحد به طول تصویر اضافه می‌شود و با گرفتن CTRL و کلید -، پنج واحد از طول کم می‌شود و در نهایت عرض تصویر متناسب با طول جدید تصویر محاسبه می‌شود تا تصویر کشیده نشود.

تکمیل کارگاه

ص ۱۸۹

مرحله ۴: برنامه را طوری تغییر دهید که با کلیدهای + و - در قسمت عددی صفحه کلید، تصویر بزرگ و کوچک شود.

```
double Ratio=(double)picSample.Width / picSample.Height;
if (e.Control == true)
{
    if (e.KeyValue == 187 || e.KeyValue==107)
        picSample.Height += 5;
    else if (e.KeyValue == 189 || e.KeyValue==109)
        picSample.Height -= 5;
    picSample.Width = (int)(Ratio * picSample.Height);
}
```

مرحله ۵: برنامه را طوری تغییر دهید که با تغییر اندازه تصویر، کادر تصویر در وسط فرم قرار گیرد.

```
private void Form1_Load(object sender, EventArgs e)
{
    picSample.Top = (this.Height - picSample.Height) / 2;
    picSample.Left = (this.Width - picSample.Width) / 2;
}
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    double Ratio=(double)picSample.Width/picSample.Height;
    if (e.Control) // e.Control == true
    {
        if(e.KeyValue==Keys.Oemplus || e.KeyValue== Keys.Add)
            picSample.Height += 5;
        else if (e.KeyValue == Keys.OemMinus
|| e.KeyValue== Keys.Subtract)
            picSample.Height -= 5;
        picSample.Width = (int)(Ratio * picSample.Height);
    }//if
    picSample.Top = (this.Height - picSample.Height) / 2;
    picSample.Left = (this.Width - picSample.Width) / 2;
}
```

مرحله ۶: برنامه را طوری تغییر دهید که با تغییر اندازه‌ی تصویر، ارتفاع تصویر حداکثر ۳۰۰ و حداقل ۵۰ پیکسل شود.

```
bool flag1=e.KeyValue==Keys.Oemplus || e.KeyValue ==
Keys.Add;
bool flag2= e.KeyValue == Keys.OemMinus
|| e.KeyValue == Keys.Subtract;
if( picSample.Height <= 300+5 && flag1)
    picSample.Height += 5;
else if ( picSample.Height >=50+5 && flag2)
    picSample.Height -= 5;
```

فعالیت منزل
ص ۱۹۰

برنامه کارگاه ۵ را با استفاده از ویژگی KeyData بنویسید.

```
double aspect=(float)picSample.Width / picSample.Height;
if (e.KeyData.ToString()=="Oemplus, Control")
    picSample.Height += 5;
else if (e.KeyData.ToString()=="OemMinus, Control")
    picSample.Height -= 5;
picSample.Width = (int)(aspect * picSample.Height);
```

کارگاه ۶- فوکوس (Focus)

پاسخ به فعالیت‌ها

کنجکاوی
ص ۱۹۰

اگر کلیدی حرفی تایپ کنید، آیا متنی در کادر متن نوشته می‌شود؟ بله
کلید Tab را فشار دهید. اگر کلیدی حرفی تایپ کنید متنی در کادر متن نوشته
می‌شود؟ خیر
اگر کلیدی فشار دهید، آیا کد اسکی و نویسه نمایش داده می‌شوند؟ آیا رویداد
Form_KeyDown رخ می‌دهد؟ خیر

کنجکاوی
ص ۱۹۱

آیا در مرحله ۴ باید مقدار ویژگی KeyPreview true شود؟ چرا؟
پاسخ: خیر، چون کنترل برجسب فوکوس نمی‌گیرد.
برای تغییر فوکوس در برنامه از چه کلیدی استفاده می‌کنید؟ کلید Tab
در هنگام برنامه‌نویسی از چه کدی برای تغییر فوکوس استفاده می‌کنید؟
پاسخ: متد Focus() مثل TextBox1.Focus();

در کارگاه ۷ برنامه را تغییر دهید تا با فشردن کلیدهای جهت‌دار، سفینه به صورت خودکار در همان جهت حرکت کند. و با برخورد به دیواره‌ی فرم در جهت خلاف برگردد. از زمان‌سنج استفاده کنید
متغیر key را در یخس کلاس فرم تعریف کنید.

```
public partial class Form1 : Form
{
    Keys key;
    //متغیر key را در متد رویداد Form1_KeyDown مقاردهی کنید.
    private void Form1_KeyDown(object sender, KeyEventArgs e)
    {
        key = e.KeyCode;
        timer1.Enabled = true;
    }
    //در متد رویداد timer1_Tick کادر تصویر سفینه را حرکت دهید. و با برخورد به
    //دیواره فرم در چهار جهت برمی‌گردد.
    private void timer1_Tick(object sender, EventArgs e)
    {
        switch (key)
        {
            case Keys.Left:
            case Keys.A:
                picSpacecraft.Left -= 5; break;
            case Keys.Right:
            case Keys.D:
                picSpacecraft.Left += 5; break;
            case Keys.Up:
            case Keys.W:
                picSpacecraft.Top -= 5; break;
            case Keys.Down:
            case Keys.S:
                picSpacecraft.Top += 5; break;
        } //switch
        if (picSpacecraft.Left <= 0) key = Keys.Right;
        if (picSpacecraft.Left + picSpacecraft.Width >=
            this.ClientRectangle.Width)
            key = Keys.Left;
        if (picSpacecraft.Top <= 0) key = Keys.Down;
        if (picSpacecraft.Top + picSpacecraft.Height >=
            this.ClientRectangle.Height)
            key = Keys.Up;
    }
}
```


برنامه‌ای بنویسید که در آن پنجره‌ی برنامه، با فشردن کلید ESC بسته شود.

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Escape) Close();
}
```

کارگاه ۸ – رویداد KeyDown کنترل کادر متن

پاسخ به فعالیت‌ها

عملکرد متد رویداد txtFirstName_KeyDown چیست؟

کنجکاوی
ص ۱۹۴

پاسخ: در صورت فشردن دکمه Enter در کادر متن txtFirstName، فوکوس را به کنترلر txtLastName منتقل می‌کند. و کنترلر txtLastName به‌عنوان کنترلر فعال انتخاب می‌شود.

تکمیل کارگاه
ص ۱۹۳

مرحله ۴: متد رویداد KeyDown را برای کنترل‌های کادر متن دیگر هم بنویسید.

```
private void txtLastName_KeyDown(object sender,
    KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter) txtFatherName.Focus();
}

private void txtFatherName_KeyDown(object sender,
    KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter) txtMelliCode.Focus();
}

private void txtMelliCode_KeyDown(object sender,
    KeyEventArgs e)
{
    // فوکوس روی دکمه ثبت
    if (e.KeyCode == Keys.Enter) btnSabt.Focus();
}
```

مرحله ۵: آخرین کنترل‌ری که فوکوس دریافت می‌کند، کدام است؟ دکمه ثبت

پژوهش
ص ۱۹۴

اگر بخواهیم با فوکوس روی هر کادر متن رنگ زمینه آن تغییر کند و با از دست دادن فوکوس رنگ آن به حالت اول درآید از چه رویدادهایی استفاده می‌کنیم؟

پاسخ: از رویداد Enter برای تشخیص رسیدن فوکوس به کنترلر و از رویداد Leave برای تشخیص از دست دادن فوکوس می‌توان استفاده کرد.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۱۹۵

مرحله ۳: در کادر متن شماره دانش‌آموزی به‌دولخواه کلیدی را تایپ کنید. چه کلیدهایی در کادر متن شماره دانش‌آموزی تایپ می‌شوند؟

پاسخ: هیچ نویسه‌ای نوشته نمی‌شود.

عملکرد ویژگی SuppressKeyPress چیست؟

پاسخ: این ویژگی ارسال یا عدم ارسال رویداد کلید به کنترل‌های زیرمجموعه را کنترل می‌کند، با true شدن این ویژگی می‌توان از ورود داده از طریق صفحه‌کلید توسط کاربر جلوگیری کرد.

مرحله ۴: لزوم تبدیل e.KeyCode به char چیست؟

پاسخ: برای تبدیل از نوع Keys به char، چون نوع ورودی char.IsDigit است.

برنامه را دوباره اجرا کنید و کلیدهایی را به‌دولخواه در کادر متن شماره دانش‌آموزی تایپ کنید. چه کلیدهایی در کادر متن شماره دانش‌آموزی تایپ می‌شوند؟

پاسخ: فقط می‌توان ارقام صفر تا نه را وارد کرد.

آیا می‌توان داده درون کادر متن را با کلید Delete یا backspace پاک کرد؟ خیر

آیا می‌توان با کلیدهای جهت‌دار، داده را پیمایش کرد؟ خیر

مرحله ۵: کد مرحله ۴ را به‌گونه‌ای تغییر دهید که بتوان عدد ورودی را حذف و پیمایش کرد.

```
private void txtStudentNO_KeyDown(object sender,
    KeyEventArgs e)
{
    bool isSpecialKey= e.KeyCode==Keys.Back
        || e.KeyCode==Keys.Delete
        || e.KeyCode==Keys.Right || e.KeyCode==Keys.Left;
    bool isDigit= char.IsDigit((char)e.KeyCode);
    e.SuppressKeyPress=! ( isDigit || isSpecialKey);
}
```

مرحله ۶: چه نویسه‌هایی در کادر متن تایپ می‌شود؟

پاسخ: فقط حروف الفبا در کادر متن txtName نوشته می‌شوند و می‌توان با BackSpace آن‌ها را پاک کرد.

مرحله ۸: برنامه را طوری تغییر دهید که بتوان حروف تایپ‌شده را پاک و پیمایش کرد.

```
private void txtName_KeyPress(object sender,
    KeyPressEventArgs e)
{
```

```

bool isSpecialKey= e.KeyCode==Keys.Back ||
e.KeyCode==Keys.Delete || e.KeyCode==Keys.Right ||
e.KeyCode==Keys.Left;
bool isLower= char.IsLower(e.KeyChar);
e.Handled =!( isLower || isSpecialKey);
}

```

مرحله ۹: برنامه را اجرا کرده، نام خانوادگی Ahmadi را در کادر متن وارد کنید. چه متنی در کادر متن نوشته می‌شود؟ AHMADI
 عملکرد رویداد txtName_KeyPress چیست؟
 پاسخ: تبدیل حروف کوچک الفبای لاتین به حروف بزرگ در کادر متن

کنجکاوی
ص ۱۹۶

عملکرد ویژگی Handled چیست؟

پاسخ: همانند عملکرد ویژگی SuppressKeyPress در رویداد KeyDown است.

فعالیت کارگاهی
ص ۱۹۷

نتیجه خروجی مربوط به مثال‌ها را در جدول ۷ بنویسید.

| مثال ۲ | مثال ۱ |
|---------------------|--------------------|
| IsDigit('5')=true | IsDigit('A')=false |
| IsLetter('@')=false | IsLetter('A')=true |
| IsUpper('M')=true | IsUpper('d')=false |
| IsLower('M')=false | IsLower('d')=true |

کنجکاوی
ص ۱۹۸

ثابت رشته‌ای چیست؟

پاسخ: رشته‌ای است که در طول برنامه مقدار آن قابل تغییر نیست و تنها برای مقادیردهی اولیه متغیرهای رشته‌ای استفاده می‌شود.

نوع داده name[0] چیست؟ پاسخ: char

آیا می‌توان این نویسه را مقداردهی کرد؟ پاسخ: خیر

چه عملیاتی در واژه‌پردازها روی رشته‌ها انجام می‌شود؟

پاسخ: جستجو، جایگزینی، انتخاب

نرم‌افزارهایی را نام ببرید که داده‌های رشته‌ای در آن استفاده می‌شود؟

پاسخ: Word، Excel، سامانه سناد، کلیه نرم‌افزارهای اداری و کاربردی

کارگاه ۱۰ – بررسی متدهای رشته‌ای

پاسخ به فعالیت‌ها

مرحله ۳: برنامه را اجرا کنید، چه مقداری در کادر متن قرار می‌گیرد، ...
مقدار ۷ در کادر متن قرار می‌گیرد، ویژگی Length طول رشته را برمی‌گرداند
مرحله ۴: چه مقداری در کادر متن قرار می‌گیرد؟ ALIREZA
عملکرد متد ToUpper چیست؟ رشته را با حروف بزرگ برمی‌گرداند.
مرحله ۵: عملکرد متد IndexOf چیست؟ مکان زیررشته Reza را در محتوای موجود در متغیر name جستجو می‌کند.
اگر بجای 0، عدد 5 قرار دهید، خروجی چه می‌شود؟ مکان زیررشته Reza را از موقعیت پنجم، جستجو می‌کند.
اگر در مرحله ۵ به جای Reza، واژه Ahmad را بنویسید، خروجی چه می‌شود؟ اگر زیررشته از موقعیت مورد نظر در رشته موجود نباشد خروجی منفی یک است.

تکمیل کارگاه
ص ۱۹۹

عملکرد متد LastIndexOf چیست؟

پاسخ: اندیس آخرین موقعیت زیر رشته جستجو شده را در رشته مورد جستجو برمی‌گرداند، اگر زیر رشته در رشته مورد نظر یافت نشود مقدار ۱- برمی‌گرداند.

کنجکاوی
ص ۱۹۹

کارگاه ۱۱ – کار با متد Compare و Trim

پاسخ به فعالیت‌ها

اگر کاربر در کادر متن نام کاربری واژه ADMIN را وارد کند، چه پیامی دریافت می‌کند؟ پاسخ: نام کاربری یا گذرواژه اشتباه است.

کنجکاوی
ص ۲۰۰

۴- برنامه را به‌گونه‌ای ویرایش کنید که ورود حروف بزرگ با کوچک در نام کاربری یا گذرواژه، تاثیری در اعتبارسنجی کاربر نداشته باشد.

```
private void btnLogin_Click(object sender, EventArgs e)
{
    string user=txtUserName.Text.Trim().ToLower();
    string pass=txtPassword.Text.ToLower();
    bool okUserPass = user.CompareTo("admin") == 0;
    okUserPass &= pass.CompareTo("123")==0;
    if (okUserPass)
        MessageBox.Show("ورود شما با موفقیت انجام شد");
    else
        MessageBox.Show("نام کاربری یا گذرواژه اشتباه است");
}
```

کارگاه ۱۱
ص ۲۰۰

کارگاه ۱۲ - به کارگیری متدهای Replace و IndexOf

پاسخ به فعالیت‌ها

فعالیت کارگاهی
ص ۲۰۱

ستون «عملکرد» و «نوع داده خروجی» را تکمیل کنید.
اگر "AliReza" sample= باشد، حاصل دستورات در ستون «مثال» را بنویسید.

| نام متد | عملکرد | نوع داده خروجی | مثال |
|-----------|--|----------------|----------------------|
| ToUpper | همه حروف رشته را به بزرگ برمی‌گرداند. | string | ALIREZA |
| ToLower | همه حروف رشته را به کوچک برمی‌گرداند. | string | alireza |
| IndexOf | اولین مکان زیررشته را از موقعیت خاصی، برمی‌گرداند. | string | 3 |
| Replace | رشته جدیدی با جایگزینی زیررشته یا کاراکترها به وجود می‌آورد. | string | MohammadReza |
| Remove | رشته جدیدی با حذف بخشی از رشته به وجود می‌آورد. | string | Alza |
| Insert | رشته جدیدی با اضافه کردن زیررشته به وجود می‌آورد. | string | AliRMohammadeza |
| Format | رشته جدیدی با اضافه کردن زیررشته به وجود می‌آورد. | string | AliReza is a student |
| CompareTo | یک رشته را با رشته فعلی مقایسه می‌کند. | Int | 1 |
| Trim() | حذف فواصل خالی از ابتدا و انتهای رشته | string | Ali |

پس از تدریس

الف) فعالیت‌های تکمیلی

پیشنهاد می‌شود پس از تدریس

با انجام کلیه تکالیف ارائه شده در هر یک از کارگاه‌ها، مشکلات ناشی از عدم درک صحیح مطالب در همان جلسه شناسایی شود و در صورت امکان برطرف شود.
از هنرجویان مستعد به عنوان معلم‌یار در آموزش موضوع برای هنرجویان ضعیف‌تر استفاده شود.

هنرجویان را ترغیب کنید که به صورت گروهی به انجام تکالیف‌های مقرر در زمان محدود اقدام کنند. برای این کار بهتر است از روش زمان‌سنج معکوس و کاهش امتیاز استفاده کنید.

متناسب با هدف این پودمان، لازم است هنرآموزان محترم ضمن توجه به سطح درک هنرجویان از برنامه‌نویسی، و توانایی آنها در ترکیب دانسته‌های قبلی، اقدام به ارائه کاربردهای ملموس‌تر (با استفاده از آموخته‌های این واحد یادگیری) کنند، تا علاوه بر ایجاد نشاط و تقویت خودباوری در آنها، میزان شایستگی کسب‌شده توسط هر هنرجو در کلاس را تشخیص دهند.

برای بالابردن درک هنرجو از رویدادها، از تدریس بازی‌محور و درگیرکردن هنرجویان در یک بازی، استفاده کنید.

یکی از کارهایی که هنرجویان در این پودمان می‌توانند انجام دهند، ایجاد یک نرم‌افزار ساده برای استفاده توسط هنرجویان پایه دهم و پایه یازدهم می‌باشد، برنامه می‌تواند دارای منویی برای کد نویسی برنامه‌های کنسول، کدهای HTML و یا PHP باشد. از هنرجو بخواهید در نرم‌افزار منویی با گزینه جدید داشته باشد که با انتخاب گزینه‌های آن، ساختار یک برنامه کنسول، یک سند ساده HTML و یا ساختار یک برنامه PHP نمایش داده شود و کاربر بتواند دستورات مورد نظرش را در آن بنویسد. در پودمان بعدی می‌توانید از هنرجو بخواهید که با توسعه برنامه، کد نوشته شده را در یک فایل ذخیره کند.

یکی دیگر از کارهایی که می‌توانید از هنرجو بخواهید، حرکت دادن یک کنترل یا یک تصویر با استفاده از ماوس و یا صفحه کلید در جهت‌های مختلف باشد.

ب) نمونه ارزشیابی پایانی

پیش‌آگاهی:

- برای تبدیل از مبنای ۱۰ به ۲ و یا از ۲ به ۱۰ می‌توانید علاوه بر آنچه در کتاب دانش فنی دهم آموخته‌اید از Convert نیز کمک بگیرید. برای مثال خروجی دستور `Convert.ToString(21, 2)` برابر ۱۰۱۰۱ است که رشته دودویی معادل عدد ۲۱ می‌باشد و یا خروجی دستور `Convert.ToInt16("1001", 2)` برابر ۹ است که معادل مبنای ۱۰ رشته دودویی ۱۰۰۱ می‌باشد.
- در نمونه سؤال این ارزشیابی، ضمن توجه به استانداردهای ارزشیابی، به منظور افزایش دقت سنجش شایستگی، امتیازبندی به شکل ریز و براساس بخش‌های مختلف هر سؤال صورت گرفته است و در نهایت نتیجه به صورت نمره ۱، ۲ یا ۳ ارائه شده است، بدیهی است این یک نمونه از سؤال ارزشیابی است و تعمیم این مدل به سوالات سایر ارزشیابی‌ها مد نظر طراح نیست.

نمونه سؤال ارزشیابی پایانی صفحه ۱۷۸ کتاب درسی

الف) به کمک یک زمان سنج، ده کنترل برچسب، یک کادر متن و یک دکمه برنامه‌ای بنویسید که خواسته‌های زیر را برآورده سازد:

- ۱) رنگ زمینه همه برچسب‌ها در ابتدا خاکستری باشد (۱ امتیاز). مقدار ویژگی Text در کادر متن برابر صفر (۰) باشد (۱ امتیاز) و عنوان دکمه «نمایش دودویی» باشد (۱ امتیاز)
- ۲) متد کلیک یکی از برچسب‌ها را به گونه‌ای برنامه‌نویسی کنید که با هر بار کلیک بر روی آن رنگ زمینه آن به رنگ سبز/خاکستری (سبز به خاکستری و خاکستری به سبز) تغییر کند (۱ امتیاز). سپس این متد را به سایر برچسب‌ها نیز تخصیص دهید. (اگر تخصیص متد به سایر برچسب‌ها ایستا باشد ۱ امتیاز و اگر با مقیدسازی پویا باشد ۲ امتیاز)
- ۳) با ورود نشانگر ماوس به محدوده برچسب، آیکون ماوس به شکل دست تغییر داده شود (۱ امتیاز)، و در صورت نگه داشتن ماوس روی هر یک از برچسب‌ها، عبارت «برای تغییر رنگ برچسب کلیک کنید» به عنوان متن راهنما نمایش داده شود. (۲ امتیاز)
- ۴) با فرض اینکه رنگ سبز معادل رقم ۱ و رنگ خاکستری معادل رقم ۰ در مبنای ۲ باشد، متد کلیک مرحله قبل را به گونه‌ای تغییر دهید که وقتی کاربر روی هر کدام از برچسب‌ها کلیک کند، هم‌زمان با تغییر رنگ زمینه برچسب، عدد معادل مبنای ۱۰ در کادر متن نمایش داده شود. (۳ امتیاز)
- ۵) کاربر بتواند یک عدد مبنای ۱۰ را در کادر متن وارد نماید سپس:
 - با کلیک بر روی دکمه نمایش دودویی معادل مبنای ۲ آن را با استفاده از تغییرات رنگ زمینه برچسب‌ها (سبز/خاکستری) مشاهده کند. (۱ امتیاز)
 - اگر کاربر عددی را وارد کرد که در تعداد ۱۰ برچسب قابل نمایش نباشد پیام «عدد کوچکتری را وارد کنید» نمایش دهد. (۱ امتیاز)
 - کاربر فقط بتواند اعداد صحیح مثبت را وارد کند. (۱ امتیاز)

ارزشیابی نهایی هنرجو در بخش الف (واکنش به رویدادهای ماوس)

| سؤال | ۱ | ۲ | ۳ | ۴ | ۵ | جمع کل نمره در بخش الف |
|-----------------------|---|---|---|---|---|-------------------------|
| حداقل نمره قبولی از ۳ | ۳ | ۲ | ۲ | ۱ | ۱ | الف |
| **نمره هنرجو از ۳ | | | | | | **نمره نهایی هنرجو از ۳ |

**چنانچه هنرجو نتواند حداقل نمره هر سؤال را کسب کند، نمره وی در آن بخش برابر ۱ منظور شود.

**اگر مجموع نمرات هنرجو کمتر از ۱۰ باشد، پایین‌تر از حد انتظار (نمره ۱ از ۳)، بزرگتر از ۱۰ و کمتر از ۱۳ باشد در حد انتظار (نمره ۲ از ۳) و بزرگتر و یا مساوی ۱۳ باشد فراتر از انتظار (نمره ۳ از ۳) ارزشیابی شود.

ب) افزودن منوی نواری به برنامه

۱) به برنامه یک منو با گزینه‌های «بازنشانی»، «توقف» و «خروج» اضافه کنید (۳ امتیاز - افزودن صحیح هر گزینه یک امتیاز)

۲) در صورت انتخاب گزینه «بازنشانی»، رنگ برچسب‌ها، مقدار کادرمتن و زمان به حالت اولیه (زمان شروع) برگردد. در این وضعیت برنامه زمان را از صفر شروع کند. (۳ امتیاز - برای بازنشانی رنگ برچسب‌ها، بازنشانی مقدار کادرمتن و بازنشانی زمان هر کدام یک امتیاز منظور می‌شود.)

راهنمایی: زمان سپری شده را در عنوان فرم برنامه نمایش دهید.

۳) در صورت انتخاب گزینه «توقف»، عنوان گزینه به «ادامه» تغییر کند، سپس شمارش زمان متوقف شود و اجازه ورود داده یا تغییر رنگ برچسب‌ها از کاربر سلب شود. (۳ امتیاز)

۴) در صورت انتخاب گزینه «ادامه»، عنوان گزینه به «توقف» تغییر کند، سپس شمارش زمان شروع شود و اجازه ورود داده یا تغییر رنگ برچسب‌ها به کاربر داده شود. (۳ امتیاز)

۵) در صورت انتخاب گزینه «خروج» از کاربر پرسیده شود که «آیا مایل به خروج از برنامه است؟» و در صورت پاسخ مثبت کاربر، فرم برنامه بسته شود. در غیر این صورت عملی انجام نشود. (۳ امتیاز)

ارزشیابی نهایی هنرجو در بخش ب (منوی نواری)

| سؤال | ۱ | ۲ | ۳ | ۴ | ۵ | جمع کل نمره در بخش ب |
|-----------------------|---|---|---|---|---|-------------------------|
| حداقل نمره قبولی از ۳ | ۱ | ۱ | ۱ | ۱ | ۱ | |
| **نمره هنرجو از ۳ | | | | | | **نمره نهایی هنرجو از ۳ |

**چنانچه هنرجو نتواند حداقل نمره هر سؤال را کسب کند، نمره وی در آن بخش برابر ۱ منظور شود.

**اگر مجموع نمرات هنرجو کمتر از ۱۰ باشد، پایین‌تر از حد انتظار (نمره ۱ از ۳)، بزرگتر از ۱۰ و کمتر از ۱۳ باشد در حد انتظار (نمره ۲ از ۳) و بزرگتر و یا مساوی ۱۳ باشد فراتر از انتظار (نمره ۳ از ۳) ارزشیابی شود.

ج (افزودن منوی زمینه به برنامه

تغییراتی در برنامه ایجاد کنید تا با راست‌کلیک کردن بر روی فرم یک منوی زمینه با همان گزینه‌های مربوط به منوی بخش ب داشته باشیم که همان عملکردها را داشته باشد. (۳ امتیاز)

ارزشیابی نهایی هنرجو در بخش ج (منوی زمینه)

| سؤال | ۱ | ۲ | ۳ | ۴ | ۵ | جمع کل نمره در بخش ج |
|-----------------------|---|---|---|---|---|-------------------------|
| حداقل نمره قبولی از ۳ | ۱ | ۱ | ۱ | ۱ | ۱ | |
| **نمره هنرجو از ۳ | | | | | | **نمره نهایی هنرجو از ۳ |

**چنانچه هنرجو نتواند حداقل نمره هر سؤال را کسب کند، نمره وی در آن بخش برابر ۱ منظور شود.

**اگر مجموع نمرات هنرجو کمتر از ۱۰ باشد، پایین‌تر از حد انتظار (نمره ۱ از ۳)، بزرگتر یا مساوی ۱۰ و کمتر از ۱۳ باشد، در حد انتظار (نمره ۲ از ۳) و بزرگتر و یا مساوی ۱۳ باشد فراتر از انتظار (نمره ۳ از ۳) ارزشیابی شود.

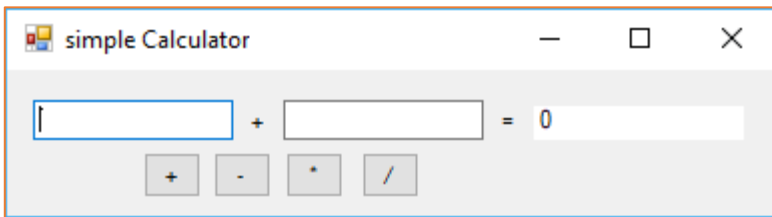
ارزشیابی نهایی هنرجو از هر ۳ بخش آزمون

| نمره نهایی هنرجو از هر بخش از آزمون | الف | ب | ج | جمع کل نمره هنرجو |
|-------------------------------------|-----|---|---|-------------------------|
| حداقل نمره قبولی از ۳ | ۲ | ۱ | ۱ | |
| **نمره هنرجو از ۳ | | | | **نمره نهایی هنرجو از ۳ |

**اگر نمره هنرجو در بخش الف کمتر از ۲ باشد، نمره نهایی وی پایین‌تر از حد انتظار (نمره ۱ از ۳) خواهد بود. در غیر اینصورت اگر مجموع نمرات هنرجو کمتر از ۴ باشد، پایین‌تر از حد انتظار (نمره ۱ از ۳)، بزرگتر یا مساوی ۴ و کمتر از ۹ باشد در حد انتظار (نمره ۲ از ۳) و در غیر این صورت فراتر از انتظار (نمره ۳ از ۳) ارزشیابی شود.

نمونه سؤال ارزشیابی پایانی صفحه ۲۰۴ کتاب درسی

میخواهیم یک ماشین حساب ساده طراحی کنیم که شامل دو کادر متن به نامهای txtA و txtB برای ورود دو عدد و یک برجسب به نام lblOP برای نمایش عمل انتخاب شده توسط کاربر و یک برجسب به نام lblResult برای نمایش نتیجه و چهار دکمه برای چهار عمل اصلی به نامهای btnAdd, btnSubtract, btnDiv, و btnMul باشد.



برنامه را طوری بنویسید که با کمترین کدنویسی بتوانید این عملیات را انجام دهید.
پاسخ: بخشی از برنامه در برنامه ماشین حساب به جای اینکه برای همه‌ی دکمه‌ها رویداد Click را بنویسید، فقط برای دکمه btnAdd بنویسید و سپس متد کلیک این دکمه را به عنوان رویداد کلیک بقیه دکمه‌ها هم تعریف کنید.

```
public Form1()
{
    InitializeComponent();
    this.btnMul.Click += new
System.EventHandler(this.btnAdd_Click);
    this.btnDiv.Click += new
System.EventHandler(this.btnAdd_Click);
    this.btnSubtract.Click += new
System.EventHandler(this.btnAdd_Click);
}
```

کد رویداد مربوط به دکمه btnAdd را به صورت زیر تعریف کنید.

```
private void btnAdd_Click(object sender, EventArgs
e)
{
    double a=double.Parse(txtA.Text);
    double b=double.Parse(txtB.Text);
    double c=0;
    Button btn=sender as Button;
```

```

if (btn!=null)
{
    lblOP.Text=btn.Text;
    if (btn.Equals(btnAdd)) c=a+b;
    else if (btn.Equals(btnSubtract)) c=a-b;
    else if (btn.Equals(btnMul)) c = a *b;
    else if (btn.Equals(btnDiv)) c = a / b;
    lblResult.Text = c.ToString();
}
}

```

(۱) به برنامه تغییراتی اعمال کنید که فقط کاربر بتواند ارقام ۰ تا ۹ (۱ امتیاز) و علامت اعشاری (۲ امتیاز) را وارد کند (دقت کنید که هر عدد حداکثر یک علامت اعشاری دارد. برنامه شما باید این موضوع را کنترل کند).

(۲) با فشردن دکمه Ctrl به همراه دکمه + اندازه فرم حداکثر شود، با فشردن Ctrl به همراه دکمه - اندازه فرم کمینه شود و با فشردن دکمه Ctrl+0 اندازه فرم به حالت اولیه برگردد.

نکته مهم: برنامه باید به کیلدهای + یا - کیلدهای NumPad و یا + یا - بالای صفحه کلید عکس العمل نشان دهد و فرقی بین فشردن هر کدام از این کیلدها با کنترل نباشد.

(۳) تغییراتی در برنامه اعمال کنید که کاربر مجاز باشد فقط با استفاده از ارقام ۰،۱،۲،۳،۴،۵،۶،۷،۸،۹ برای ورود عدد در کادر متن txtA و txtB استفاده کند (۱ امتیاز) و عدد وارد شده در کادر متن txtA صرفاً شامل ارقام فرد (۱ امتیاز) و عدد وارد شده در کادر متن txtB یک عدد زوج باشد (۱ امتیاز).

ارزشیابی نهایی هنرجو

| سؤال | ۱ | ۲ | ۳ | جمع کل نمرات هنرجو |
|--|---|---|---|-------------------------|
| حداقل نمره قبولی از ۳ | ۲ | ۱ | ۱ | |
| **نمره هنرجو از ۳ | | | | **نمره نهایی هنرجو از ۳ |
| *چنانچه هنرجو نتواند حداقل نمره هر سؤال را کسب کند، نمره وی در آن سؤال برابر ۱ منظور شود. | | | | |
| **اگر مجموع نمرات هنرجو کمتر از ۴ باشد، پایین‌تر از حد انتظار (نمره ۱ از ۳)، اگر همه نمرات هنرجو در همه سؤال‌ها بالاتر از حداقل نمره قبولی باشد، فراتر از انتظار (نمره ۳ از ۳) و در غیر اینصورت در حد انتظار (نمره ۲ از ۳) ارزشیابی شود. | | | | |



پودمان پنجم

واحد یادگیری ۹

مدیریت پایگاه داده

واحد یادگیری ۹

شایستگی مدیریت پایگاه داده

مقدمات تدریسی

الف) مفاهیم کلیدی

| مفاهیم کلیدی | | | |
|----------------------------------|--------------------------------------|---------------------|---------------------------|
| خواندن از پرونده | نوشتن در پرونده | ایجاد پرونده | پرونده متنی |
| کلاس Path | کلاس Directory | کلاس File | فضای نام System.IO |
| کنترل DataGridView | ارتباط با پایگاه داده از طریق wizard | طراحی پایگاه داده | مدیریت استثنا |
| کامپوننت BindingSource | کامپوننت TableAdaptor | کامپوننت DataSet | مقیدسازی |
| مدیریت خطا در برنامه پایگاه داده | متدهای TableAdaptor | پرونده با قالب xsd | کنترل BindingNavigator |

ب) تجهیزات لازم

الزامات نرم افزاری:

- نرم افزار Visual Studio Express 2012
- تجهیزات سخت افزاری:
 - پردازنده حداقل ۱/۶ گیگاهرتز
 - حافظه RAM حداقل ۱ گیگابایت
 - حداقل فضای موجود در دیسک سخت ۴ گیگابایت
 - کارت ویدئویی متناسب با DirectX9 و قابل اجرا در رزولوشن 1024×768 و بالاتر

ج) بودجه بندی

| جلسه | واحد یادگیری | کارگاه (موضوع) | شماره صفحات | اهداف توانمندسازی | فعالیت‌های تکمیلی |
|------|--------------|---------------------------------|-------------|---|---|
| ۲۹ | ۹ | آشنایی با پرونده و کارگاه ۱ و ۲ | ۲۱۲ تا ۲۰۷ | - آشنایی با مفهوم پرونده و انواع آن - آشنایی با متدهای کلاس File برای نوشتن در پرونده - تشخیص متد مورد نیاز برای نوشتن در پرونده متنی | نوشتن در پرونده به فرمت xml و inf |
| ۳۰ | ۹ | کارگاه ۳ و ۴ | ۲۱۴ تا ۲۱۳ | - آشنایی با متدهای کلاس File برای خواندن از پرونده متنی - تشخیص وجود پرونده در مسیر مورد نظر | توسعه برنامه Notepad و استفاده از کنترل RichTextBox |
| ۳۱ | ۹ | کارگاه ۵ و ۶ | ۲۲۰ تا ۲۱۵ | - آشنایی با استثناها در برنامه - مدیریت استثنا در برنامه - آشنایی با متدهای کلاس Directory و کلاس Path برای مشاهده فهرست پرونده‌ها و پوشه‌ها | تولید استثناهای خاص با استفاده از کلمه کلیدی throw توسعه کارگاه ۶ نمایش فهرست درایوهای سیستم و افزودن قابلیت باز و بسته کردن درایو CD یا DVD به برنامه |
| ۳۲ | ۹ | طراحی پایگاه‌داده | ۲۲۳ تا ۲۲۱ | طراحی پایگاه‌داده مورد نیاز | از هنرجویان بخواهید به صورت گروهی در مورد تجهیزات هنرستان با سرپرست کارگاه‌ها مصاحبه کنند و پایگاه‌داده مناسب را طراحی کنند. |
| ۳۳ | ۹ | کارگاه ۷ و ۸ و ۹ و ۱۰ و ۱۱ | ۲۳۱ تا ۲۲۴ | - آشنایی با ابزارهای ارتباط با پایگاه‌داده - آشنایی با ویژگی‌های کنترل DataGridView - آشنایی با اتصال کادرمتن و DataGridView به منبع داده - آشنایی با متدهای کلاس BindingSource برای حرکت بین رکوردهای جدول - دسترسی به خانه‌های DataGridView | نحوه ایجاد منبع داده (Data Source) در VS2012 و اضافه کردن به فرم - بررسی متدهای دیگر کنترل BindingSource و کنترل DataGridView |
| ۳۴ | ۹ | کارگاه ۱۲ و ۱۳ و ۱۴ و ۱۵ و ۱۶ | ۲۴۱ تا ۲۳۳ | - جست‌وجوی دقیق و مشابه واژه در جدول - جست‌وجوی ترکیبی در جدول - آشنایی با ایجاد متدهای درج و حذف و ویرایش رکورد و فراخوانی آن در برنامه | - بررسی عملگر Like در زبان SQL و استفاده در جدول مشتری - بررسی ایمن بودن اعتبارسنجی کاربر به وسیله هنرجویان - به وسیله هنرجویان بررسی شود که اگر جدول جدیدی در پایگاه‌داده فروشگاه ایجاد شود چگونه متدهای درج و حذف و ویرایش برای آن ایجاد شود. |
| ۳۵ | ۹ | کارگاه ۱۷ و ۱۸ و ۱۹ | ۲۴۷ تا ۲۴۱ | - آشنایی با روش‌های درج و حذف سطر از کنترل DataGridView - کنترل خطا در برنامه پایگاه‌داده - آشنایی با نحوه درج همزمان چندین رکورد در یک پایگاه‌داده ارتباطی | امکانات درج و ویرایش در کنترل DataGridView قرار گیرد. پیام خطای مناسب هر خطا در برنامه نمایش داده شود. از هنرجویان بخواهید بررسی کنند چگونه می‌توان یک برگه خرید را در پایگاه‌داده فروشگاه حذف کرد. |

طرح درس روزانه (هفتگی) پیشنهادی

| پایه: یازدهم | | درس: توسعه برنامه‌سازی و پایگاه‌داده | |
|--|---|--|-------------|
| پیام جلسه (هدف کلی): ارتباط با پایگاه‌داده از طریق Wizard و سفارشی کردن DataGridView | | | |
| اهداف یادگیری | فعالیت‌ها | زمان | مدت |
| طبقة هدف: حیطه عاطفی / شناختی / روانی - حرکتی | کار هنر آموز | کار هنرجو | مدت (دقیقه) |
| ارزشیابی رفتار ورودی | معطوف کردن توجه هنرجویان به ابزارهایی که در برنامه ویندوزی برای ارتباط با پایگاه‌داده نیاز هست مثل کنترل DataGridView | از هنرجو بخواهید که کنترل DataGridView را در جعبه ابزار پیدا و مشخصات آن را بررسی کند. | ۱۰ |
| ایجاد انگیزه | با استفاده از ویدئو پروژکتور یا اجرای چند پروژه نمونه نمایش‌های مختلف داده‌ها را در DataGridView و دیگر کنترل‌ها مقید نمایش دهد. | از هنرجو بخواهید که نمونه برنامه‌هایی در محیط اطراف خود را که داده‌ها را نمایش می‌دهد، نام ببرد. | ۲۰ |
| ارائه مفاهیم کلیدی (توضیح هنر آموز) | پس از ایجاد پایگاه‌داده نمونه (Dictionary) و جدول (computer) فیلم نمایش داده‌های جدول در DataGridView را نمایش دهد و یک نمونه ارتباط ایجاد کند. | هنرجو با دقت فیلم و توضیحات هنر آموز را ببیند. | ۳۰ |
| انجام فعالیت کارگاهی شماره ۱ (تمرین هنرجویان) | هنر آموز توضیح کامل در خصوص فعالیت کارگاهی ۱ به هنرجو بدهد و از هنرجو بخواهد به سوالات کارگاه ۱ پاسخ دهد. | هنرجویان به صورت گروهی اقدام به انجام فعالیت کارگاه ۱ کرده، به سوالات کارگاه ۱ پاسخ دهند. | ۶۰ |
| نظارت بر عملکرد هنرجویان و ارزیابی فعالیت‌ها | بررسی نقاط ضعف هنرجویان در درک مفاهیم کلیدی | رفتار هنرجو را در هنگام اجرای تکالیف زیر نظر داشته باشد و در صورت لزوم از هنرجویان مستعدتر برای آموزش مفاهیم به آن‌ها کمک بگیرد. | ۳۰ |
| انجام فعالیت کارگاهی شماره ۲ (تمرین هنرجویان) | هنرجو باید کنترل DataGridView را بشناسد و آن را سفارشی کند. | هنرجویان به صورت گروهی اقدام به انجام فعالیت کارگاه ۲ کرده، جدول ۴ را تکمیل کنند. | ۶۰ |
| نظارت بر عملکرد هنرجویان و ارزیابی فعالیت‌ها | بررسی نقاط ضعف هنرجویان در درک مفاهیم کلیدی | رفتار هنرجو را در هنگام اجرای تکالیف زیر نظر داشته باشد و در صورت لزوم از هنرجویان مستعدتر برای آموزش مفاهیم به آن‌ها کمک بگیرد. | ۳۰ |
| دریافت بازخود از تدریس | هنرجو بتواند فعالیت منزل صفحه ۲۵۷ کتاب را انجام دهد. | تقسیم‌بندی وظایف کار فعالیت در منزل بین اعضای گروه و تحویل در زمان‌بندی تعیین‌شده | ۶۰ |
| ابزارهای موردنیاز | ویدئو پروژکتور، رایانه، تخته آموزشی، دفتر یادداشت | | |

د) ورود به بحث

شیوه پیشنهادی برای ارائه‌ی این پودمان روش تدریس پروژه‌محور است. در این روش هنرآموز، هنرجویان را به گروه‌هایی تقسیم کرده، پس از مشخص نمودن هدف درس و بیان پایه‌های تدریس، آنها با فعالیت گروهی، کارکرد گزینه‌های مختلف را کشف می‌کنند. می‌توانید هر قسمت از درس را به یک گروه محول کرده و پس از پایان زمان تعیین‌شده، یکی از اعضای گروه مفاهیم کلیدی را که فراگرفته‌اند، به دیگران توضیح دهد. در انتها بهتر است هنرآموز جمع‌بندی و تکمیل مبحث را انجام دهد.

این پودمان شامل دو بخش کار با پرونده‌ها و پوشه‌ها و ارتباط با پایگاه داده است. در بخش اول این پودمان هنرجویان ابتدا با مقدمات ذخیره پایدار داده‌های متنی در حافظه جانبی و بازیابی داده‌ها از حافظه جانبی آشنا می‌شوند. چگونگی ایجاد پرونده متنی، نوشتن داده‌ها در پرونده متنی و خواندن داده‌ها از پرونده متنی و ایجاد فهرست از محتویات درایوها، پوشه‌ها و پرونده‌های موجود در یک مسیر مشخص، در قالب کارگاه‌های مختلف آموزش داده شده و تمرین می‌شود.

کلاس‌ها و متدهایی که در بخش اول این پودمان معرفی شده و به کار گرفته می‌شوند در نمودار زیر آمده است.



سوالات تعیین سطح

- اگر در زمان کار با یک نرم افزار مانند Visual Studio کار خود را ذخیره نکرده باشیم و سیستم را خاموش کنیم چه اتفاقی برای اطلاعات ما رخ می دهد؟
- چگونه پس از خاموش شدن سیستم، اطلاعات حفظ می شوند؟
- نوشته ها چگونه در سیستم ذخیره می شوند؟
- تصویر، صوت و ویدیو چگونه در سیستم ذخیره می شوند؟
- اگر یک پرونده تصویری را در برنامه Notepad باز کنیم، چه چیزی خواهیم دید؟

پیش نیازهای لازم برای انجام فعالیت های این بخش از پودمان

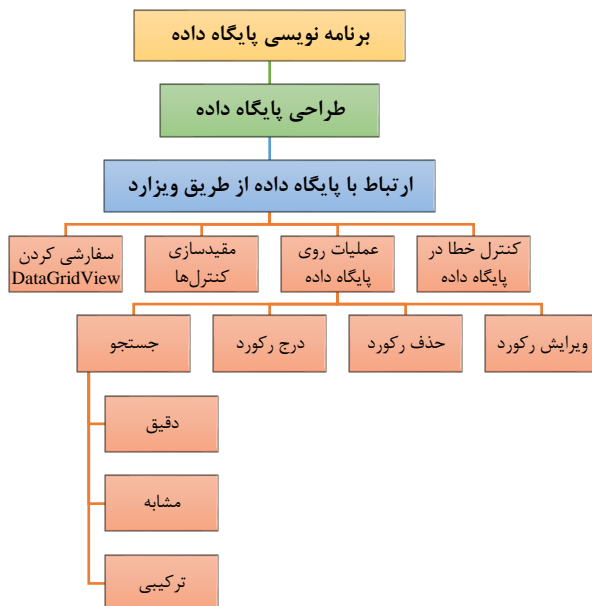
- آشنایی با نحوه طراحی واسط کاربری
- آشنایی با کنترل های پر کاربرد مانند کنترل برچسب، کادر متنی و ...
- آشنایی با مفهوم پوشه و پرونده و توانایی مدیریت پوشه ها و پرونده ها در سیستم عامل
- توانایی ایجاد و ذخیره پرونده متنی در ویرایشگر Notepad

در بخش دوم این پودمان نحوه برنامه نویسی برای ایجاد ارتباط با پایگاه داده Access در یک محیط عملیاتی آموزش داده می شود. در این بخش هنرجو ابتدا باید پایگاه داده مناسب برای محیط عملیاتی مورد نظر را طراحی و ایجاد کند؛ سپس از طریق Wisard یک ارتباط با پایگاه داده ایجاد کرده و پس از آن می تواند عملیات مختلف را روی پایگاه داده انجام دهد. در نهایت هنرجو باید بتواند عمل کنترل خطا را در یک برنامه متصل به پایگاه داده انجام دهد.

در پودمان اول کتاب درسی نحوه کار با برنامه Access برای انجام عملیات روی پایگاه داده به هنرجویان آموزش داده می شود. قبل از شروع به تدریس این بخش، ممکن است این سوال برای هنرجویان مطرح شود که با وجود برنامه Access دیگر چه نیازی به ایجاد یک برنامه در سی شارپ برای کار با پایگاه داده است.

- از هنرجویان بپرسید آیا همه کاربران نهایی می توانند پایگاه داده را در Access مدیریت کنند.
 - با نمایش برخی از برنامه های کاربردی مختلف ویندوزی و تلفن های هوشمند، از هنرجویان بپرسید که کدام برنامه ها کاربر پسندتر هستند، مدیریت برنامه در Access یا ایجاد یک برنامه جذاب ویندوزی برای مدیریت پایگاه داده؟
- مزیت های استفاده از برنامه پایگاه داده نسبت به پرونده های ترتیبی و دودویی در کلاس بیان شود.
- با مراجعه به کارگاه دفترچه تلفن در بخش اول این پودمان، تفاوت برنامه مبتنی بر پرونده ترتیبی و برنامه مبتنی بر پایگاه داده را نشان دهید.

برای آموزش این بخش، هنرجو باید با مفاهیم اولیه پایگاه داده نظیر طراحی پایگاه داده و عملیات روی پایگاه داده و زبان پرس و جوی SQL آشنایی اولیه داشته باشد. نمودار زیر نقشه مفهومی بخش دوم این پودمان را نشان می‌دهد و شما می‌توانید با داشتن یک دورنما از مطالب و محتوای آموزشی، تدریس خود را به شکل بهتری مدیریت کنید.



تدریس

پرونده و انواع آن

متن ساده (Plain text)

در علوم رایانه، متن ساده (Plain text) به داده‌هایی مانند محتویات پرونده گفته می‌شود که فقط از کاراکترهای قابل خوانده شدن تشکیل شده‌اند و هیچ نمایش گرافیکی مربوط به کاراکترها و یا اشیای دیگر مانند تصاویر، قلم و نظایر آن را شامل نمی‌شود. متن ساده همچنین می‌تواند شامل تعداد محدودی کاراکترهای کنترلی مثل کاراکتر انتهای خط یا کاراکتر جدول بندی (tab) باشد که چیدمان ساده متن را کنترل می‌کنند. متن ساده با متن قالب بندی شده که اطلاعات مربوط به سبک و قالب بندی را شامل می‌شود و همچنین با پرونده‌های دودویی که رکوردهای آنها باید به صورت فیلدهای دودویی متفاوتی تفسیر شوند (مثلاً اعداد صحیح، اعداد حقیقی، رشته متنی،

تصاویر و...) تفاوت دارد. به طور سنتی در کدگذاری متن ساده از استاندارد اسکی (ASCII) با محدودیت کد ۷ یا ۸ بیت استفاده می‌شود. هرچند کدگذاری یونیکد مانند UTF-8 یا UTF-16 در حال جایگزینی با کدهای اسکی است.

روشی که برای تمیز دادن پرونده‌های متنی ساده از پرونده‌های دودویی در اینجا پیشنهاد شده است، استفاده از یک ویرایشگر متنی ساده مانند Notepad است. البته باید توجه داشت که این روش یک ملاک و معیار قطعی برای تشخیص پرونده متنی از پرونده دودویی نیست. به عنوان مثال پرونده‌های از نوع rtf یک قالب ترکیبی از متن و محتویات دودویی شامل تصاویر هستند. در ضمن باید این نکته را هم در نظر داشت که یک واژه‌پرداز مانند word محتوای پرونده‌های متنی ساده را به درستی نشان می‌دهد اما اگر پرونده‌ای در واژه‌پرداز word به درستی نشان داده شد، این پرونده لزوماً یک پرونده متنی ساده نیست. به عنوان مثال پرونده‌های از نوع pdf که پرونده دودویی محسوب می‌شوند توسط واژه‌پرداز word نسخه ۲۰۱۳ به بعد باز می‌شوند.

پاسخ به فعالیت‌ها

در جدول زیر با کمک هم گروهی خود مشخص کنید، کدام پرونده‌ها از نوع دودویی و کدام پرونده‌ها از نوع متنی هستند. سعی کنید این نوع از پرونده‌ها را با برنامه Notepad باز کنید. چه تفاوتی مشاهده می‌کنید؟

| MP3 Audio File | C# source code | Microsoft Word Document | Adobe pdf Document | html |
|-------------------|-------------------|----------------------------|-----------------------|------|
| دودویی | متنی | دودویی | دودویی | متنی |

فعالیت گروهی

ص ۲۰۷

از هنرچوها بخواهید که ابتدا نوع پرونده را از نظر محتوا حدس بزنند؛ سپس در یک پوشه انواع پرونده‌های خواسته شده را گردآوری کرده، سعی کنند با برنامه Notepad آن‌ها را باز کنند. علاوه بر این می‌توانید از هنرچوها بخواهید که انواع دیگری از پرونده‌ها مانند پرونده‌های اجرایی (exe,dll)، ویدیویی (avi,mpg,mp4)، تصویری (jpg,tiff,gif) و پرونده‌های متنی با پسوند های log, inf, bat, xml را روی سیستم خود جست‌وجو کرده، آن‌ها را به این ترتیب امتحان کنند.

کارگاه ۱ – ایجاد پرونده و نوشتن در آن

در این کارگاه برای نوشتن داده رشته‌ای در پرونده متنی دو متد WriteAllText و AppendAllText معرفی شده، تفاوت آنها بررسی می‌شود. در زمان طراحی فرم دقت شود که فرم طراحی شده هنرجویان با محتوای فرم تناسب داشته باشد و کنترل‌ها به صورت منظم و مطابق شکل ارائه شده چیده شوند. از دیگر موارد مطرح شده در

این کارگاه استفاده از دنباله "\r\n" به عنوان نویسه خط جدید (New Line) و "\t" برای ایجاد یک Tab فاصله در پرونده متنی است. در این کارگاه با زدن دکمه ثبت، اطلاعات فردی شامل نام، نام خانوادگی، نام پدر و کدملی در پرونده متنی ثبت می‌شود.

```
string data = String.Format("{0} \t {1} \t {2} \t {3} \r\n",  
    txtFirstName.Text,  
    txtLastName.Text,  
    txtFatherName.Text,  
    txtNationalCode.Text);  
File.AppendAllText("Info.txt", data);
```

برای توسعه این کارگاه می‌توانید از هنرجویان بخواهید برای هر رکورد از اطلاعات فردی و نیز برای هر کدام از فیلدها یک برچسب (tag) به صورت زیر ایجاد کنند.

```
[Person]
```

```
Name=علی
```

```
LastName=احمدی
```

```
FatherName=رضا
```

```
NID=۱۲۵۴۷۸۹۶۵۴
```

برای این منظور یک متغیر رشته‌ای با نام newRecord برای ذخیره رکورد جدید تعریف کرده و آن را مطابق قالب مشخص شده، با داده‌های ورودی مقداردهی کنید.

```
string newRecord = String.Format("[Person] \r\n Name={0} \r\n \r\n \r\n LastName={1} \r\n \r\n \r\n FatherName={2} \r\n \r\n \r\n NID={3} \r\n \r\n",  
    txtFirstName.Text,  
    txtLastName.Text,  
    txtFatherName.Text,  
    txtNationalCode.Text);
```

```
File.AppendAllText("ContactList.ini", newRecord);
```

سیستم عامل ویندوز از روشی مشابه برای نگهداری برخی داده‌های مرتبط با پیکربندی خود در قالب پرونده‌هایی با پسوند ini یا inf استفاده می‌کند. به عنوان مثال پرونده‌های win.ini و system.ini را در پوشه نصب ویندوز ببینید.

کنجکاوی: اگر بخواهیم اطلاعات در یک مسیر مشخص ذخیره شود، متدهای نوشتن در پرونده را چگونه فراخوانی می‌کنید؟

باید آدرس پرونده را به عنوان ورودی به این متدها داد. آدرس ورودی می‌تواند مطلق یا نسبی باشد.

در روش آدرس‌دهی مطلق، آدرس از ریشه درایو داده می‌شود. به فرض اگر شما در هنرستان هستید و می‌خواهید به دوست خود آدرس یک سینما را بدهید، می‌توانید

آدرس سینما را از میدان مرکزی شهر مشخص کنید. دستور زیر نحوه آدرس‌دهی مطلق را نشان می‌دهد:

```
File.WriteAllText(@"C:\Users\Admin\Desktop>ContactList.txt", newContact + "\r\n");
```

در روش آدرس‌دهی نسبی، آدرس از مکان پوشه کاری جاری داده می‌شود. به فرض اگر محل فعلی شما را داخل هنرستان فرض کنیم شما می‌توانید آدرس یک مجتمع ورزشی را که چند کوچه یا خیابان با هنرستان فاصله دارد از محل هنرستان مشخص کنید. دستور زیر یک پرونده را در پوشه data که در مسیر پوشه کاری جاری قرار دارد ایجاد می‌کند.

```
File.WriteAllText("data\\ContactList.txt", newContact + "\r\n");
```

برای گرفتن آدرس پوشه کاری جاری می‌توان از متد `GetCurrentDirectory` از کلاس `Directory` استفاده کرد. همچنین در کلاس `Path` متدهایی برای کار با آدرس پرونده‌ها و پوشه‌ها وجود دارد که در ادامه این پودمان هنرجویان با برخی از متدهای آن‌ها آشنا می‌شوند.

کارگاه ۲ – پروژه Notepad

در این کارگاه با استفاده از ابزارهای منوی استاندارد و کنترل `saveFileDialog` نحوه ایجاد برنامه‌ای مشابه برنامه `Notepad` ویندوز برای ذخیره پرونده آموزش داده می‌شود. چگونگی برنامه‌نویسی برای گزینه‌های `save`، `save as` و `new` از اهداف اصلی این کارگاه است. این گزینه‌ها در اغلب برنامه‌های ویندوزی کاربرد دارند. رعایت جزئیات پیاده‌سازی نظیر استفاده از فیلتر مناسب برای پنجره محاوره‌ای `save` و تطبیق منوی برنامه با منوی برنامه `Notepad` از نکاتی است که در این کارگاه باید مورد توجه قرار گیرد.

مرحله ۴: ویژگی فیلتر کادر محاوره‌ای `saveFileDialog` برنامه `Notepad` به شکل زیر است:

```
"Text Documents|*.txt|All Files|*.*"
```

مرحله ۶: مسیر ذخیره پرونده در یک متغیر رشته‌ای به نام `documentFilePath` قرار می‌گیرد. در ابتدای اجرای برنامه، مقدار این رشته تهی است. تهی بودن این رشته نشان دهنده این است که هنوز پرونده ذخیره نشده است. با هر بار زدن گزینه `Save As` کادر محاوره‌ای `saveFileDialog` نشان داده‌شده و رشته مسیر پرونده از نو مقداردهی می‌شود.

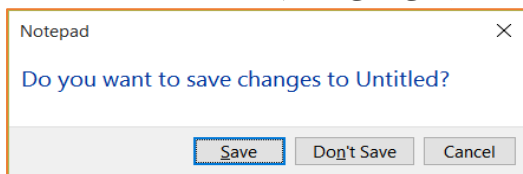
با زدن گزینه save در صورتی که رشته مسیر تهی باشد مانند روال گزینه save as عمل می‌شود و رشته مسیر مقدار می‌گیرد. در دفعات بعدی استفاده از گزینه save، پرونده متنی در مسیر مشخص شده با رشته مسیر documentFilePath ذخیره می‌شود. کد رخدادهای کلیک گزینه save به صورت زیر خواهد بود:

```
if (documentFilePath == "")// save as عمل انجام
{
    if (saveFileDialog.ShowDialog() == DialogResult.OK)
    {
        File.WriteAllText(saveFileDialog.FileName,
            mainDocument.Text);
        documentFilePath = saveFileDialog.FileName;
    }
}
else// save عمل انجام
{
    File.WriteAllText(documentFilePath,
        mainDocument.Text);
}
```

گزینه new محتوای کادر متنی سند را پاک کرده و رشته مسیر documentFilePath را نیز با رشته تهی "" مقداردهی می‌کند. کد رخداد کلیک گزینه new به صورت زیر خواهد بود:

```
mainDocument.Clear();
documentFilePath = "";
```

مرحله ۷: در برنامه notepad ویندوز اگر آخرین تغییرات متنی ذخیره نشده باشد، هنگام خروج از برنامه پیامی مبنی بر عدم ذخیره تغییرات به کاربر نشان داده می‌شود.



در این حالت به کاربر فرصت داده می‌شود که قبل از خروج، تغییرات انجام‌شده را ذخیره کند. برای افزودن این قابلیت به پروژه notepad مراحل زیر را دنبال کنید:

(۱) در پروژه notepad یک متغیر از نوع bool با نام saveStatusFlag و با مقدار اولیه true به کلاس فرم اصلی اضافه کنید. وظیفه این متغیر یا پرچم، نشان دادن وضعیت ذخیره شدن محتوای کادر متنی است.

```
public partial class MainForm : Form
{
    bool saveStatus = true;
}
```

بود یعنی محتوا قبلاً ذخیره شده است و true برابر با saveStatusFlag اگر مقدار برنامه می‌تواند بسته شود. در غیر اینصورت باید از بسته شدن برنامه جلوگیری شود و منتظر تأیید کاربر برای خروج بمانیم.

هر بار که نوشته کادر متنی تغییر می‌کند مقدار متغیر saveStatusFlag باید false شود. بنابراین در رخداد TextChanged برای کادر متنی، saveStatusFlag را false می‌کنیم.

```
private void mainDocument_TextChanged(object sender, EventArgs e)
{
    saveStatus = false;
}
```

(۲) در متدهای save و save as، پس از نوشتن موفقیت‌آمیز داده‌ها در پرونده، مقدار saveStatusFlag را دوباره true می‌کنیم.

```
saveStatus = true;
```

(۳) در رخداد FormClosing از فرم اصلی وضعیت ذخیره شدن متن را بررسی می‌کنیم. برای جلوگیری از بسته شدن فرم، در رخداد FormClosing مقدار e.Cancel را true کنید.

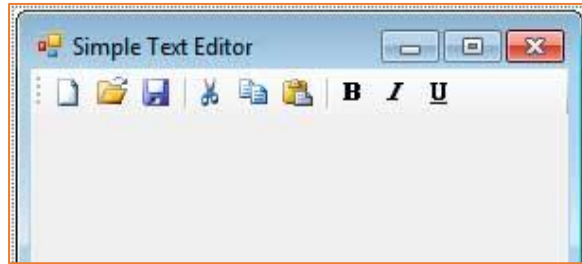
```
private void mainForm_FormClosing(object sender, FormClosingEventArgs e)
{
    if (!saveStatus)
    {
        DialogResult result = MessageBox.Show(this,
            "Do you want to save changes?",
            "Notepad", MessageBoxButtons.YesNoCancel);
        switch (result)
        {
            case DialogResult.Yes:
                // کد عمل save در اینجا قرار می‌گیرد//
                e.Cancel = !saveStatus;
                break;
            case DialogResult.Cancel:
                e.Cancel = true;
                break;
        }
    }
}
```

برای توسعه بیشتر این کارگاه می‌توانید از هنرجویان بخواهید که به‌جای استفاده از کنترل کادرممتنی ساده از یک کنترل کادرممتنی غنی RichTextBox استفاده کنند. کنترل RichTextBox برای نمایش دادن متن‌های قالب بندی شده در قالب Rich

به‌سادگی می‌توان از آن برای ذخیره پرونده استفاده کرد. به عنوان مثال در رخداد کلیک گزینه save as می‌توان به شکل زیر پرونده را با پسوند rtf و یا doc ذخیره کرد.

```
if (saveFileDialog1.ShowDialog() == DialogResult.OK)
{
    // پرونده مسیر روزرسانی به
    documentFilePath = saveFileDialog1.FileName;
    // پرونده در متن نوشتن
    richTextBox1.SaveFile(documentFilePath);
}
```

می‌توان با اضافه کردن یک نوار ابزار ToolStrip به برنامه، امکانات قالب‌بندی متن نظیر تغییر رنگ، تغییر قلم و ضخیم یا مورب کردن متن انتخاب‌شده را به برنامه افزود.



پاسخ به فعالیت‌ها

فعالیت کارگاهی

ص ۲۱۱

برنامه‌ای بنویسید که قابلیت ذخیره مخاطبان در دفترچه تلفن را داشته باشد.

| | | | |
|-----|--------|----------|--------|
| علی | علوی | 35429090 | نوربان |
| بوس | محمدی | 50901020 | مسهد |
| رضا | فرهادی | 43509070 | تبریز |

نام: رضا
نام خانوادگی: فرهادی
تلفن: 43509070
آدرس: تبریز

افزافه به فهرست <<<

شکل ۳- فرم دفترچه تلفن

پاسخ: هدف از این فعالیت کارگاهی تمرین کار با متد `AppendAllText` برای افزودن متن به انتهای پرونده است. با زدن دکمه "افزافه به فهرست" محتویات کادرهای متنی همزمان به انتهای کادر فهرست و پرونده متنی اضافه می‌شوند.

```
string newContact =  
String.Format("{0}\t{1}\t{2}\t{3}\r\n",  
txtFirstName.Text,  
txtLastName.Text,  
txtPhone.Text,  
txtAddress.Text);  
File.AppendAllText("ContactList.txt", newContact);  
lstContact.Items.Add(newContact);
```

از یک کاراکتر خط عمود '|' می‌توان برای جدا کردن فیلدها از یکدیگر استفاده کرد.

برای توسعه این فعالیت کارگاهی می‌توانید از هنجارها بخواهید که رکورد داده‌های افراد را در پرونده `ContactList.xml` با قالب `xml` به شکل زیر ذخیره کنند.

```
<person>  
  <name>رضا</name>  
  <surname>فرهادی</surname>  
  <tel>۰۹۸۷۶۵۴۳۲۱</tel>  
  <address>تبریز</address>  
</person>
```

برای این منظور یک متغیر رشته‌ای با نام `newRecord` برای ذخیره رکورد جدید تعریف کرده، آن را مطابق قالب `xml` با داده‌های ورودی مقداردهی کنید. از رشته‌های `"\t"` و `"\r\n"` برای قالب‌بندی متن استفاده کنید.

```

string newRecord =
String.Format("<person>\r\n\t<name>{0}</name>\r\n\t<surname>{1}</surname>\r\n\t<tel>{2}</tel>\r\n\t<address>{3}</address>\r\n</person>",
txtFirstName.Text,
txtLastName.Text,
txtPhone.Text,
txtAddress.Text);
File.AppendAllText("ContactList.xml", newRecord);

```

XML مخفف زبان نشانه‌گذاری قابل گسترش (eXtensible Markup Language) است. XML کاربردهای فراوانی دارد. با XML می‌توان داده‌ها را بین دو سیستم ناهمسان مبادله کرد، فایل‌های متن ساده را برای به اشتراک گذاشتن داده‌ها به کار برد، فایل‌های متن ساده را برای ذخیره داده‌ها به کار گرفت و داده‌ها را در دسترس کاربران بیشتری قرار داد. تشخیص رکوردها و فیلدها و ذخیره و بازیابی رکوردها در قالب پرونده متنی، مقدمه‌ای برای ورود به بحث ارتباط با پایگاه داده است.

فعالیت منزل
ص ۲۱۱

بازی سرعت تایپ یا اعصاب‌سنج در فصل ۴ را توسعه دهید تا پس از پایان بازی نام کاربر و امتیازش را در یک پرونده متنی ذخیره کند. پاسخ: هنرجویان می‌توانند با استفاده از متد AppendAllText نام کاربر و امتیازش را در یک پرونده متنی به صورت خط به خط اضافه کنند. توصیه می‌شود برای ذخیره داده‌ها از قالب xml استفاده شود.

```

string newRecord =
String.Format("<player>\r\n\t<name>{0}</name>\r\n\t<score>{1}</score>\r\n</player>".
userName.
score);
File.AppendAllText("score.xml", newRecord);

```

کارگاه ۳ – خواندن پرونده متنی با متد ReadAllText

در تکمیل پروژه Notepad هنرجویان در این کارگاه عملکرد گزینه Open را فعال می‌کنند.

مرحله ۳: عملکرد رویداد کلیک گزینه Open چیست؟

در رخداد کلیک گزینه Open از منوی برنامه، ابتدا با فراخوانی متد ShowDialog از کنترل OpenFileDialog پنجره باز کردن پرونده نمایش داده می‌شود. در صورتی که کاربر پرونده‌ای را برای باز کردن انتخاب کند و دکمه Open را بزند، آدرس پرونده در متغیر سراسری documentFilePath ذخیره شده، محتوای پرونده با متد ReadAllText خوانده و در کادر متنی نمایش داده می‌شود.

مرحله ۴: کد مرحله ۳ را تغییر دهید تا قبل از باز کردن پرونده جدید، از کاربر برای ذخیره پرونده باز شده قبلی، سؤال کند.

رخداد کلیک گزینه Open را به صورت زیر تغییر می‌دهیم:

```
private void openToolStripMenuItem_Click(object sender,
EventArgs e)
{
    if (!saveStatus)
    {
        DialogResult result = MessageBox.Show(this,
            "Do you want to save changes?", "Notepad",
            MessageBoxButtons.YesNoCancel);
        if (result == DialogResult.Yes)
        {
            //کد عمل save در اینجا قرار می‌گیرد//
        }
        if (result == DialogResult.Cancel)
        {
            return;
        }
    }
    if (openFileDialog.ShowDialog() == DialogResult.OK)
    {
        documentFilePath = openFileDialog.FileName;
        mainDocument.Text =
        File.ReadAllText(documentFilePath);
        saveStatus = true;
    }
}
```

با تنظیم خاصیت راست به چپ فرم و ترجمه گزینه‌های منو به زبان فارسی، برنامه Notepad را فارسی کنید.

پاسخ: در این تمرین هنجاریان باید به معادل‌های متعارف زبان فارسی برای واژه‌های استفاده شده در منوی برنامه توجه کنند. برای این منظور می‌توان از واژه‌های مصوب فرهنگستان زبان و ادب پارسی استفاده کرد.

| معادل فارسی | واژه | معادل فارسی | واژه |
|-------------|-------|------------------------|---------|
| برش | Cut | سند نو، سند تازه، جدید | New |
| رونوشت | Copy | بازکردن سند | Open |
| چسباندن | Paste | حفظ سند، ذخیره سند | Save |
| قلم | Font | حفظ با نام، ذخیره در | Save as |

- برای گزینه‌های منوی Edit برنامه Notepad کدنویسی کنید.

با استفاده از متدهای SelectAll، Cut، Copy و Paste از کادر متنی، به ترتیب قابلیت‌گزینه‌های SelectAll، Cut، Copy و Paste در منوی Edit را می‌توان به برنامه اضافه کرد.

```
private void copyToolStripMenuItem_Click(object sender,
EventArgs e)
{
    mainDocument.Copy();
}

private void pasteToolStripMenuItem_Click(object sender,
EventArgs e)
{
    mainDocument.Paste();
}

private void cutToolStripMenuItem_Click(object sender,
EventArgs e)
{
    mainDocument.Cut();
}

private void selectAllToolStripMenuItem_Click(object
sender, EventArgs e)
{
    mainDocument.SelectAll();
}
```

کارگاه ۴ - خواندن پرونده متنی با متد ReadAllLines

در این کارگاه عملکرد متدهای ReadAllLines و ReadAllText در خواندن داده‌های پرونده با هم مقایسه می‌شود. همچنین برای اطمینان از وجود پرونده در مسیر مشخص شده متد File.Exists معرفی می‌شود.

پاسخ به فعالیت‌ها

مرحله ۲: - برنامه را اجرا کنید. چه اطلاعاتی را در فهرست مشاهده می‌کنید؟

اطلاعات تماس ذخیره شده در پرونده ContactList.txt

- رشته data حاوی چه مقادیری است؟

محتویات پرونده ContactList.txt

- عملکرد متد ReadAllLines چیست؟

خواندن خطوط پرونده متنی و ذخیره آن‌ها در یک آرایه از رشته‌ها.

- عملکرد متد AddRange چیست؟

تمام محتویات یک آرایه از رشته‌ها را خط به خط به کادر فهرست اضافه می‌کند.

مرحله ۳: برنامه را اجرا کنید. تفاوت متد ReadAllLines با متد ReadAllText در چیست؟

متد ReadAllLines تمام محتویات پرونده متنی را خط به خط خوانده، این خطوط را به ترتیب در آرایه‌ای از رشته‌ها قرار می‌دهد، اما متد ReadAllText تمام محتویات پرونده را در یک متغیر رشته‌ای برمی‌گرداند.

از متد ReadAllLines در جایی استفاده می‌شود که بخواهیم به خطوط پرونده متنی به طور مستقیم دسترسی پیدا کرده، پرونده را خط به خط ویرایش کنیم. به عنوان مثال اگر بخواهید خطوط پنجم و هفتم یک پرونده متنی را با هم جابه‌جا کنید، کافی است که متد ReadAllLines را فراخوانی کرده، در آرایه خروجی جای عنصر پنجم و هفتم را جابه‌جا کنید، سپس با فراخوانی متد WriteAllLines می‌توان محتویات این آرایه را در پرونده بازنویسی کرد.

```
string[] str = File.ReadAllLines(fileName);  
string temp = str[4];  
str[4] = str[6];  
str[6] = temp;  
File.WriteAllLines(fileName, str);
```

تکمیل کارگاه

ص ۲۱۴

مرحله ۴: پرونده ContactList.txt را از پوشه Debug برنامه حذف کنید. برنامه را اجرا کنید. چه خطایی رخ می‌دهد؟ چرا؟
 یک خطای استثنای مدیریت نشده (FileNotFoundException) رخ می‌دهد و نشان می‌دهد که در مسیر مشخص شده پرونده ContactList.txt وجود نداشته.
 مرحله ۵: نوع خروجی متد Exists چیست؟
 خروجی این متد از نوع bool است. اگر پرونده در مسیر path وجود داشته باشد مقدار true و در غیر اینصورت مقدار false برگردانده می‌شود.
`public static bool Exists(string path);`

کارگاه ۵ – مدیریت خطا با دستور try-catch

برای اداره کردن استثناها در C# از کلاس Exception استفاده می‌شود. این کلاس، کلاس پایه تمام استثناها است و استثناهای دیگر از این کلاس مشتق می‌شوند. کلاس Exception ویژگی‌ها و متدهای مختلفی دارد که می‌توان در راهنمای مایکروسافت به آدرس "docs.microsoft.com" آن‌ها را مشاهده کرد. یکی از پرکاربردترین ویژگی‌های این کلاس، ویژگی Message است. این ویژگی یک پیام متنی است و توضیحی در مورد استثنایی که رخ داده، در اختیار برنامه‌نویس قرار می‌دهد.

علاوه بر استثناهای تعریف شده در کتابخانه Net، برنامه‌نویس می‌تواند کلاس استثنای خاص خود را با مشتق کردن این کلاس از کلاس پایه Exception ایجاد کرده و در صورت لزوم با استفاده از کلمه کلیدی throw، شیء استثنای ایجادشده از این کلاس را به صورت صریح (explicit) در برنامه پرتاب (throw) کند.

در مثال زیر ابتدا یک کلاس به نام UserListNotFoundException که از کلاس Exception مشتق شده است، در برنامه تعریف می‌شود. سپس در جایی از برنامه مثلاً رخداد کلیک یک دکمه، یک شیء استثناء جدید از این کلاس ایجاد شده و با استفاده از کلمه کلیدی throw در برنامه پرتاب می‌شود. استثنای پرتاب شده در قسمت catch از ساختار try-catch به دام افتاده و پیام آن به کاربر اعلام می‌شود.

```
public class UserListNotFoundException : Exception
{
    public UserListNotFoundException(string message)
        : base(message)
    {
    }
}
```

نحوه ایجاد یک شی از کلاس استثنای تعریف شده توسط برنامه‌نویس و پرتاب این استثنا در برنامه:

```
try
{
    throw new UserListNotFoundException("لیست کاربران یافت نشد.");
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message, "خطا");
}
```

در حالتی که ممکن است در بخش try بیش از یک حالت استثنا ایجاد شود، برنامه‌نویس می‌تواند حالت‌های استثنایی مختلف را یک به یک و به صورت مجزا در بلوک‌های catch متفاوت مدیریت کند. ساختار کلی دستور try-catch برای مدیریت استثناهای چندگانه به صورت زیر است:

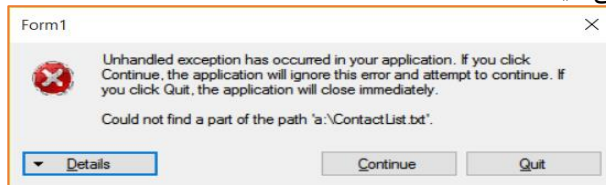
```
try
{
    // قسمتی از کد که ممکن است استثنا ایجاد کند
}
catch (ExceptionName1 e1)
{
    // کدهای مدیریت خطا برای استثنای e1
}
catch (ExceptionName2 e2)
{
    // کدهای مدیریت خطا برای استثنای e2
}
catch (ExceptionNameN eN)
{
    // کدهای مدیریت خطا برای استثنای eN
}
finally
{
    // دستوراتی که در نهایت اجرا می‌شوند
}
```

اگر استثنایی رخ دهد کنترل برنامه به قسمت catch مربوط به آن استثنا منتقل می‌شود. در هر صورت اگر استثنایی رخ دهد و یا رخ ندهد در انتها بلوک دستورات finally اجرا خواهد شد. به عنوان مثال اگر شما پرونده‌ای را برای خواندن یا نوشتن باز کرده‌اید، در هر صورت چه استثنایی رخ دهد چه رخ ندهد باید پرونده را در انتهای کار ببندید.

پاسخ به فعالیت‌ها

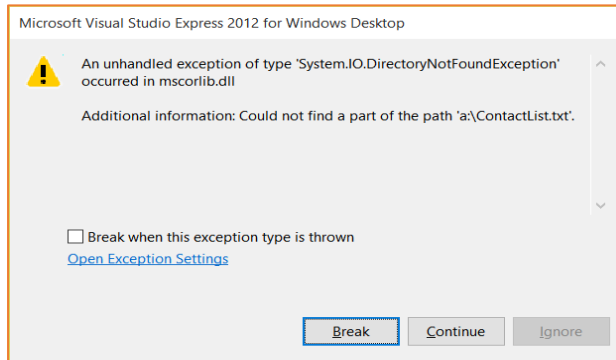
تکمیل کارگاه
ص ۲۱۵

مرحله ۲: برنامه را اجرا کنید. چه خطایی مشاهده می‌کنید؟ با کمک هنرآموز خود پیام نشان داده شده در پنجره خطا را به فارسی معنی کنید. پاسخ: اگر برنامه را با **Ctrl+F5** و بدون دیباگ کردن اجرا کنید، پنجره خطا نمایش داده می‌شود. این حالت مانند وقتی است که با دو بار کلیک روی پرونده exe برنامه را اجرا می‌کنید.



یک استثنای مدیریت نشده در برنامه رخ داده است. اگر بر روی دکمه **Continue** کلیک کنید، برنامه از این خطا چشم‌پوشی کرده، سعی خواهد کرد که اجرای خود را ادامه دهد. اگر بر روی **Quit** کلیک کنید، برنامه بلافاصله بسته خواهد شد. بخشی از مسیر "a:\ContactList.txt" یافت نشد.

اگر برنامه را با **F5** و در حالت دیباگ اجرا کنید، پنجره خطا به شکل زیر خواهد بود.



مرحله ۴: برنامه را اجرا کنید. اجرای برنامه در این حالت چه تفاوتی با قبل دارد؟ پاسخ: به جای پیام خطای قبلی، پیام زیر مشاهده می‌شود.



مشاهده فهرست پرونده‌ها و پوشه‌ها

بسیاری از اوقات کار با پرونده‌ها در یک برنامه مستلزم انجام عملیاتی نظیر ایجاد، حذف، ایجاد نسخه مشابه، جابه‌جایی و برخی کارهای دیگر روی پوشه‌ها و یا استخراج پسوند یا نام پرونده از یک مسیر است. در این بخش با معرفی کلاس‌های Directory و Path و آشنایی با برخی از متدهای پرکاربرد آن‌ها هنرجویان آماده می‌شوند که برنامه‌ای شبیه به مرورگر پرونده (FileExplorer) ویندوز بنویسند.

پاسخ به فعالیت‌ها

با استفاده از MSDN جدول متدهای کلاس Directory را تکمیل کنید.

| نام متد | نوع ورودی | نوع خروجی | عملکرد |
|------------------|------------|---------------|--|
| GetLogicalDrives | ندارد | آرایه رشته‌ای | نام همه درایوها را در یک آرایه برمی‌گرداند. |
| GetDirectories | رشته(مسیر) | آرایه رشته‌ای | نام و مسیر تمام زیرپوشه‌های مسیر ورودی را در یک آرایه برمی‌گرداند. |
| GetFiles | رشته(مسیر) | آرایه رشته‌ای | نام و مسیر همه پرونده‌های مسیر ورودی را در یک آرایه برمی‌گرداند. |
| Exists | رشته(مسیر) | bool | تشخیص می‌دهد که آیا مسیر ورودی به پوشه‌ای که روی دیسک وجود دارد اشاره می‌کند یا خیر. |

فعالیت کارگاهی
ص ۲۱۷

باید توجه شود که برخی متدها مانند متد GetDirectories و متد GetFiles دارای چند ریخت مختلف با تعداد ورودی‌ها و یا نوع خروجی متفاوت هستند. در زبان‌های برنامه‌نویسی شیء‌گرا مانند C# می‌توان دو و یا بیشتر از دو متد هم نام را با پارامترهای ورودی متفاوت در یک کلاس تعریف کرد. به این کار Overload کردن متدها گفته می‌شود. Overloading یکی از روش‌های پشتیبانی از چندریختی (polymorphism) در زبان‌های برنامه‌نویسی است.

از دیگر روش‌های پشتیبانی از چندریختی در زبان‌های برنامه‌نویسی شیء‌گرا، Override کردن متدها است. Overriding قابلیت است که در آن یک کلاس فرزند می‌تواند پیاده‌سازی خاص خود را از متدهای کلاس پدر داشته باشد.

با استفاده از راهنمای میکروسافت جدول متدهای کلاس Path را تکمیل کنید.

| نام متد | نوع ورودی | نوع خروجی | عملکرد |
|-------------|-------------------------|----------------|--|
| GetFileName | یک رشته (مسیر) | یک رشته | نام و پسوند پرونده مسیر ورودی را برمی گرداند. |
| Combine | دو رشته (مسیر) یا بیشتر | یک رشته (مسیر) | آرایه‌ای از رشته‌ها را با هم ترکیب کرده و از ترکیب آنها یک مسیر برمی گرداند. |

کارگاه ۶ – کاربرد کلاس Path و Directory

پاسخ به فعالیت‌ها

۴: برنامه را اجرا کنید و روی دکمه «فهرست پوشه‌ها» کلیک کنید. چه عناصری در فهرست مشاهده می‌کنید؟

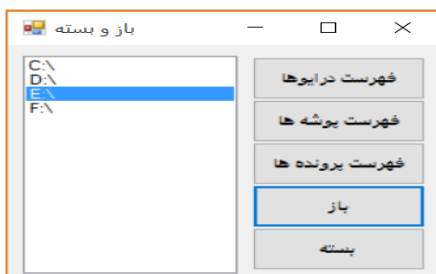
پاسخ: فهرست پوشه‌های موجود در ریشه درایو C

۵: کد مرحله ۴ را تغییر دهید تا مسیر `directory` را از کادرمتن دریافت کند.
`string directory = txtPath.txt;`

پاسخ:

۸: برنامه را اجرا کنید. کد مرحله ۴ و ۸ را با هم مقایسه کنید.

پاسخ: در مرحله ۴ مسیر کامل پوشه نمایش داده می‌شود اما در مرحله ۸ فقط نام پوشه نمایش داده می‌شود.



به عنوان توسعه این کارگاه می‌توانید دکمه‌ای را برای نمایش فهرست درایوهای سیستم و همچنین دکمه‌هایی را برای باز و بسته کردن درایو CD یا DVD به این برنامه اضافه کنید.

کد رخداد کلیک دکمه فهرست درایوها به شکل زیر است.

```
private void btnAddDrives_Click(object sender, EventArgs e)
{
    lstBoard.Items.Clear();
    string[] drives = Directory.GetLogicalDrives();
    lstBoard.Items.AddRange(drives);
}
```

برای افزودن قابلیت باز و بسته کردن درایو CD یا DVD به برنامه، فضای نام مناسب را اضافه کنید.

```
using System.Runtime.InteropServices;
```

در داخل کلاس فرم کد زیر را اضافه کنید. این کد تابع `mciSendString` را به عنوان یک متد خارجی به کلاس فرم اضافه می‌کند. تابع `mciSendString` یکی از API‌های ویندوز برای کار با تجهیزات MCI است.

```
[DllImport("winmm.dll", EntryPoint = "mciSendStringA",  
CharSet = CharSet.Ansi)]  
protected static extern int mciSendString(string  
lpstrCommand,  
StringBuilder lpstrReturnString,  
int uReturnLength,  
IntPtr hwndCallback);
```

در رخدادهای کلیک دکمه "باز" کد زیر را بنویسید. دستور `mciSendString` اول، درایو را مشخص می‌کند و دستور دوم فرمان باز یا بسته شدن را به درایو انتخاب شده ارسال می‌کند.

```
string driveLetter = listBox1.SelectedItem.ToString();  
StringBuilder returnString = new StringBuilder();  
mciSendString("open " + driveLetter + ": type CDAudio  
alias drive" + driveLetter,  
returnString, 0, IntPtr.Zero);  
mciSendString("set drive" + driveLetter + " door open",  
returnString, 0, IntPtr.Zero);
```

در رخدادهای کلیک دکمه "بسته" کد زیر را بنویسید.

```
string driveLetter = listBox1.SelectedItem.ToString();  
StringBuilder returnString = new StringBuilder();  
mciSendString("open " + driveLetter + ": type CDAudio  
alias drive" + driveLetter,  
returnString, 0, IntPtr.Zero);  
mciSendString("set drive" + driveLetter + " door closed",  
returnString, 0, IntPtr.Zero);
```

کاربرد پایگاه داده

یکی از مهمترین بخش‌های برنامه نویسی پایگاه داده، طراحی پایگاه داده مربوطه است که اگر طراحی مناسب صورت نگیرد، برنامه دچار مشکلات بزرگی خواهد شد. برای استخراج گزارش‌های کاربردی و جلوگیری از مشکلاتی نظیر وجود داده‌های تکراری و تداخل و ... باید پایگاه داده دقیق و ارتباطی ایجاد شود. جهت ایجاد پایگاه داده مناسب، پودمان اول کتاب برای هنرجو به اختصار یادآوری شود و پایگاه داده فروشگاه به صورت گام به گام به هنرجو آموزش داده شود و برای تقویت طراحی، فعالیت‌های این واحد انجام شود.

پاسخ به فعالیت‌ها

فعالیت کارگاهی
ص ۲۲۱

به کمک هم گروهی خود یک پایگاه داده برای ذخیره مخاطبان ایجاد کنید و به سوالات زیر پاسخ دهید.

■ چگونه پایگاه داده را در Access ایجاد می‌کنید؟

پاسخ: پس از اجرای برنامه Access 2016 و انتخاب گزینه Blank desktop database، تعیین مسیر مناسب برای ایجاد پایگاه داده، نام پایگاه داده را PhoneNotebook قرار دهید و دکمه Create را کلیک کنید.

■ چگونه در پایگاه داده، جدول مخاطبان را ایجاد می‌کنید؟

پاسخ: از سربرگ Create گزینه Table را انتخاب کنید. روی جدول ایجاد شده در سمت چپ راست کلیک کرده، گزینه Design View را انتخاب کنید. نام جدول را Contacts قرار دهید و فیلدهای مناسب را در جدول قرار دهید.

■ چه فیلدهایی را برای جدول مخاطبان تعریف می‌کنید؟ نوع داده این فیلدها چیست؟

پاسخ: در جدول زیر حداقل فیلدها برای جدول مخاطبان تعریف شده است:

| Field Name | Data Type | |
|----------------|------------|--------------------|
| CountryCode | Short Text | کد کشور |
| MobileNumber | Short Text | شماره همراه |
| ContactFName | Short Text | نام مخاطب |
| ContactLName | Short Text | نام خانوانگی مخاطب |
| Oragnization | Short Text | شرکت |
| Email | Short Text | ایمیل |
| ContactPicture | Short Text | آدرس تصویر مخاطب |

■ چگونه رکورد جدید در جدول مخاطبان وارد می‌کنید؟
 با دابل کلیک روی جدول می‌توان در صفحه گسترده (DataSheet View) باز شده، زیر هر ستون مقدار مناسب را وارد کرد و یک رکورد جدید با کامل کردن یک سطر صفحه گسترده ایجاد می‌شود.

■ چگونه یک رکورد را حذف می‌کنید؟
 با راست کلیک روی سطر دلخواه در صفحه گسترده و با انتخاب Delete Record می‌توان یک رکورد دلخواه را حذف کرد.

کنجکاوی
 ص ۲۲۱

- چند نرم‌افزار پیرامون خود را نام ببرید که از پایگاه داده برای ذخیره و بازیابی اطلاعات استفاده می‌کنند.
 پاسخ: نرم‌افزار دانا مدارس - نرم‌افزار حسابداری گردو

- نام بازی‌هایی را بنویسید که فکر می‌کنید از پایگاه داده برای ذخیره اطلاعات بازی استفاده می‌کنند.
 پاسخ: Soccer Stars و Pes2016

- چند نرم‌افزار فارسی تلفن هوشمند را نام ببرید که از پایگاه داده استفاده می‌کند.
 پاسخ: بازار - لغت نامه معین - اسنپ

- آیا امکانات موجود در نرم‌افزارهای مدیریت پایگاه داده نظیر Access برای تولید یک نرم‌افزار که کار ذخیره و بازیابی اطلاعات را انجام دهد، کافی نیست؟ چرا؟
 پاسخ: خیر. چون برای استفاده کامل از امکانات Access کاربر باید حرفه‌ای باشد؛ بنابراین برای مدیریت بهتر باید نرم‌افزار کاربردی طراحی کرد تا کاربر معمولی بتواند به سادگی از آن استفاده کند.

- یک بازی برخط (online) که مشخصات و سطح و امتیاز کاربران در آن ذخیره می‌شود را بررسی کنید. فکر می‌کنید پایگاه داده این بازی از چه جدول‌هایی تشکیل شده است؟
 پاسخ: جدول کاربران، جدول بازی‌ها، جدول ابزار و....

ایجاد پایگاه داده

پاسخ به فعالیت‌ها

جدول بالا را در Access پیاده‌سازی کنید و رکوردهای بالا را در جدول وارد کنید. اگر دو کارمند فرهنگی نام و نام خانوادگی یکسانی داشته باشند، آیا برای صورت حساب آن‌ها مشکلی پیش می‌آید؟ چرا؟ بله. مغایرت در داده‌ها پیش می‌آید. مشکل را چگونه حل می‌کنید؟
 پاسخ: با تعریف کلید اصلی برای جدول.

فعالیت کارگامی
 ص ۲۲۲

برگ خرید شامل چه داده‌هایی می‌شود؟

برای نمایش کالاها در برگ خرید نیاز به «جدول کالا» داریم، این جدول شامل چه فیلدهایی می‌شود؟

| Goods جدول کالا | | |
|-----------------|------------|-----------|
| Field Name | Data Type | |
| GoodsCode | Number | کد کالا |
| GoodsName | Short Text | نام کالا |
| GoodsUnitPrice | Number | قیمت واحد |
| GoodsStock | Number | موجودی |

نکته: در جدول بالا می‌تواند فیلدهای دیگری نیز تعریف شود. کد کالا می‌تواند از نوع AutoNumber هم تعریف شود.

برای نمایش اطلاعات مربوط به کارمند فرهنگی در برگ خرید نیاز به «جدول مشتری» داریم، این جدول شامل چه فیلدهایی می‌شود؟

| Customers جدول مشتری | | |
|----------------------|------------|--------------------|
| Field Name | Data Type | |
| CustomerCode | Number | کد مشتری |
| CustomerFirstName | Short Text | نام مشتری |
| CustomerLastName | Short Text | نام خانوادگی مشتری |
| CustomerMobile | Short Text | شماره همراه مشتری |

نکته: در جدول بالا می‌تواند فیلدهای دیگری نیز تعریف شود. کد مشتری می‌تواند از نوع AutoNumber هم تعریف شود.

برای ایجاد هر برگ خرید نیاز به «جدول برگه خرید» داریم که مشخص کند، مشتری در چه تاریخی چه کالاهایی خریده است. این جدول شامل چه فیلدهایی می‌شود؟

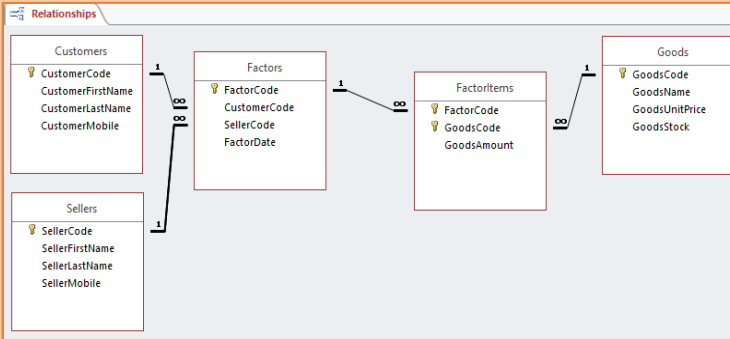
| Factors جدول برگه خرید | | |
|------------------------|-----------|-----------------|
| Field Name | Data Type | |
| FactorCode | Number | شماره برگه خرید |
| CustomerCode | Number | کد مشتری |
| FactorDate | Date/Time | تاریخ برگه خرید |

نکته: شماره برگه خرید می‌تواند از نوع AutoNumber هم تعریف شود.

مشتری در هر بار مراجعه ممکن است چندین قلم کالا خریداری کند؛ بنابراین نمی‌توان کالاهای خریداری شده را در «جدول برگه خرید» ذخیره کرد به همین دلیل، جدول دیگری به نام «جدول ثبت کالا در برگه خرید» ایجاد می‌کنیم. این جدول شامل چه فیلدهایی می‌شود؟

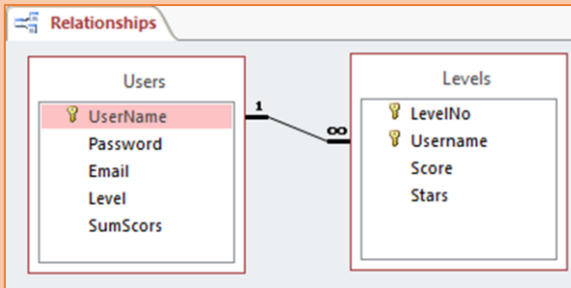
| FactorItems جدول ثبت کالا در برگه خرید | | |
|--|-----------|-----------------|
| Field Name | Data Type | |
| FactorCode | Number | شماره برگه خرید |
| GoodsCode | Number | کد کالا |
| GoodsAmount | Number | تعداد کالا |

فرض کنید در فروشگاه فرهنگیان بیش از یک فروشنده کار فروش کالا را انجام می‌دهند، اگر قرار باشد در برگ خرید نام فروشنده نیز قید شود، چه تغییری باید در ساختار پایگاه داده فروشگاه صورت گیرد؟



جدول Seller جدول فروشنده است.

- در یک بازی می‌توان با ایجاد یک نام کاربری وارد بازی شد. هر کاربر با کسب حداقل امتیاز به سطح بالاتر می‌رود و با توجه به امتیاز خود ستاره کسب می‌کند. پایگاه داده مناسب را برای این بازی طراحی کنید تا نشان دهد که هر کاربر در هر سطح چند امتیاز و چند ستاره دارد.



کارگاه ۷ - ارتباط با پایگاه داده از طریق Wizard

ساده‌ترین راه برای ایجاد ارتباط با پایگاه‌داده در سی‌شارپ، یک ارتباط Wisard است که می‌توان با قرار دادن یک کنترل داده‌ای نظیر DataGridView این ارتباط را ایجاد کرد. با ایجاد این ارتباط، نسخه‌ای از پرونده پایگاه‌داده در مسیر جاری پروژه قرار می‌گیرد و پرونده با پسوند xsd (DataSet.xsd + نام پایگاه‌داده) در بخش Solution Explorer قرار می‌گیرد. با ایجاد ارتباط با پایگاه داده به صورت Wisard، سه کامپوننت (DataSet و BindingSource و TableAdaptor) به صورت خودکار به فرم اضافه می‌شود. یک DataSet برای یک پایگاه‌داده و برای هر جدول یا View یک BindingSource و TableAdaptor متناظر اضافه می‌شود. البته اگر شما از طریق مقیدسازی کنترلی به جدول اقدام نکرده‌اید، کامپوننت‌های BindingSource و TableAdaptor اضافه نمی‌شوند.

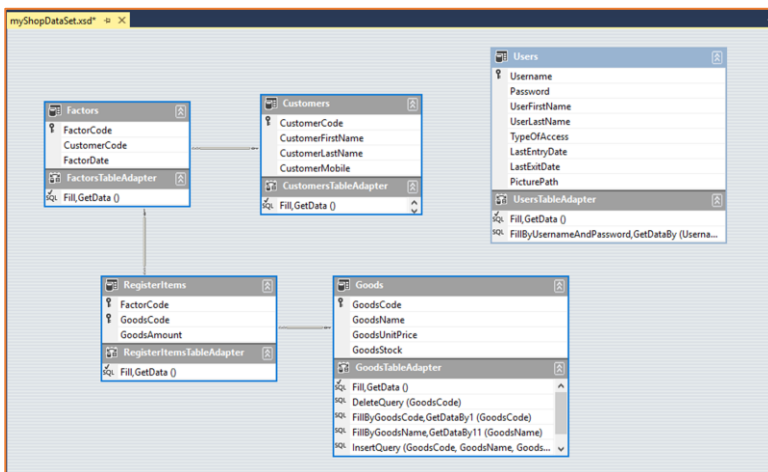
کلاس DataSet

کلاس DataSet برای نگهداری اطلاعات به دست آمده از پایگاه‌داده در حافظه به کار می‌رود. این کلاس شامل مجموعه‌ای از جدول‌ها، رابطه‌ها، قید و شرط‌ها و دیگر مواردی است که از پایگاه‌داده خوانده شده است. این کلاس خود همانند یک موتور کوچک پایگاه‌داده عمل می‌کند که می‌تواند داده‌ها را در درون خود در جدولی مجزا نگهداری کرده، به کاربر اجازه دهد که آن‌ها را ویرایش کند و با استفاده از کلاس DataView پرس‌وجوهای را روی داده‌های موجود در آن اجرا کند.

DataSet یک رونوشت (Cache) درون حافظه‌ای از داده‌ها است. در طول برنامه می‌توان داده‌های موجود در آن را حذف، ویرایش یا اضافه کرد و پس از اتمام تغییرات مورد نظر دوباره با استفاده از DataAdaptor به پایگاه‌داده متصل شده، تغییرات را در پایگاه‌داده ذخیره کرد.

کلاس DataSet از ساختار XML برای ذخیره داده‌ها استفاده می‌کند. به این ترتیب می‌توانید داده‌های موجود در یک نمونه از کلاس DataSet را به سادگی در یک فایل ذخیره کرده، یا آن را با استفاده از شبکه به کامپیوتر دیگری منتقل کنید. البته هنگام برنامه‌نویسی و کار با DataSet لازم نیست با آن‌ها در قالب XML رفتار کنید، بلکه کافی است تمام کارهای مورد نظر خود را با استفاده از ویژگی‌ها و متدهای موجود در DataSet انجام دهید.

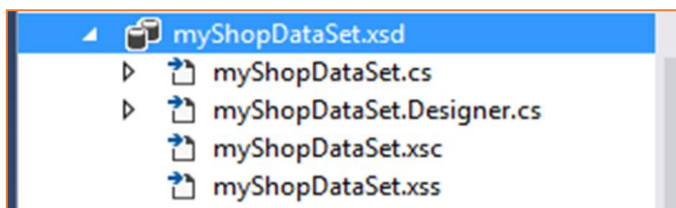
مانند هر سند XML یک DataSet نیز دارای یک الگو است. پرونده‌ای با پسوند XSD ساختارهای داده‌ای درون یک یا چند پرونده XML را شرح می‌دهد. شکل زیر پرونده myShopDataSet.xsd را نمایش می‌دهد.



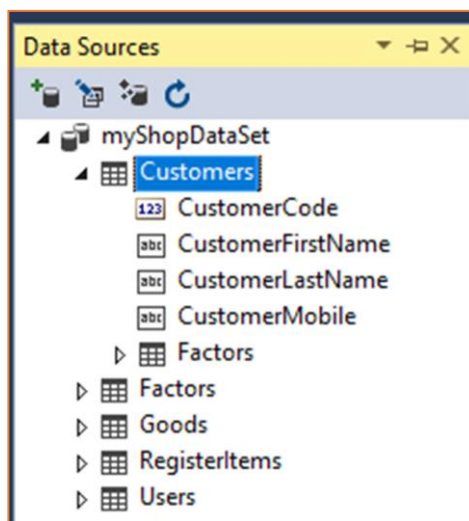
این پرونده (myShopDataSet.xsd) حاوی الگوی XML است که داده‌های myShopDataSet را نگهداری می‌کند. VS به وسیله این پرونده، یک کلاس را از کلاس DataSet مشتق می‌کند تا بتواند داده‌های دریافت شده از پایگاه داده را در نمونه کلاس نگهداری کند. البته تمام موارد نیز از دید برنامه‌نویس پنهان می‌ماند و توسط DataSet انجام می‌شود.


یکی از متدهای کلاس DataSet متد WriteXmlSchema است که می‌توان این Schema را در قالب یک پرونده xsd ایجاد کرد. این پرونده کاربردهای بسیاری دارد. برای همه عملیات روی پایگاه داده از جمله پرس و جوهای انتخابی (جست‌وجو و نمایش) و عملیاتی (درج، حذف و ویرایش) و تجمعی می‌توان یک متد روی جدول یا TableAdaptor مربوط به جدول ایجاد کرد و در برنامه به آسانی آن را فراخوانی کرد. برای اضافه کردن متد روی TableAdaptor جدول موردنظر راست کلیک کرده، گزینه Add Query را انتخاب کنید و با توجه به فهرست پرس و جوها، پرس و جوی مورد نظر را به صورت Wisard طراحی و با کلیک روی دکمه Query Builder به راحتی ویرایش کنید. نام مناسب برای این متدها قرار دهید.

با کلیک روی فلش پرونده xsd در بخش Solution Explorer می‌توان پرونده‌های دیگری را مشاهده کرد که پس زمینه عملیات ایجاد این Schema است.



برای مشاهده همه منابع داده از منوی View گزینه Other Windows و انتخاب گزینه Data Sources استفاده کنید و یا با فشار دادن همزمان کلیدهای Shift+Alt+D پنجره منابع داده را نمایش دهید. در منبع داده فهرست همه جدول‌ها یا دیدها به همراه فیلدهای جدول وجود دارد که با Drag کردن جدول یک کنترل DataGridView به همراه کنترل BindingNavigator برای پیمایش رکوردها به فرم اضافه می‌شود.



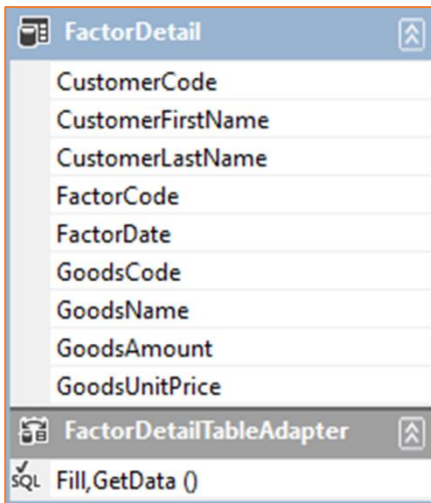
بدون استفاده از کنترل‌های مقید نظیر DataGridView می‌توانید از نماد جمع  پنجره Data Sources گزینه Add New Data Source یک منبع داده جدید اضافه کنید.

ایجاد پرونده xsd

برای ایجاد کردن یک پرونده با قالب xsd به صورت دستی ابتدا از طریق منوی Project → Add New Item عنصر DataSet را انتخاب کرده، با نوشتن نام مناسب دکمه Add را کلیک کنید. یک پرونده با نام xsd در بخش Solution Explorer اضافه می‌شود. با دابل کلیک روی این پرونده مشاهده می‌شود که این DataSet حاوی چیزی نیست. برای قرار دادن جدول و عناصر دیگر از منوی View پنجره Server Explorer را فعال کنید و از طریق گزینه Add Connection یک پایگاه داده جدید به این پنجره اضافه کنید و در جدول‌های پایگاه داده مورد نظر یک جدول را انتخاب کرده، روی پرونده xsd بکشید و رها کنید. مشاهده می‌شود که جدول به همراه فیلدها به این پرونده اضافه می‌شود و می‌توانید متدهای مورد نیاز را ایجاد کرده، در برنامه فراخوانی کنید.

افزودن یک TableAdaptor جدید

اگر بخواهیم داده‌هایی از چند جدول مرتبط به هم در یک پایگاه داده را مدیریت کنیم باید در پرونده xsd راست کلیک کرده و از گزینه Add، گزینه TableAdaptor را



انتخاب کنید و با ایجاد یک اتصال جدید و از طریق Query Builder یک پرس‌وجوی جدید از چند جدول مرتبط به هم ایجاد کنید. با زدن دکمه Finish یک جدول جدید به همراه یک TableAdaptor جدید ایجاد می‌شود و می‌توان آن را همانند دیگر TableAdaptorها مدیریت کرد.

جدول بالا از فیلدهای چهار جدول مرتبط به هم ایجاد شده است.

پاسخ به فعالیت‌ها

چه کنترل‌هایی به فرم برنامه اضافه شده است؟
پاسخ: پس از اتصال کنترل DataGridView به منبع داده سه کامپوننت به dictionaryDataSet و computerTableAdapter و computerBindingSource به فرم اضافه می‌شوند.

- کلید F7 را فشار دهید. آیا کدی به برنامه اضافه شده است؟
پاسخ: در رویداد Form_Load کدی به صورت زیر به برنامه اضافه می‌شود.

```
this.computerTableAdapter.Fill(this.dictionaryDataSet.computer);
```

تکمیل کارگاه
ص ۲۲۴

کارگاه ۸ - سفارشی کردن کنترل DataGridView

هدف از کارگاه ۸ آموزش استفاده از ویژگی‌های کاربردی کنترل DataGridView برای نمایش سفارشی و جذاب‌تر این کنترل است. از هنرجو بخواهید فهرست ویژگی‌های این کنترل را ببیند و با آزمون و خطا به کاربرد برخی از این ویژگی‌ها پی ببرد.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۲۲۶

مرحله ۲: فیلد ID باید از ستون‌های DataGridView حذف شود.
 مرحله ۳: ستون‌ها فارسی می‌شوند.
 مرحله ۴: کنترل DataGridView تمام فرم را در برمی‌گیرد.
 مرحله ۵: با تنظیم این ویژگی می‌توانید اندازه ستون‌ها را متناسب با سلیقه خود تنظیم کنید. با استفاده از ویژگی FillWeight هر ستون می‌توانید درصد عرض ستون را مشخص کنید.
 مرحله ۶: همه ستون‌های کنترل DataGridView حذف می‌شود.

فعالیت گارگاهی
ص ۲۲۶

جدول ۴ برخی از ویژگی‌های کنترل DataGridView را نمایش می‌دهد. این جدول را تکمیل کنید.

| نام ویژگی | شرح |
|-------------------------------|---|
| Dock | مشخص می‌کند که کدام گوشه از کنترل به فرم محدود شود. |
| AutoSizeColumnsMode | وضعیت اندازه خودکار برای ستون را مشخص می‌کند. |
| DefaultCellStyle | قالب‌بندی خانه‌های کنترل DataGridView |
| DataSource | نام جدول داده یا نام کنترل DataSource |
| HeaderText | عنوان در DataGridView |
| ColumnHeadersDefaultCellStyle | قالب‌بندی خانه‌های عنوان کنترل DataGridView |
| EnableHeadersVisualStyles | با False کردن این ویژگی دستور Application.EnableVisualStyles() در متد Main کلاس Program غیرفعال و در نتیجه قالب‌بندی روی DataGridView اعمال می‌شود. |

جدول مخاطبان تلفن همراه را روی فرم نمایش دهید.

پاسخ: پس از ایجاد پایگاه داده و جدول مخاطبان Contacts و یک پروژه ویندوزی به نام PhoneBook یک کنترل DataGridView به فرم اضافه کنید و از طریق Wisard یک منبع داده ایجاد کنید و کنترل DataGridView جدول مخاطبان را به منبع داده مقید کنید. با راست کلیک روی DataGridView و انتخاب گزینه Edit Columns فیلدهای اضافی را حذف کنید و نام سرستون‌ها را ویرایش کنید. ویژگی Dock را Fill قرار دهید. ویژگی AutoSizeColumnsMode را Fill قرار دهید. به کمک ویژگی ColumnHeadersDefaultCellStyle و EnableHeadersVisualStyle قالب‌بندی سرستون‌ها را انجام دهید و با ویژگی DefaultCellStyle قالب‌بندی خانه‌های DataGridView را انجام دهید. برای عدم نمایش خطوط عمودی DataGridView ویژگی CellBorderStyle را

| شماره همراه | نام | نام خانوادگی |
|-------------|-------|--------------|
| ۰۹۱۱۷۷۶۹۰۹۰ | رضا | محمدی |
| ۰۹۱۲۳۴۰۶۰۵۰ | باقر | فرهادی |
| ۰۹۱۵۹۰۰۷۰۸۰ | غلام | علوی |
| ۰۹۱۷۸۰۰۵۰۹۰ | بهرام | بهرامی |
| ۰۹۱۸۷۰۰۶۰۹۰ | رضا | یوسفی |

SingleHorizontal قرار دهید.

اگر بخواهید نام و نام خانوادگی را در قالب یک ستون نمایش دهید، پرس‌وجوی متد Fill را به صورت زیر ویرایش کنید.

```
SELECT MobileNumber, CountryCode, ContactFName+'
'+ContactLname AS ContactFullName, Oragnization, Email,
ContactPicture
FROM Contacts
```

نکته: در بخش Edit Column ویژگی DataPropertyName در کنترل DataGridView با نام مستعار ContactFullName تغییر یابد و ستون «نام خانوادگی» حذف شود.

از پایگاه داده فروشگاه، جدول کالا را روی فرم نمایش دهید.

همانند نمایش جدول مخاطبان روی فرم اقدام کنید.

| کد کالا | نام کالا | قیمت واحد | موجودی |
|---------|--------------|-----------|--------|
| 8 | قند | 2500 | 60 |
| 9 | دستمال کاغذی | 2000 | 30 |
| 10 | ببیر خامه ای | 3200 | 15 |
| 11 | کلوچه | 500 | 100 |
| 12 | کیک | 500 | 200 |
| 13 | خامه | 3000 | 30 |
| 14 | ژرد گندم | 2000 | 100 |
| 15 | ناگت مرغ | 6000 | 20 |
| 16 | بفک | 500 | 40 |
| 17 | جیمس | 1500 | 29 |
| 18 | شیرت پرتقال | 6000 | 20 |
| 19 | گرم کاکائو | 3500 | 60 |
| 20 | مایع ظرفشویی | 3500 | 12 |
| 1 | شیر | 2000 | 100 |

پژوهش
ص ۲۲۷

در مورد ویژگی‌های دیگر کنترل DataGridView تحقیق کنید.

| شرح | نام ویژگی |
|--|-----------------------------------|
| به کاربر اجازه می‌دهد به DataGridView سطر جدید اضافه کند. | AllowUserToAddRow |
| به کاربر اجازه می‌دهد سطر دلخواه را انتخاب و با دکمه Delete صفحه کلید سطر انتخاب‌شده را از DataGridView حذف کند. نکته اینکه این سطر فقط از DataGridView حذف می‌شود و از پایگاه داده حذف نمی‌شود. | AllowUserToDeleteRow |
| به کاربر اجازه می‌دهد که اندازه (عرض) ستون‌ها را با استفاده از ماوس تغییر دهد. | AllowUserToResizeColumns |
| به کاربر اجازه می‌دهد که بتواند اندازه (ارتفاع) سطرها را با استفاده از ماوس تغییر دهد. | AllowUserToResizeRows |
| با استفاده از این ویژگی می‌توانید ظاهر DataGridView از جمله رنگ زمینه، قلم، رنگ متن، رنگ سطرهای فرد را تغییر دهید. | AlternatingRowsDefaultCell |
| با تنظیم این ویژگی می‌توانید اندازه ستون‌ها را متناسب با سلیقه خود تنظیم کنید. | AutoSizeColumnsMode |

| نام ویژگی | شرح |
|---------------------------------|--|
| AutoSizeRowsMode | با تنظیم این ویژگی می‌توانید اندازه سطرها را متناسب با سلیقه خود تنظیم کنید. |
| BackgroundColor | رنگ پشت زمینه DataGridView را تعیین می‌کند. |
| BorderStyle | نوع خط کادر DataGridView را تعیین می‌کند. |
| CellBorderStyle | نوع خطوط بین سلول‌های DataGridView را تعیین می‌کند. از جمله نوع‌های موجود در این ویژگی خط ساده، برجسته افقی، برجسته عمودی و... هستند. |
| ColumnHeadersBorderStyle | نوع خطوط بین سلول‌های عنوان ستون‌ها را تعیین می‌کند. |
| ColumnHeaderHeight | این ویژگی، اندازه ارتفاع عنوان ستون‌ها را تعیین می‌کند. |
| ColumnHeadersVisible | با False کردن این ویژگی، عنوان ستون‌ها مخفی می‌شوند. |
| Columns | با این ویژگی می‌توانید ستون‌های DataGridView را مدیریت (اضافه، حذف و ویرایش) کنید. |
| Cursor | شکل نماد ماوس را برای زمانی که وارد DataGridView می‌شود، تعیین می‌کند. مثلاً با انتخاب علامت Hand، وقتی ماوس روی DataGridView قرار می‌گیرد نماد ماوس به شکل دست تغییر می‌کند. |
| DefaultCellStyle | با استفاده از این ویژگی می‌توانید ظاهر سلول‌های سطرهای فرد DataGridView را تغییر دهید از جمله رنگ زمینه، قلم، رنگ متن، تنظیمات متن و... |
| Dock | با استفاده از این ویژگی می‌توانید اندازه کنترل DataGridView را متناسب با اندازه فرم تنظیم کنید. مثلاً می‌توانید تعیین کنید که کنترل DataGridView به طور کامل سمت چپ فرم را بپوشاند و با تغییر اندازه فرم، اندازه DataGridView نیز تغییر کند. |
| MultiSelect | با استفاده از این ویژگی می‌توانید تعیین کنید که آیا کاربر می‌تواند یک سطر یا بیش از یک سطر را همزمان انتخاب (Select) کند. |

| نام ویژگی | شرح |
|------------------------------|---|
| ReadOnly | با True کردن این ویژگی، کاربر نمی‌تواند متن درون سلول‌ها را ویرایش کند و تنها می‌تواند متن درون DataGridView را مشاهده کند. |
| RightToLeft | با True کردن این ویژگی، سلول‌های درون DataGridView، در سمت راست کنترل DataGridView قرار می‌گیرند. در صورتی که متن درون DataGridView فارسی است، این گزینه را روی Yes قرار دهید. |
| RowHeadersBorderStyle | نوع خطوط بین سلول‌های عنوان سطرها (علامت ستاره کنار هر سطر) DataGridView را تعیین می‌کند. |
| RowHeaderVisible | با False کردن این ویژگی، سلول‌های خالی که در اول هر سطر قرار دارند، مخفی می‌شوند. |
| RowHeaderWidth | این ویژگی، طول سلول‌های خالی را که در اول هر سطر قرار دارند، تعیین می‌کند. |
| ScrollBars | تعیین می‌کند که اگر تعداد سطرها و یا ستون‌های DataGridView بیشتر از اندازه طول و یا عرض کنترل شد، آیا نوار پیمایش در پایین و یا سمت راست کنترل نمایش داده شود یا خیر. یا اینکه تنها نوار افقی نمایش داده شود و یا تنها نوار عمودی و یا هر دوی آن‌ها و یا هیچکدام. |
| SelectionMode | این ویژگی تعیین می‌کند، زمانی که روی یکی از سلول‌ها کلیک شد، آن سطر یا ستون به طور کامل انتخاب شود. |

کارگاه ۹ – مقیدسازی کنترل‌ها

هدف از این کارگاه توضیح مفهوم مقیدسازی و نحوه مقیدسازی کنترل‌ها با Wisard و کدنویسی است. از هنرجو بخواهید که کنترل‌های مقید و نامقید را در جعبه ابزار مشخص کند.

پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۲۲۷

مرحله ۳: چه مقداری در txtWordName قرار می‌گیرد؟ مقدار واژه که روی سطر آن کلیک شده است.

مرحله ۴: برای مقیدسازی کنترل کادرفهرست چگونه ویژگی DataSource و DisplayMember را تغییر می‌دهید؟

پاسخ: از طریق فهرست DataSource کنترل کادرفهرست computerBindingSource را انتخاب و از طریق فهرست DisplayMember یکی از فیلدهای جدول computer را انتخاب کنید.

مرحله ۶: ویژگی DataSource کنترل BindingNavigator را computerBindingSource قرار دهید. با اجرای برنامه به وسیله این کنترل می‌توان رکوردها را روی کنترل‌های مقید نظیر DataGridView و listBox پیمایش کرد.

کنجکاو
ص ۲۲۸

چه کنترل‌هایی را می‌توان مقید کرد؟ نام چند کنترل را نام ببرید.
پاسخ: کنترل‌هایی که ویژگی DataBinding دارند مقید هستند مثل checkbox و radioButton

چگونه با کدنویسی می‌توان یک کنترل را نامقید کرد؟
پاسخ: برای نامقید کردن کنترل‌ها باید مقدار DataSource را null کنید.

```
listBox1.DataSource = null;
```

برای نامقید کردن کنترل‌هایی که ویژگی DataSource ندارند باید از طریق متد DataBinding.Clear اقدام کرد.

```
txtWordName.DataBindings.Clear();
```

کارگاه ۱۰ - بارگذاری داده و پیمایش رکوردها در

DataGridView

هدف این کارگاه آموزش استفاده از متد Fill کامپوننت TableAdaptor برای بارگذاری داده و متدهای پیمایش (Move) کامپوننت BindingSource برای پیمایش رکوردها در برنامه نویسی است.

هنگام ایجاد اتصال به صورت Wisard دو کامپوننت TableAdaptor و BindingSource به فرم اضافه می‌شوند. در رویداد Form_Load نیز کد زیر به صورت خودکار اضافه می‌شود.

```
this.goodsTableAdapter.Fill(this.myShopDataSet.Goods);
```

با دابل کلیک روی پرونده xsd متد Fill جدول Goods را به هنرجو نشان دهید و نحوه فراخوانی این متد را در کدنویسی آموزش دهید.

پاسخ به فعالیت‌ها

کنجکاوی
ص ۲۲۹

- کامپوننت MyShopDataSet شامل چه جدول‌های دیگری است؟ اگر در هنگام ایجاد منبع داده به صورت Wisard همه جدول‌ها انتخاب شده باشد، جدول Customers و Factor و RegisterItems نیز در MyShopDataSet وجود دارد.

- آیا می‌توان برای مقیدسازی کنترل dgvGoods دستور دیگری نوشت؟

```
this.dgvGoods.DataSource = goodsTableAdapter.GetData();
```

مرحله ۴: برای رفتن به آخرین رکورد (کالا) چه دستوری می‌نویسید؟

```
goodsBindingSource.MoveLast();
```

مرحله ۵: برای رفتن به رکورد بعدی (کالا) چه دستوری می‌نویسید؟

```
goodsBindingSource.MoveNext();
```

کارگاه ۱۱ - دسترسی به خانه‌های DataGridView

هدف از این کارگاه دسترسی به خانه‌های کنترل DataGridView از طریق کدنویسی است. اهمیت دسترسی به خانه‌های این کنترل برای نمایش در کادرمین یا برچسب و یا انجام برخی محاسبات روی خانه‌های DataGridView لازم است. برای یادگیری بهتر این کارگاه، مفهوم Row و Cell را در کنترل DataGridView به هنرجو آموزش دهید. در این کارگاه برای دسترسی به رکورد جاری ویژگی CurrentCell.RowIndex و مفهوم رویداد CellClick نیز آموزش داده شود.

پاسخ به فعالیت‌ها

کنجکاوی
ص ۲۳۰

مقدار اندیس(شماره) اولین رکورد چیست؟

پاسخ: صفر

کارگاه ۱۲ - جست‌وجو در جدول

هدف از این کارگاه جست‌وجوی دقیق و مشابه واژه‌ها در جدول است.

- از هنرجو در مورد کاربرد دستور Select که در پودمان اول فراگرفته، سوال کنید و بخواهید ساختار آن را یادآوری کند و سپس بخواهید وارد پرونده xsd شود و روی جدول یا TableAdaptor راست کلیک کرده، مشاهدات خود را بگویند و یک متد جدید را با پرس‌وجو Select ایجاد کرده، نام مناسب برای متد قرار دهد.

- در مورد دکمه Query Builder برای ایجاد پرس و جوی انتخابی با شرط توضیح دهید. در پنجره Query Builder نحوه اجرای یک پرس و جوی را نشان دهید.
- عملکرد عملگر Like در پرس و جوی Select را با چند مثال در Access به هنرجو نشان دهید و از او بخواهید که متد جدیدی برای جست و جوی واژه‌های مشابه ایجاد کند.
- نحوه فراخوانی متدهای Select را در برنامه به صورت زیر به هنرجو آموزش دهید.

(ورودی‌های متد، نام جدول، نام DataSet) نام متد • نام TableAdaptor

کار با داده‌ها با استفاده از DataView

در ADO.NET برای انجام عملیات (جست و جوی، مرتب سازی و فیلتر کردن) روی داده‌ها در یک DataTable دو روش وجود دارد:

(۱) استفاده از متد Select شیء DataTable

(۲) استفاده از شیء DataView

شیء DataView دارای توانایی بیشتری نسبت به متد Select شیء DataTable است و کار با آن نیز ساده‌تر است. شیء DataView مانند یک منبع برای اتصال کنترل‌های دیگر به آن مورد استفاده قرار می‌گیرد. شیء DataView نمایی از یک شیء DataTable است که برای جست و جوی، فیلتر و مرتب‌سازی و حتی دستکاری داده‌های آن شیء DataTable مورد استفاده قرار می‌گیرد.

هر شیء DataView شامل یک مجموعه DataRowView است که مشابه شیء DataRow در DataTable هستند. این اشیاء دارای ویژگی‌هایی هستند که در زیر به اختصار توضیح داده شده‌اند:

■ **DataView**: شیء DataView را برمی‌گرداند که سطر جاری به آن تعلق دارد.

■ **Item**: مقدار یک ستون از سطر فعلی را برمی‌گرداند.

■ **Row**: شیء DataRow را برمی‌گرداند که در حال حاضر در حال بررسی است.

اشیای DataView در داخل یک شیء دیگر به نام DataViewManager قرار می‌گیرند. رابطه بین این دو شیء مانند رابطه بین DataSet و DataTable است، با این تفاوت که امکان دارد یک شیء DataTable دارای چند شیء DataView باشد که برای انجام جست و جوی و فیلتر کردن مختلف روی داده‌های آن استفاده می‌شوند. در حقیقت در زمان اتصال یک کنترل به یک شیء DataSet در پروسه Data-Binding آن کنترل به شیء DataView مربوط به آن DataSet متصل می‌شود که با ویژگی DefaultViewManager شیء DataSet مشخص می‌شود.

در مثال زیر یک شیء DataView می‌سازیم:

```

DataGridView myDataGridView = new DataGridView();
myDataGridView.Table = dictionaryDataSet.computer;
1stWordName.DataSource = myDataGridView;

```

در مثال بالا ابتدا یک شیء **DataGridView** جدید به نام **myDataGridView** ایجاد شده است. سپس ویژگی **Table** آن برابر با جدول کامپوننت **DataSet** که در برنامه کاربردی ما موجود است، قرار داده شده است. شیء **DataGridView** دارای ویژگی‌هایی است که در زیر به اختصار توضیح داده شده‌اند:

■ **AllowDelete** و **AllowEdit** و **AllowNew**: با تنظیم این مقادیر **Boolean** می‌توان با دستکاری و اضافه و حذف مقادیر در **DataGridView** مقادیر متناظر آن‌ها را در شیء **DataTable** مربوط تغییر داد.

■ **Count**: تعداد اشیای **RowView** موجود در **DataGridView** جاری را برمی‌گرداند (مشابه ویژگی **Count** از شیء **DataTable**).

■ **Item**: شیء **DataRowView** را برمی‌گرداند که با اندیس به وسیله پارامتر به آن ارسال شده است.

■ **RowFilter**: عبارتی که برای فیلتر کردن سطرهای درون **DataGridView** استفاده می‌شود.

■ **Sort**: عبارتی که برای مرتب کردن سطرهای درون **DataGridView** به کار می‌رود.

■ **Table**: شیء **DataTable** متناظر با **DataGridView** را مشخص می‌کند.

برای استفاده از ویژگی‌ها **RowFilter** و **Sort** عبارت‌هایی هستند که مشابه عبارت‌های محاسباتی برای متد **Select** از شیء **DataTable** مورد استفاده قرار می‌گیرند. برای تنظیم مقادیر ویژگی **Sort** که مربوط به روش مرتب‌سازی سطرهای **DataRow** است، لازم است مقادیر ستون‌های مورد نظر جهت مرتب‌سازی را به ترتیب در داخل علامت " " قرار دهیم. برای مثال برای مرتب‌سازی یک **DataGridView** به ترتیب بر حسب ستون‌های **Date** و **Author** به روش زیر عمل می‌کنیم:

```

myDataGridView.Sort = "WordName";

```

برای تنظیم مقدار **RowFilter** می‌توان از عبارات **DataColumn** استفاده کرد. این عبارات شامل یک سری توابع، عملگرها، عبارات و نام ستون‌ها است که جهت فیلتر کردن مورد استفاده قرار می‌گیرند. در این عبارات می‌توان از تمام عملیات دستکاری روی رشته‌ها و متغیرهای رشته‌ای استفاده کرد. به علاوه برای استفاده از مقادیر تاریخ باید آن‌ها را در بین علامات # # قرار داد.

برای استفاده از یک ستون خاص در این عبارات تنها کافی است، از نام آن ستون استفاده کنید. برای استفاده از سطرهای فرزند از عبارت **Child** قبل از آن‌ها و برای استفاده از سطر مادر نیز از عبارت **Parent** قبل از آن‌ها استفاده کنید.

برخی از توابع قابل استفاده در این عبارات عبارتند از: Sum و Avg و Min و Max و Count و برخی از عملگرهای منطقی مورد استفاده عبارتند از: And و Or و Not و < و In و Like.

برنامه جست و جوی واژه (کارگاه ۱۱) با استفاده از DataView

ابتدا نمونه‌ای از کلاس DataView در بخش کلاس فرم ایجاد کنید.

```
public partial class Form1 : Form
{
```

```
    DataView myDataView=new DataView();
```

در رویداد Form_Load کدهای زیر را برای مقیدسازی کنترل کادرفهرست به DataView بنویسید.

```
private void Form1_Load(object sender, EventArgs e)
{
```

```
    this.computerTableAdapter.Fill(this.dictionaryDataSet.computer);
```

```
    myDataView.Table = dictionaryDataSet.computer;
```

```
    lstWordName.DataSource = myDataView;
```

```
}
```

برای جست و جوی واژه می‌توان فقط از نمونه کلاس DataView استفاده کرد.

```
private void txtSearchWord_TextChanged(object sender, EventArgs e)
{
```

```
    myDataView.RowFilter = "";
```

```
    if (txtSearchWord.Text.Trim().Length > 0)
```

```
        myDataView.RowFilter = "WordName Like '" + txtSearchWord.Text + "%'";
```

```
        lstWordName.DataSource = myDataView;
```

```
        txtWordMeans.DataBindings.Clear();
```

```
        txtWordMeans.DataBindings.Add("Text", myDataView, "means");
```

```
}
```

پاسخ به فعالیت‌ها

روی چه کنترل‌هایی می‌توان کنترل SplitContainer را قرار داد؟
پاسخ: روی کنترل‌هایی که ظرف (container) هستند مثل groupBox و Panel

کنجکاو

ص ۲۲۲

عملکرد دستور SELECT چیست؟
پاسخ: انتخاب رکوردهایی که نام واژه آن برابر با ورودی باشد.

کنجکاو

ص ۲۲۳

تکمیل کارگاه
ص ۲۳۴

مرحله ۸: عملکرد دستور if چیست؟
پاسخ: در صورتی جست و جو انجام می شود که در کادرمتن txtSearchWord متنی نوشته شده باشد.
مرحله ۱۰: عملکرد دستور Like چیست؟
پاسخ: جست و جوی بر اساس شباهت ابتدای واژه صورت می گیرد.

کنجکاوی
ص ۲۳۴

نوع ویژگی DataSource کنترل computerBindingSource چیست؟
پاسخ: DataSet

فعالیت منزل
ص ۲۳۴

- قابلیت جست و جو بر اساس فیلد نام و کد کالا را برای فرم کالا ایجاد کنید.
پاسخ: ابتدا متدی به نام FillByGoodsCode با پرس و جوی زیر برای جست و جو بر اساس کد کالا ایجاد می کنیم:

```
SELECT GoodsCode, GoodsName, GoodsUnitPrice,  
GoodsStock FROM Goods  
WHERE (GoodsCode = ?)
```

سپس متدی به نام FillByGoodsName با پرس و جوی زیر برای جست و جوی مشابه بر اساس نام کالا ایجاد می کنیم:

```
SELECT GoodsCode, GoodsName, GoodsUnitPrice,  
GoodsStock FROM Goods  
WHERE (GoodsName like ? + '%')
```

در فرم جست و جو، نام کادرمتن مربوط به کد کالا txtSGoodsCode و نام کادرمتن مربوط به نام کالا txtSGoodsName است.

```
private void txtSGoodsCode_TextChanged(object sender,  
EventArgs e)  
{  
    if (txtSGoodsCode.Text.Length > 0)  
  
goodsTableAdapter.FillByGoodsCode(myShopDataSet.Goods,  
int.Parse(txtSGoodsCode.Text));  
    else  
        goodsTableAdapter.Fill(myShopDataSet.Goods);  
}  
private void txtSGoodsName_TextChanged(object sender,  
EventArgs e)  
{  
    if (txtSGoodsName.Text.Length > 0)
```

```

goodsTableAdapter.FillByGoodsName(myShopDataSet.Goods,
txtSGoodsName.Text);
else
goodsTableAdapter.Fill(myShopDataSet.Goods);
}

```

- قابلیت جست‌وجو براساس نام و شماره همراه را برای جدول مخاطبان ایجاد کنید. با دابل‌کلیک روی PhoneBookDataSet.xsd یک متد برای ContactsTableAdapter با نام FillByContactFullName با پرس‌وجوی زیر ایجاد کنید.

پاسخ:

```

SELECT      MobileNumber, ContactFName, ContactLName,
Oragnization, Email, ContactPicture, CountryCode FROM
Contacts

```

```

WHERE      (ContactFName LIKE '%' + ? + '%') AND
(ContactLName LIKE '%' + ? + '%')

```

دو کنترل txtFname و txtLname به فرم اضافه کنید و در رویداد TextChanged کد زیر را بنویسید.

```

contactsTableAdapter.FillByContactFullName(
PhoneBookDataSet.Contacts,txtFname.Text, txtLname.Text);

```

کارگاه ۱۳ - جست‌وجوی ترکیبی در جدول

هدف از این کارگاه جست‌وجو در جدول بر اساس چند فیلد است که در این کارگاه جست‌وجو بر اساس دو فیلد نام کاربری و گذرواژه انجام شده است. با استفاده از دکمه Query Builder به هنرجو نشان دهید که به آسانی می‌توان یک پرس‌وجوی انتخابی با ترکیب چند شرط ایجاد کرد. ویژگی Count را که تعداد رکوردهای جدول DataSet را نمایش می‌دهد، با اجرای گام به گام به هنرجو توضیح دهید.

پاسخ به فعالیت‌ها

مرحله ۶: مفهوم شرط دستور if چیست؟

پاسخ: اگر نتیجه اجرای پرس‌وجو حداقل یک رکورد داشته باشد، به این معنا است که کاربر معتبر و فرم MainForm نمایش داده می‌شود و گرنه پیام نامعتبر بودن کاربر نمایش داده می‌شود.

- برنامه را اجرا کنید. عملکرد متد btnLogon_Click چیست؟

پاسخ: متد FillByUsernameAndPass پرس‌وجوی انتخابی با شرط نام کاربری و گذرواژه است و اگر تعداد سطرهای حاصل از اجرای پرس‌وجو بیش از صفر باشد کاربر معتبر و گرنه نامعتبر است.

تکمیل کارگاه
ص ۲۳۵

برنامه ورود به فروشگاه را طوری تغییر دهید که با زدن تیک کادر علامت، نام کاربری برای ورود بعدی ذخیره شود و تصویر کاربر نیز نمایش داده شود. پاسخ: متد زیر را پس از ورود به فرم اصلی فراخوانی کنید.

```
private void SaveUserName()
{
    if (checkBoxSaveUserName.Checked)
        File.WriteAllText("SaveUserName.txt",
txtUsername.Text);
    else
        if (File.Exists("SaveUserName.txt"))
            File.Delete("SaveUserName.txt");
}
```

در رویداد Form_Load بررسی کنید اگر فایل ذخیره کاربری موجود است نام کاربر در کادرمتن نام کاربری نمایش داده شود و تصویر کاربر در کادرتصویر نمایش داده شود.

یک متد با نام FillByUserName ایجاد کنید که جست‌وجو را فقط بر اساس نام کاربری انجام دهد.

```
private void Form1_Load(object sender, EventArgs e)
{
    if (File.Exists("SaveUserName.txt"))
    {
        checkBoxSaveUserName.Checked = true;
        txtUsername.Text =
File.ReadAllText("SaveUserName.txt");
        usersTableAdapter.FillByUserName (
myShopDataSet.Users,txtUsername.Text.Trim;())
string strFilename =
myShopDataSet.Users.Rows[0]["PicturePath"].ToString();
        if (File.Exists(strFilename))
            picUser.ImageLocation = strFilename;
    }
}
```

نکته: در جدول کاربران می‌تواند چندین کاربر وجود داشته باشد. این برنامه تمرینی برای یک برنامه در شبکه است که نام کاربری هر فرد روی سیستم خودش ذخیره شود و به راحتی بدون ورود نام کاربری به برنامه چند کاربره وارد شود.

با بازکردن جدول کاربران در پایگاه داده هر فردی می‌تواند نام کاربری و گذرواژه همه کاربران را ببیند و به برنامه فروشگاه وارد شود. برای امنیت بیشتر برنامه چه کار باید کرد؟

پاسخ: یکم، از راه‌هایی که دسترسی افراد به پایگاه داده را ناممکن می‌سازد، گذاشتن رمز روی پرونده پایگاه داده اکسس و روش دیگر رمزگذاری داده‌های درون پایگاه داده است.

درهم‌ریزی داده‌ها (Hashing) روشی برای غیرقابل خواندن کردن پرونده‌ها و نوشته‌ها است که معمولاً از یک زوج الگوریتم تشکیل می‌شود؛ یکی برای رمزگذاری و دیگری برای رمزگشایی (Decoding). به‌طور مثال در یک حلقه for تک تک کاراکترها را با XOR، ۲۵۶، کرده و جواب را می‌نویسید و بخش رمزگشایی را با الگوریتم دوم که قرینه الگوریتم اول هست به حالت اول برمی‌گردانیم. البته استفاده از XOR یک مثال ساده است و برای مسائل مهم کسی از این روش استفاده نمی‌کند. یکی از الگوریتم‌های معروف در زمینه کدکردن، الگوریتم MD5 است.

الگوریتم MD5 (منبع MSDN مایکروسافت):

```
using System.Security.Cryptography;
static string GetMd5Hash(MD5 md5Hash, string input)
{
    // Convert the input string to a byte array and compute
    // the hash.
    byte[] data =
    md5Hash.ComputeHash(Encoding.UTF8.GetBytes(input));
    // Create a new StringBuilder to collect the bytes
    // and create a string.
    StringBuilder sBuilder = new StringBuilder();
    // Loop through each byte of the hashed data
    // and format each one as a hexadecimal string.
    for (int i = 0; i < data.Length; i++)
    {
        sBuilder.Append(data[i].ToString("x2"));
    }
    // Return the hexadecimal string.
    return sBuilder.ToString();
}
```

متد GetMd5Hash ورودی input را با md5Hash کدگذاری و در یک رشته sBuilder قرار می‌دهد.

```
// Verify a hash against a string.
static bool VerifyMd5Hash(MD5 md5Hash, string input, string
hash)
{
    // Hash the input.
    string hashOfInput = GetMd5Hash(md5Hash, input);
```

```

// Create a StringComparison and compare the hashes.
StringComparer comparer =
StringComparer.OrdinalIgnoreCase; if (0 ==
comparer.Compare(hashOfInput, hash))
{
return true;
}
else
{
return false;
}
}
}

```

متد VerifyMd5Hash بررسی می‌کند که اگر ورودی input با md5Hash کد شود، آیا رشته کدگذاری شده hash معادل است یا خیر.

نحوه استفاده از الگوریتم MD5 در برنامه:

برنامه اضافه کردن کاربر جدید را به صورت زیر تغییر دهید:

```

MD5 md5hash = MD5.Create();
usersTableAdapter.AddUser(txtUsername.Text,
GetMd5Hash(md5hash, txtPassword.Text), txtFirstName.Text,
txtLastName.Text);

```

پس از اضافه شدن کاربر جدید مشاهده می‌شود که گذرواژه به صورت یک رمز در جدول اضافه شده است.

برنامه اعتبارسنجی کاربر به صورت زیر تغییر می‌کند:

```

MD5 md5hash = MD5.Create();
usersTableAdapter.FillByUsernameAndPassword(myShopDataSet.
Users,
txtUsername.Text, GetMd5Hash(md5hash, txtPassword.Text));
if (myShopDataSet.Users.Rows.Count > 0)
{
succeeded = true;
this.Close();
}
else

```

```

MessageBox.Show("نام کاربری یا گذرواژه اشتباه است", "خطا",
MessageBoxButtons.OK, MessageBoxIcon.Error);

```

کارگاه ۱۴ - درج یک رکورد در جدول

هدف از این کارگاه نحوه ایجاد متد درج در جدول و فراخوانی آن در برنامه و آشنایی با دستور INSERT INTO است. از هنرجو بخواهید ساختار دستور INSERT INTO را در پودمان اول فراگرفته توضیح دهد. سپس از او بخواهید متدی را در TableAdaptor جدول کالا در پرونده xsd ایجاد کند که دستور درج رکورد را انجام دهد. نحوه فراخوانی متد پرس و جوی عملیاتی را به صورت زیر به هنرجو آموزش دهید. (ورودی‌های متد) نام متد • نام TableAdaptor

پاسخ به فعالیت‌ها

ساختار دستور INSERT INTO در زبان SQL چیست؟
پاسخ: مقدار VALUES (.... , نام فیلد ۲, نام فیلد ۱) نام جدول INSERT INTO (.... مقدار فیلد ۲, فیلد ۱)

کنجکاو
ص ۲۳۷

تکمیل کارگاه
ص ۲۳۷

مرحله ۲:

- متد InsertQuery دارای چند آرگومان ورودی است؟ ۴ آرگومان.
- چرا برخی از ورودی‌ها عددی است؟ چون نوع فیلد در جدول عددی است.
- چرا از دستور if در برنامه استفاده شده است؟ چون ممکن است کاربر برخی از ورودی‌هایی که لازم است مقدار داشته باشند را خالی گذاشته باشد.
- عملکرد متد رویداد btnAddGoods_Click چیست؟ در صورت ورود نام، قیمت و کد کالا یک کالای جدید به جدول Goods اضافه می‌کند.
- عملکرد دستور goodsTableAdapter.InsertQuery چیست؟ متدی برای اضافه کردن رکورد جدید به جدول Goods است.
- آیا رکورد جدید در dgvGoods نمایش داده می‌شود؟ چرا؟ بله. چون پس از اضافه شدن رکورد در جدول متد Fill نمایش داده‌ها را در dgvGoods به روز می‌کند.
- آیا رکورد اضافه شده در dgvGoods وجود دارد؟ خیر.
- جدول Goods پایگاه داده MyShop را باز کنید. آیا رکورد اضافه شده در جدول وجود دارد؟ خیر

مرحله ۳:

- آیا در جدول Goods این کالا اضافه شده است؟ بله

فعالیت منزل
ص ۲۳۸

- برای جدول مشتری برنامه درج رکورد را بنویسید.
- برای جدول مخاطبان برنامه ثبت مخاطب جدید را بنویسید.
پاسخ به این فعالیت منزل به پیوست در لوح نوری همراه کتاب است.

کارگاه ۱۵ - حذف یک رکورد در جدول

هدف از این کارگاه نحوه ایجاد متد حذف در جدول و فراخوانی آن در برنامه و آشنایی با دستور DELETE است. از هنرجو بخواهید ساختار دستور DELETE را که در پودمان اول فراگرفته توضیح دهد. سپس از او بخواهید متدی را در TableAdaptor جدول کالا در پرونده xsd ایجاد کند که دستور حذف رکورد را انجام دهد.
پاسخ به فعالیت‌ها

تکمیل کارگاه
ص ۲۳۹

۱) دستور را تغییر دهید تا حذف بر اساس نام کالا باشد.
DELETE FROM Goods WHERE (GoodsName = ?)
۲) دستور را برای حذف تمامی رکوردها تغییر دهید.
DELETE FROM Goods

فعالیت منزل
ص ۲۴۰

برای جدول مشتری برنامه حذف رکورد را بنویسید. (فیلم مربوطه در لوح نوری همراه کتاب موجود است).

برای جدول مخاطبان برنامه حذف مخاطب را بنویسید.

پاسخ: با دابل کلیک روی PhoneBookDataSet.xsd در بخش Solution Explorer یک متد برای حذف مخاطب ایجاد کنید. روی جدول Contacts راست کلیک کنید. گزینه Query از منوی Add و سپس گزینه Use SQL statement را انتخاب و دکمه Next را کلیک کنید، سپس گزینه Delete را انتخاب و دکمه Next را کلیک کنید. سپس پرس و جوی زیر را با Query Builder ایجاد کنید.
DELETE FROM Contacts WHERE (MobileNumber = ?) AND (CountryCode = ?)

در فرم مدیریت مخاطب یک دکمه به نام btnDeleteContact اضافه کنید و کد زیر را بنویسید.

```
private void btnDeleteContact_Click(object sender, EventArgs e)
{
    if (txtCountyCode.Text.Length > 0 && txtMobileNO.Text.Length > 0)
        contactsTableAdapter.DeleteQuery(txtMobileNO.Text, txtCountyCode.Text);
}
```

کارگاه ۱۶ – ویرایش یک رکورد در جدول

هدف از این کارگاه نحوه ایجاد متد ویرایش در جدول و فراخوانی آن در برنامه و آشنایی با دستور UPDATE است. از هنرجو بخواهید ساختار دستور UPDATE را که در پودمان اول فراگرفته توضیح دهد. سپس متدی را در TableAdaptor جدول کالا در پرونده xsd ایجاد کند که دستور ویرایش رکورد را انجام دهد. برای پاسخ فعالیت‌های منزل این کارگاه به لوح نوری همراه کتاب رجوع کنید.

کارگاه ۱۷ – درج و حذف سطر از DataGridView

هدف از این کارگاه نحوه اضافه و حذف کردن سطر از یک کنترل DataGridView نامقید است. با استفاده از متد Add و مفهوم Row و Cell کنترل DataGridView نحوه افزودن سطر جدید را به هنرجو آموزش دهید. متد RemoveAt یکی از متدهای مشترک بین کادرفهرست و کنترل DataGridView است که می‌توانید از این وجه تشابه، متد RemoveAt را آموزش دهید. برای پیمایش سطر به سطر کنترل DataGridView می‌توان از حلقه for استفاده کرد.

پاسخ به فعالیت‌ها

مرحله ۲: ویژگی‌های ImageLayout و FillWeight را به دلخواه تغییر دهید. کاربرد این ویژگی‌ها چیست؟

ویژگی ImageLayout نحوه قرارگیری تصویر در ستونی است که کنترل کادرتصویر قرارداد که در این کارگاه Stretch قرار داده می‌شود.

ویژگی FillWeight درصد قرارگیری عرض ستون است هنگامی که ویژگی AutoSizeMode برابر با Fill قرار داده می‌شود.

۸- متد رویداد کلیک دکمه جست‌وجوی مشتری (btnCustomerTab) در سربرگ برگه خرید را بنویسید.

```
tabShop.SelectedTab = tabPageCustomers; // فعال کردن سربرگ مشتری
txtSearchCustomerLastName.Focus();
```

۹- متد رویداد کلیک دکمه «درج در برگه خرید» در سربرگ مشتری را بنویسید.

```
private void btnCustomerInsertFactor_Click(object sender, EventArgs e)
{
    lblCustomerFullName.Text = txtCustomerFirstName.Text +
    " " +
    txtCustomerLastName.Text;
    txtFCustomerCode.Text = txtFCustomerCode.Text;
```

تکمیل کارگاه
ص ۲۴۱

```

فعال کردن سربرگ برگه //
tabShop.SelectedTab = tabPageFactor;
خرید
txtFCustomerCode.Text = txtCustomerCode.Text;
lblCustomerName.Text = txtCustomerName.Text + " " +
txtCustomerFamily.Text;
txtFCustomerCode.Focus();
}

```

مرحله ۱۰:

■ چگونه می‌توان تعداد سطرهای یک `DataGridView` را تعیین کرد؟
 با ویژگی `dgvFactors.Rows.Count` می‌توان تعداد سطرها را مشخص کرد.
 چون یک سطر خالی دارد از یک کم می‌شود.

■ عملکرد متد `Add` کنترل `dgvFactors` چیست؟ یک سطر خالی ایجاد می‌کند.
 ■ قیمت کل در کدام خانه `dgvFactors` قرار می‌گیرد و چگونه محاسبه می‌شود؟
 ستون قیمت کل حاصل ضرب تعداد کالا در قیمت کالا است که در ستون چهارم
 قرار می‌گیرد.

۱۱- کد مرحله ۱۰ را تغییر دهید تا در صورت خالی بودن کادر متن کد کالا خطا
 رخ ندهد.

قبل از اضافه کردن به سبد خرید دستورات زیر را بنویسید.

```

if (txtFGoodsCode.TextLength==0 || txtFAmount.TextLength
==0)
{
    MessageBox.Show("کد کالا یا تعداد کالا را وارد کنید");
    txtFGoodsCode.Focus();
    return;
}

```

۱۲- به متد رویداد کلیک دکمه «درج در سبد خرید» محاسبه مبلغ قابل پرداخت
 را اضافه کنید.

```

int sum = 0;
for (int i = 0; i < n+1; i++)
{
    int price=(int)dgvFactors.Rows[i].Cells[4].Value;
    sum += price;
}
lblAllCost.Text = sum.ToString();

```

کارگاه ۱۸ – کنترل خطا در برنامه پایگاه داده

هدف از این کارگاه آموزش کنترل خطا در برنامه‌های پایگاه داده با استفاده بلاک try-catch-finally است. ابتدا از هنرجو بخواهید در مورد این بلاک توضیحی ارائه کند و یک برنامه نمونه داده‌ای را وارد کند که خطای زمان اجرا رخ دهد. برای فهرست خطاهای احتمالی هر کدام یک مثال در برنامه پایگاه داده بزنید.

پاسخ به فعالیت‌ها

مرحله ۲: با درج کد تکراری چه اتفاقی رخ می‌دهد؟ خطا رخ می‌دهد.

| نام عمل | فهرست برخی از خطاهای ممکن |
|---------|---|
| درج | خطای ناشی از ساختار نادرست دستور درج - خطای ناشی از خالی ماندن فیلدی که نباید Null باشد - خطای ناشی از ورودی غیرممنوع - خطای ناشی از ورود رکورد تکراری |
| حذف | خطای ناشی از ساختار نادرست دستور حذف - خطای ناشی از حذف رکوردی در جدول پدر که در ارتباط با جدول فرزند است. |
| ویرایش | خطای ناشی از ساختار نادرست دستور ویرایش - خطای ناشی از خالی ماندن فیلدی که نباید Null باشد - خطای ناشی از ورودی غیرممنوع - خطای ناشی از ورود رکوردی که کلید اصلی تکراری دارد. |

تکمیل کارگاه
ص ۲۴۵

مرحله ۳: کالایی را ثبت کنید که کد کالای آن تکراری باشد، چه اتفاقی رخ می‌دهد؟ پیام «خطا در ثبت کالا» نمایش داده می‌شود.

مرحله ۴: آیا دستور if در برنامه عمل کنترل خطا را انجام می‌دهد؟ بله اگر کد کالا غیر عددی نوشته شود، آیا خطایی رخ می‌دهد؟ پیام «خطا در ثبت کالا» نمایش داده می‌شود.

کارگاه ۱۹ – توسعه پروژه فروشگاه

هدف از این کارگاه ذخیره برگه خرید در برنامه فروشگاه است. ابتدا از هنرجو بخواهید توضیح دهد با ذخیره برگه خرید چه اتفاقی در پایگاه داده MyShop رخ می‌دهد. سپس با نمایش فیلم «ذخیره برگه خرید» این برنامه را در کلاس تکمیل کنید.

نکات تکمیلی پروژه فروشگاه:

اگر موجودی کالایی برای درج در سبد خرید کافی نباشد برنامه به کاربر اعلام کند.

یک کنترل برچسب به نام lblStock به فرم برگه خرید اضافه کنید و هم‌زمان با مقداری برچسب نام کالا و قیمت کالا این برچسب را مقداری کنید. کد زیر را در txtFGoodsCode_KeyDown اضافه کنید.

```
lblStock.Text=
myShopDataSet.Goods.Rows[0][ "GoodsStock" ].ToString();
و کد زیر را در «درج در برگه خرید» سربرگ کالا اضافه کنید.
lblStock.Text = txtStock.Text;
و کدهای زیر را در دکمه «درج در سبد خرید» اضافه کنید.
if (txtFGoodsCode.TextLength == 0 ||
txtFAmount.TextLength==0)
{
    MessageBox.Show("کد کالا یا تعداد کالا را وارد کنید");
    txtFGoodsCode.Focus();
    return;
}
if(int.Parse(txtFAmount.Text)>int.Parse(lblStock.Text))
{
    MessageBox.Show(" موجودی کافی نیست ");
    txtFAmount.Focus();
    return;
}
}
پس از ثبت برگه خرید باید موجودی کالاهایی که خریداری شده نیز کم شود و
اطلاعات برگه خرید جاری حذف شود و شماره برگه جدید ارائه شود. متد
SetFactorNumber یک واحد به شماره برگه خرید اضافه می‌کند.
private void SetFactorNumber()
{
    factorTableAdapter.Fill(myShopDataSet.Factor);
    lblFactorNO.Text = (myShopDataSet.Factor.Rows.Count +
1).ToString();
}
```

پاسخ به فعالیت‌ها

فعالیت منزل
ص ۲۴۶

- سربرگ جدیدی به برنامه فروشگاه به نام کاربران اضافه کنید که فقط مدیر فروشگاه دسترسی به آن داشته باشد و عملیات ایجاد، حذف و ویرایش کاربر را انجام دهد.

پاسخ: فیلد TypeOfAccess در جدول Users در پایگاه داده MyShop نوع دسترسی کاربران را مشخص می‌کند. اگر کاربر مدیر فروشگاه باشد مقدار این فیلد ۱ یک در نظر گرفته می‌شود.

یک سربرگ جدید «کاربران» به نام `tabPageUsers` با طراحی زیر ایجاد کنید.

فیلد `typeOfAccess` را برای تعیین نوع کاربر در بخش کلاس فرم `LoginForm` به صورت عمومی (`Public`) اعلان کنید. کد اعتبارسنجی کاربر را در متد رویداد `btnLogin_Click` به صورت زیر تغییر دهید:

```
usersTableAdapter.FillByUsernameAndPass(myShopDataSet.Users, txtUsername.Text, txtPassword.Text);
if (myShopDataSet.Users.Rows.Count > 0)
{
    succeeded = true;
    typeOfAccess =
byte.Parse(myShopDataSet.Users.Rows[0]["TypeOfAccess"]).ToString());
    this.Hide();
}
```

کد زیر را برای نمایش یا عدم نمایش سربرگ «کاربران» در انتهای رویداد `MainForm_Load` بنویسید.

```
LoginForm frm = new LoginForm();
if (frm.typeOfAccess != 1)
tabShop.TabPages.Remove(tabPageUsers);
```

برای مدیریت پایگاه داده کتابخانه در `C#` باید عملیات جست‌وجو و درج و حذف و ویرایش برای سه جدول عضو و کتاب و امانت انجام شود. پس از ایجاد یک اتصال `Wisard` به پایگاه داده و نمایش سطرهای جدول روی `DataGridView`، با باز کردن پرونده `xsd` می‌توان متدهای جست‌وجو، درج، حذف و ویرایش را ایجاد کرد و سپس در برنامه فراخوانی کرد. برای مدیریت امانت در برنامه کتابخانه می‌توان فرمی همانند شکل زیر ایجاد کرد و عملیات ذخیره را روی جدول انجام داد.

PM 11:52:40 9/15/2018

مدیر ویرایش راهنما خروج

جستجو (Ctrl + F) اعضا (Ctrl + M) کتاب (Ctrl + B) لیست کتاب ها (Ctrl + K) لیست اعضا (Ctrl + U) امانت (Ctrl + R)

| کد کتاب | نام خانوادگی عضو | نام عضو | کد عضو |
|---------|------------------|---------|--------|
| | | | |

امانت جدید

انتخاب عضو: 1

انتخاب کتاب: 1

تاریخ امانت:

تاریخ بازگشت:

ذخیره

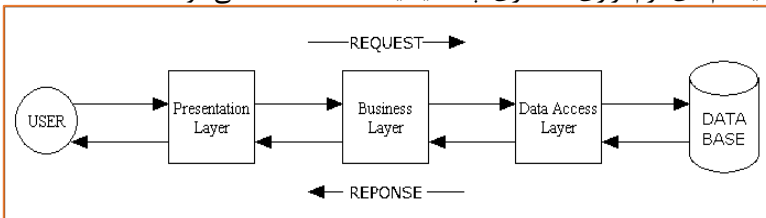
برای ذخیره امانت، کد عضویت و کد کتاب را از کنترل کادر ترکیبی متناظر انتخاب کنید و پس از وارد کردن تاریخ امانت و تاریخ بازگشت دکمه ذخیره را کلیک کنید. ذخیره امانت در واقع درج یک رکورد در جدول امانت است که می‌توانید متد آن را در پرونده xsd ایجاد کنید. سورس کد مربوط به پروژه پودمان پنجم در لوح نوری همراه کتاب موجود است.

الف) مطالب تکمیلی

قصد داریم پروژه مدیریت مشتریان (ManageCustomers) را با برنامه‌نویسی سه لایه‌ای انجام دهیم. قبل از انجام این کار مدل سه‌لایه در معماری نرم‌افزار و تکنولوژی ADO.NET شرح داده می‌شود.

مدل سه‌لایه در معماری نرم‌افزار

در مهندسی نرم‌افزار، برای کاهش پیچیدگی و ساده‌تر شدن مراحل طراحی، پیاده‌سازی و نگهداری نرم‌افزارها، سیستم‌های نرم‌افزاری را به چند زیر سیستم تقسیم کرده و قسمت‌های مختلف سیستم را به صورت لایه‌های جداگانه و مستقل از هم طراحی و پیاده‌سازی می‌کنند. این لایه‌ها با یکدیگر در ارتباط هستند و بین آن‌ها انتقال اطلاعات انجام می‌گیرد. هر کدام از این لایه‌ها وظیفه خاص خود را دارند و به لایه‌های بالایی و پایینی خود سرویس داده و از آن‌ها سرویس می‌گیرند. کاربر نهایی فقط با لایه بیرونی در ارتباط است و کاری با لایه‌های دیگر ندارد. به این روش طراحی سیستم‌های نرم‌افزاری، معماری چندلایه یا N-Tier گفته می‌شود.



معماری سه‌لایه (3-Tier) حالت خاصی از معماری چند لایه است که در آن سیستم‌ها به سه لایه جداگانه تقسیم می‌شوند. این لایه‌ها عبارتند از:

- ۱) **لایه نمایش (Presentation Layer):** این لایه که به آن لایه رابط کاربر (User Interface) نیز گفته می‌شود، عناصری مانند فرم‌ها، کنترل‌های روی فرم‌ها، منوهای برنامه و هر چیزی را دربرمی‌گیرد که کاربر نهایی از سیستم مشاهده می‌کند. کاربر سیستم به طور مستقیم فقط با این لایه در ارتباط است و در واقع درخواست خود را از طریق این لایه به لایه‌های زیرین انتقال می‌دهد. لایه نمایش اطلاعات لازم را از کاربر گرفته، در صورت لزوم برخی اعتبارسنجی‌ها (Validation) مانند کنترل طول فیلدها یا کنترل اجباری بودن برخی فیلدها را انجام می‌دهد و سپس این اطلاعات را برای هرگونه پردازش دیگری به لایه بعدی ارسال می‌کند.
- ۲) **لایه منطق تجاری (Business Logic Layer):** وظیفه اصلی این لایه که به آن لایه میانی (Middle Tier) نیز گفته می‌شود، اعمال منطق اصلی

برنامه روی درخواست کاربران و نیز برقراری ارتباط بین لایه نمایش و لایه داده است. کلیه درخواست‌هایی که در اثر تعامل کاربر با لایه نمایش ایجاد شده است به این لایه منتقل می‌شود و تمام پردازش‌های لازم بر اساس منطق اصلی برنامه در این لایه انجام می‌شود. نتیجه این پردازش به لایه نمایش منتقل شده، برای کاربر به نمایش درمی‌آید. گاهی اوقات درخواست کاربر به گونه‌ای است که لایه منطق تجاری برای انجام آن نیاز دارد که با لایه داده یعنی لایه زیرین خود نیز ارتباط داشته باشد.

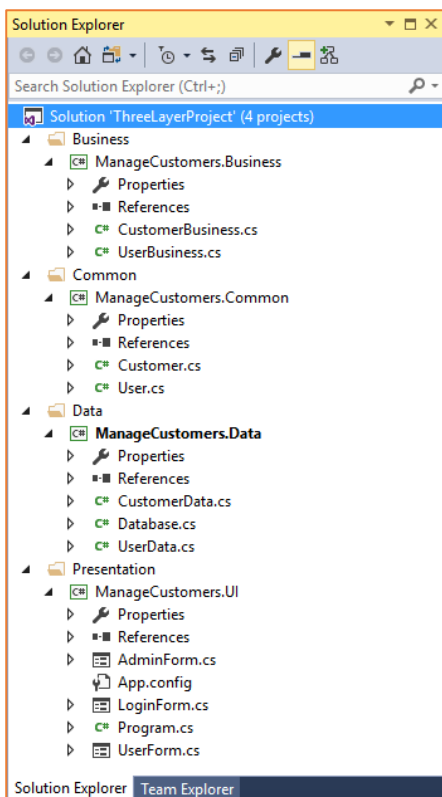
۳) **لایه دسترسی به داده (Data Access Layer):** این لایه که به آن لایه بانک اطلاعاتی نیز گفته می‌شود، وظیفه مدیریت اطلاعات موجود در بانک اطلاعاتی را بر عهده دارد. این لایه بر اساس درخواست‌هایی که از لایه بالایی خود دریافت می‌کند عملیاتی نظیر حذف، اضافه، اصلاح و خواندن اطلاعات را بر روی بانک اطلاعاتی انجام داده، نتیجه عمل را به لایه بالایی خود ارسال می‌کند. باید توجه داشت که ارتباط با بانک اطلاعاتی فقط از طریق لایه داده انجام می‌گیرد.

از مزایای معماری چندلایه می‌توان به عدم وابستگی لایه‌ها به همدیگر اشاره کرد. برای نمونه لایه داده مستقل از لایه‌های دیگر عمل کرده، در صورت لزوم می‌توان با سرعت بالا و هزینه پایین این لایه را تغییر داد. به عنوان مثال در برنامه‌ای که از بانک اطلاعاتی SQL Server استفاده می‌شود به راحتی می‌توان بانک اطلاعاتی را به Access و یا هر بانک اطلاعاتی دیگر تغییر داد و یا با کمترین هزینه می‌توان منطق اصلی برنامه را با تغییر لایه منطق تجاری تغییر داد. اما معماری سه‌لایه یا چندلایه همیشه راه‌حل مناسبی برای طراحی سیستم‌ها نیست. به عبارت دیگر برای طراحی هر سیستمی نمی‌توان از این معماری استفاده کرد.

برای درک بهتر برنامه نویسی سه‌لایه‌ای پروژه مدیریت مشتریان در لوح نوری همراه کتاب ارائه شده است. در برنامه مدیریت مشتریان کاربرانی وجود دارند که توسط مدیر ایجاد می‌شوند و هر کدام به اطلاعات برخی از مشتریان دسترسی دارند.

برای مدیریت پایگاه داده ManageCustomers در برنامه نویسی سه‌لایه به صورت زیر اقدام کنید:

پس از اجرای برنامه VS یک پروژه Blank Solution به نام ThreeLayerProject ایجاد کنید. برای تعریف DTO های کاربر و مشتری یک پوشه Common ایجاد کنید و پروژه‌ای Class Library به نام ManageCustomers.Common و کلاس‌های User و Customer را ایجاد کنید. سه پوشه برای سه‌لایه مطابق با شکل ایجاد کنید. در پوشه‌های Data و Business برای لایه داده و منطق دو پروژه Class Library و در پوشه Presentation برای لایه نمایش یک پروژه ویندوزی با سه فرم ایجاد کنید.



تعریف متدها و ویژگی‌ها و استفاده از کلاس‌ها در پروژه ویندوزی را در برنامه ManageCustomers در لوح نوری همراه کتاب ببینید.

تکنولوژی ADO.Net :

در NET Framework. دستیابی داده با استفاده از ADO.NET انجام می‌شود. ADO.NET در حقیقت یک رابط برنامه‌نویسی است که مجموعه امکانات لازم به‌منظور برقراری اتصال با پایگاه‌داده را در اختیار پیاده‌کنندگان برنامه‌های ویندوزی یا تحت وب قرار می‌دهد. ADO.NET برای دسترسی به داده‌ها از معماری غیرمتصل استفاده می‌کند. در معماری غیرمتصل ابتدا برنامه به موتور پایگاه‌داده مورد نظر متصل می‌شود و داده‌های مورد نیاز خود را از پایگاه‌داده می‌خواند و در حافظه کامپیوتر ذخیره می‌کند، سپس اتصال از پایگاه‌داده قطع می‌شود و تغییرات مورد نظر خود را در داده‌های ذخیره شده در حافظه انجام می‌دهد. هر زمان که نیاز باشد تغییرات ایجاد شده در پایگاه‌داده ذخیره شود، برنامه یک اتصال جدید را به پایگاه‌داده ایجاد کرده، از طریق این اتصال، تغییراتی را که در داده‌ها اعمال کرده بود در جدول اصلی ایجاد می‌کند. ADO.Net امکان دسترسی به منابع داده‌ای مختلف مثل پایگاه‌داده SQL Server، فایل XML، پایگاه‌های ODBC و... را فراهم می‌سازد. ADO.Net

دسترسی به داده و دستکاری داده را از هم جدا کرده است، که هر کدام را به تنهایی می‌توانید به کار برده یا با هم استفاده کنید. ADO.Net شامل فراهم‌کننده‌های داده (NET FrameWork Providers). برای اتصال به منابع داده‌ای، اجرای دستورات و بازگرداندن نتایج به درخواست‌کننده است. جدول زیر فهرستی از فراهم‌کننده‌های داده‌ای را نمایش می‌دهد.

| نام فراهم‌کننده داده‌ای | نشانه پیشوندی | توصیف Data Source | نام فضای نام مورد استفاده |
|--------------------------|---------------|---|---------------------------|
| ODBC Data Provider | Odbc | برای منابع داده‌ای به همراه یک واسط ODBC | System.Data.Odbc |
| OleDb Data Provider | OleDb | منابع داده‌ای که یک واسط OleDb مثل Access یا Excel را ارایه می‌کند. | System.Data.OleDb |
| Oracle Data Provider | Oracle | برای پایگاه‌داده Oracle | System.Data.OracleClient |
| SQL Server Data Provider | Sql | برای ارتباط با Microsoft Sql Server | System.Data.SqlClient |

فراهم‌کننده‌های داده‌ای NET. اجزایی هستند که به‌طور خاص برای دستکاری و نقل و انتقال و خواندن اطلاعات طراحی شده‌اند. اشیاء اصلی این فراهم‌کننده‌ها عبارتند از:

■ شیء Connection: شرایط و امکانات لازم برای اتصال به منبع داده‌ای را فراهم می‌کند.

■ شیء Command: این شیء امکان دسترسی به دستورات لازم برای دریافت، ویرایش، درج اطلاعات و... و همچنین اجرای توابع (Stored Procedure) SQL و ارسال پارامترها را فراهم می‌کند.

■ شیء DataReader: یک شیء کامل برای خواندن حجم گسترده‌ای از اطلاعات از منابع داده‌ای مختلف.

■ شیء DataAdapter: این شیء یک پل رابط بین شیء DataSet و منبع داده‌ای است. شیء DataAdapter از شیء Command برای اجرای دستورات مختلف SQL برای دو هدف استفاده می‌کند:

۱) دریافت اطلاعات مورد نظر از پایگاه‌داده و قراردادن آن‌ها در شیء DataSet

۲) دریافت تغییراتی که در اطلاعات شیء DataSet رخ داده و اعمال و به روزرسانی آن‌ها در منبع داده‌ای.

کلاس‌های کار با پایگاه داده Access در فضای نام System.Data.OleDb قرار دارند. نام کلاس‌ها عبارتند از:

OleDbConnection و OleDbCommand و OleDbDataAdapter و ...

کلاس‌های کار با پایگاه داده MS SQL Server در فضای نام System.Data.SqlClient قرار دارند. نام کلاس‌ها عبارتند از:

SqlConnection و SqlCommand و SqlDataAdapter و ... که شرح این کلاس‌ها در بالا آمده است.

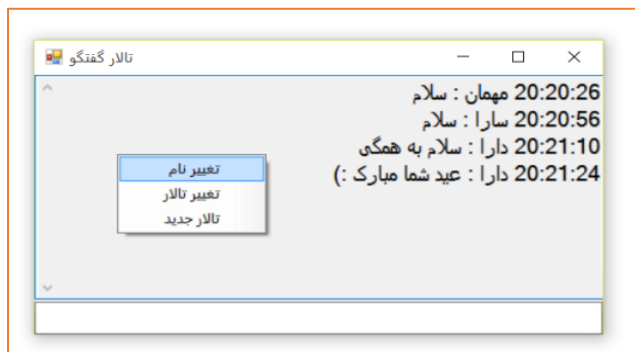
ب) فعالیتهای تکمیلی

در هر کلاس هنرجویانی هستند که به نسبت اشتیاق و علاقه بیشتری به مباحث برنامه‌نویسی دارند و ممکن است از قبل هم تا حدودی با برنامه‌نویسی آشنا بوده باشند. معمولاً روند عادی کلاس برای این هنرجویان کند و ملال آور بوده و این امر می‌تواند باعث سرخوردگی و از بین رفتن انگیزه آن‌ها شود. هنرآموزان اغلب برای حفظ روحیه و ایجاد انگیزه در این هنرجویان، در گروه‌بندی‌های کلاسی آن‌ها را به عنوان سرگروه انتخاب می‌کنند. یک راهکار مؤثر دیگر برای تشویق این هنرجویان و ایجاد انگیزه مضاعف در آن‌ها، تعریف و مشخص کردن برخی پروژه‌های ساده برای آن‌ها است. در انجام این پروژه‌ها هنرجویان دیگر هم می‌توانند به صورت گروهی مشارکت داشته باشند. این پروژه‌ها را هنرجویان در منزل انجام می‌دهند و برای رفع اشکال به هنرآموز خود مراجعه می‌کنند. هنرآموزان باید دقت داشته باشند که متناسب با میزان آمادگی هنرجویان پروژه‌ها را تعریف کنند و در نظر داشته باشند که سپردن یک پروژه سنگین که خارج از توان هنرجو است نتیجه عکس خواهد داد و موجب افت انگیزه در هنرجویان خواهد شد. هنرآموزان می‌توانند فیلم آموزشی از پیاده‌سازی این پروژه‌ها را در اختیار هنرجویان قرار دهند.

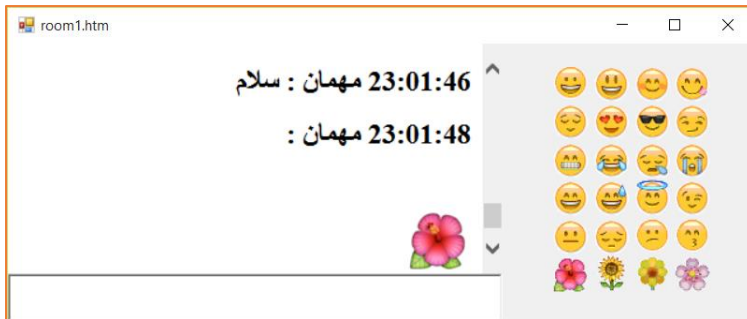
پنج پروژه زیر به عنوان فعالیتهای تکمیلی قسمت اول پودمان یعنی کار با پرونده‌ها و پوشه‌ها در نظر گرفته شده‌اند.

۱) پروژه تالار گفتگو (Chat Room)

در این پروژه هنرجویان می‌توانند با استفاده از یک کامیوننت نگهبان پرونده (fileSystemWatcher) و نوشتن پیام در یک پرونده متنی مشترک و خواندن پیام‌ها از این پرونده، یک تالار گفتگوی مجازی ایجاد کنند.



برای توسعه این پروژه می‌توان قابلیت نمایش استیکر و ایموجی (Emoji) نمادها را به برنامه افزود. برای این منظور می‌توان به جای کنترل کادر متنی از یک کنترل مرورگر وب (webBrowser) و به جای پرونده متنی ساده از پرونده html برای نمایش پیامها استفاده کرد. به این ترتیب کافی است که پیامها را در قالب کدهای html در پرونده متنی مشترک با پسوند htm نوشته و هر بار که محتوای این پرونده تغییر کرد، پرونده را در مرورگر وب بارگذاری کرد.



۲) پروژه شبیه‌ساز سیستم خودپرداز بانکی (ATM Simulator)

در این پروژه هنرجویان یک برنامه شبیه‌ساز سیستم خودپرداز بانکی می‌نویسند. این خودپرداز ۵ صفحه مختلف دارد. هنرجویان می‌توانند برای طراحی صفحات خودپرداز از نرم‌افزار powerpoint استفاده کنند. هر صفحه در قالب یک اسلاید طراحی شده، در انتها اسلایدها به صورت پرونده تصویری ذخیره می‌شوند.

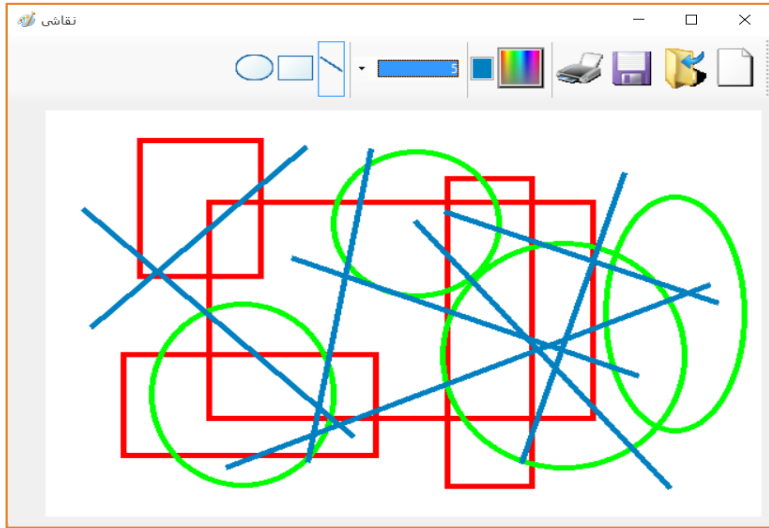


هنرجویان پس از طراحی رابط کاربری در ادامه برای رخدادهای شبیه‌ساز کدنویسی می‌کنند. بر اساس فعالیت‌های صورت‌گرفته توسط کاربر، صفحات خودپرداز تغییر می‌کنند. برای تغییر صفحات، تصویر صفحه جدید جایگزین تصویر قبلی در کادر تصویر می‌شود. از یک کادر متنی که روی تصویر قرار داده شده، برای گرفتن گذرواژه کاربر استفاده شده است.

برای توسعه این پروژه می‌توان صفحات جدیدی به آن افزود. مثلاً صفحه پرداخت قبض و صفحه دریافت یا انتقال وجه. همچنین می‌توان به‌جای استفاده از کنترل کادر تصویر، از یک کنترل مرورگر وب (webBrowser) برای نمایش صفحات خودپرداز استفاده کرد. در این حالت هر صفحه به صورت یک پرونده html طراحی می‌شود. همچنین می‌توان مدت زمان توقف در هر صفحه را با استفاده از تایمر محدود کرد.

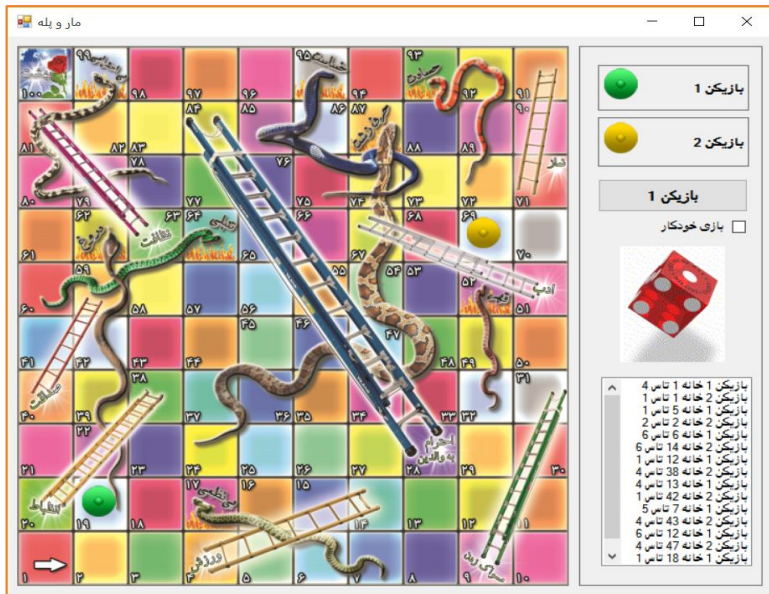
۳) پروژه برنامه نقاشی (Paint)

در این پروژه هنرجویان برنامه‌ای نظیر برنامه نقاشی ویندوز ایجاد خواهند کرد. کنترل‌های استفاده شده در این برنامه همگی برای هنرجو آشنا هستند و تنها موارد جدید در این برنامه استفاده از کلاس‌های Graphics و Bitmap برای ایجاد اشکال گرافیکی خط، دایره و مستطیل است. این برنامه تمرینی برای کار با پرونده‌های تصویری در برنامه است.



۴) پروژه بازی مارپله (Snake & Ladder)

بازی‌ها همیشه جذابیت خود را دارند و نوشتن یک بازی جذابتی مضاعف برای هنرجویان خواهد داشت. این پروژه فرصتی برای هنرجویان است که از آموخته‌های خود برای نوشتن یک بازی استفاده کنند.



۵) پروژه ماشین حساب (Calculator)



در این پروژه هنرجویان یک ماشین حساب ساده ایجاد خواهند کرد.

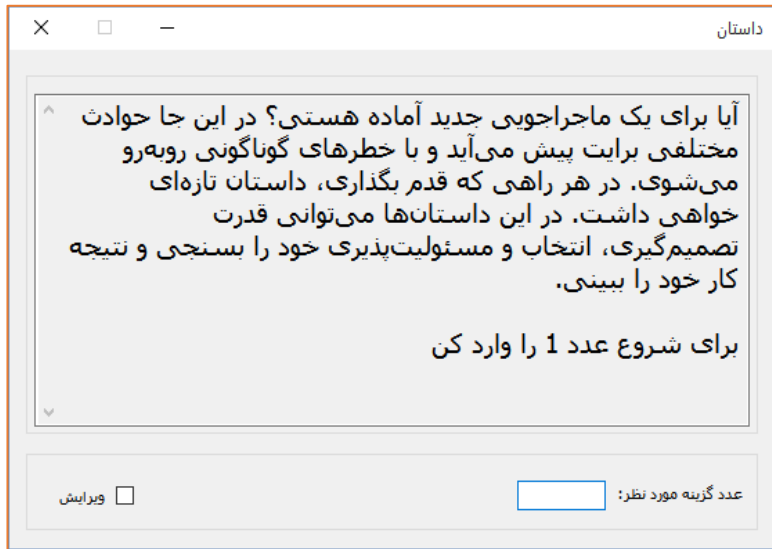
پروژه برای قسمت ارتباط با پایگاه داده:

برنامه انتخابات شورای دانش‌آموزی را بنویسید. معاون پرورشی مدرسه داده‌های مربوط به دانش‌آموزان و نامزدها را وارد می‌کند. هر دانش‌آموز با نام کاربری و گذرواژه وارد فرم اخذ رأی می‌شود و به نامزدهای مورد علاقه خود رأی می‌دهد، سپس حق رأی از او گرفته می‌شود. در پایان انتخابات معاون پرورشی می‌تواند نتایج انتخابات را به ترتیب تعداد رأی مشاهده کند. برنامه این پروژه به پیوست در لوح نوری همراه کتاب است.

ج) نمونه ارزشیابی پایانی

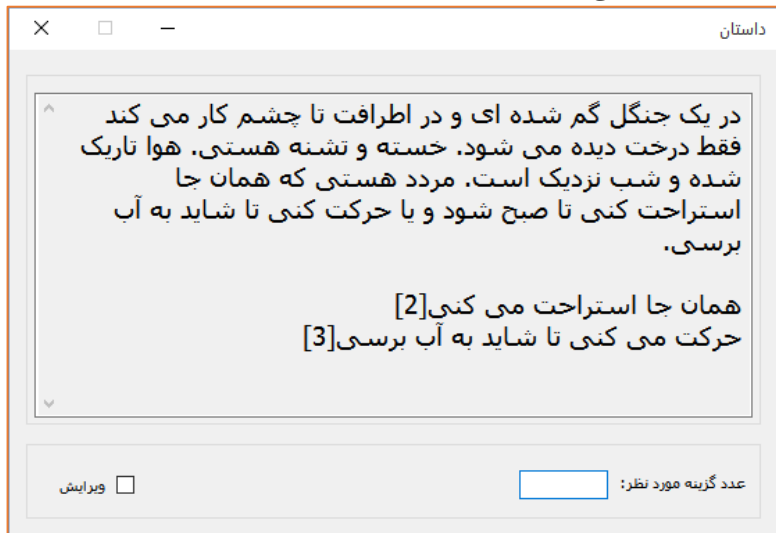
قسمت اول: کار با پرونده‌ها و پوشه‌ها

می‌خواهیم برنامه‌ای برای خواندن و دنبال کردن یک داستان متنی پرماجرا بنویسیم. قهرمان این داستان در موقعیت‌های مختلفی قرار می‌گیرد و با توجه به تصمیم‌هایی که می‌گیرد و راهی که برای نجات خود انتخاب می‌کند، حوادث مختلفی برایش پیش می‌آید. هر بخش از این داستان در یک پرونده متنی جداگانه قرار دارد. نام پرونده‌ها به صورت عددی است. مانند 0.txt, 1.txt, 2.txt, ...



متن پرونده 0.txt

در هر پرونده متنی، بخشی از داستان روایت می‌شود و در انتهای هر بخش دو راه پیش پای خواننده قرار می‌گیرد. خواننده با وارد کردن شماره پرونده، مسیر بعدی داستان را انتخاب می‌کند.



متن پرونده 1.txt

مراحل انجام کار:

(۱) یک پروژه از نوع WindowsForm ایجاد کنید. نام پروژه را TextGame بگذارید.

(۲) ابتدا چیدمان فرم را برای زبان فارسی تنظیم کنید، سپس فرم پروژه را مطابق شکل طراحی کنید.

در این فرم از دو کنترل Group Box، یک کنترل Label، دو کنترل Text Box و یک کنترل Check Box استفاده شده. نام کنترل Check Box را chkEdit و بگذارید. ویژگی کنترل کادر متنی بزرگتر را مطابق جدول زیر تنظیم کنید.

| نام ویژگی | مقدار | توضیح |
|------------|-----------------|-----------|
| Name | txtStory | نام کنترل |
| Font | Tahoma, 14.25pt | |
| Multiline | True | |
| ReadOnly | True | |
| ScrollBars | Vertical | |
| TabIndex | 1 | |

ویژگی کنترل کادر متنی کوچکتر را مطابق جدول زیر تنظیم کنید.

| نام ویژگی | مقدار | توضیح |
|-----------|-----------|-----------|
| Name | txtSelect | نام کنترل |
| TabIndex | 0 | |

(۱) در پوشه Debug برنامه یک پوشه به نام story ایجاد کنید. در داخل این پوشه شش پرونده متنی به نام‌های 0.txt, 1.txt, 2.txt, ..., 5.txt ایجاد کنید و در هر کدام از این پرونده‌ها متن کوتاهی بنویسید.

(۲) در کلاس فرم خود یک متغیر رشته‌ای به نام fileName برای ذخیره مسیر پرونده جاری تعریف کنید و مقدار آن را برابر با "story\0.txt" قرار دهید. در رخداد FormLoad پنجره اصلی، محتویات پرونده fileName را بخوانید و در کادر متنی txtStory قرار دهید.

(۳) در رخداد KeyDown کادر متنی txtSelect محتوای پرونده انتخاب شده را نمایش دهید. برای این منظور کد دستورات زیر را بنویسید.

- بررسی کنید که اگر کلید **Enter** زده شده بود و کادر متنی `txtSelect` خالی نبود، کارهای زیر انجام شود.
- با استفاده از تابع `string.Format`، نام پرونده متنی بعدی را از کادر متنی `txtSelect` بگیرید و به صورت قالب‌بندی‌شده در متغیر `filename` قرار دهید.
 - اگر پرونده `filename` وجود داشت:
 - محتویات پرونده `filename` را بخوانید و در کادر متنی `txtStory` قرار دهید.
 - محتوای کادر متنی `txtSelect` را در عنوان فرم نمایش دهید.
 - محتوای کادر متنی `txtSelect` را پاک کنید.
- در غیر اینصورت:
- در کادر متن `txtStory` متن زیر را قرار دهید:
 - "این قسمت از داستان هنوز نوشته نشده است و شما می‌توانید ادامه داستان را خودتان بنویسید."

- (۱) در رخداد `CheckedChanged` کنترل `chkEdit` حالت فقط خواندنی کادر متنی را تغییر دهید و محتویات کادر متنی `txtStory` را در پرونده `filename` ذخیره کنید. برای این منظور کد دستورات زیر را بنویسید.
- حالت فقط خواندنی کادر متنی `txtStory` را برابر با نقیض `chkEdit.Checked` قرار دهید.
 - محتویات کادر متنی `txtStory` را در پرونده `filename` ذخیره کنید.
 - محتوای کادر متنی `txtSelect` را در عنوان فرم نمایش دهید.
 - محتوای کادر متنی `txtSelect` را پاک کنید.
- (۲) برنامه را اجرا کنید و یک ماجرای داستانی را با آن روایت کنید.

قسمت دوم: ارتباط با پایگاه داده

می‌خواهیم سامانه‌ای برای ایجاد یک سرویس ثبت آگهی خرید و فروش املاک، طراحی و پیاده‌سازی کنیم. در این سامانه هر کاربر با نام کاربری و گذرواژه خود وارد برنامه می‌شود و می‌تواند آگهی‌های ثبت شده توسط دیگران را مشاهده کرده و آگهی‌های خود را ثبت و ویرایش کند.

طراحی و پیاده‌سازی پایگاه‌داده این سامانه قبلاً در اکسس انجام شده است و این پایگاه‌داده با نام `advert.mdb` در اختیار شما قرار داده می‌شود. این پایگاه‌داده شامل

دو جدول به نام‌های جدول Users و جدول Adverts است. در جدول Users اطلاعات کاربران سیستم و در جدول Adverts آگهی‌های کاربران ثبت می‌شود.

| Field Name | Data Type | Description (Optional) |
|--------------|------------|------------------------|
| UserID | Short Text | کد مشتری |
| UserPassword | Short Text | گذرواژه |
| UserName | Short Text | نام |
| UserLastName | Short Text | نام خانوادگی |
| Email | Short Text | پست الکترونیک |
| Phone | Short Text | شماره تلفن |
| Address | Short Text | آدرس پستی |

جدول Users

| Field Name | Data Type | Description (Optional) |
|--------------|------------|------------------------|
| AdvertID | AutoNumber | کد آگهی |
| UserID | Short Text | شناسه کاربر |
| BuildingType | Short Text | نوع ملک |
| Area | Number | متراف |
| Address | Short Text | محل ملک |
| Bedrooms | Number | تعداد اتاق خواب |
| Price | Currency | قیمت |
| RecordDate | Date/Time | تاریخ ثبت آگهی |
| Description | Short Text | توضیحات |

جدول Adverts

در ادامه یکی از بخش‌های "الف" یا "ب" را انتخاب کرده و به سوالات آن پاسخ دهید.
بخش الف: پیاده‌سازی واحد ثبت نام، احراز هویت و ویرایش اطلاعات کاربر

(۱) پروژه جدیدی با نام Avderts در C# ایجاد کنید و نام فرم اصلی برنامه را MainForm بگذارید. دو فرم به نام‌های RegisterForm و LoginForm به برنامه اضافه کنید.

(۲) با اجرای برنامه، پنجره LoginForm نمایش داده می‌شود و از کاربر

خواسته می‌شود که نام کاربری و گذرواژه خود را وارد کند. این فرم را مطابق شکل طراحی کنید.

با زدن دکمه ورود، در صورتی که نام کاربری و گذرواژه معتبر باشد، پنجره LoginForm بسته شده و پنجره MainForm برای کاربر نمایش داده می‌شود. در غیر اینصورت پیام خطایی به شکل زیر ظاهر می‌شود.

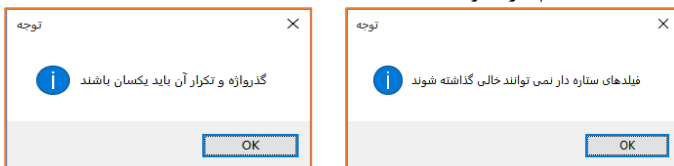
- پایگاه داده advert.mdb را به پروژه اضافه کرده و متد لازم برای احراز هویت کاربر را به Table Adapter جدول Users اضافه کنید. سپس با توجه به توضیحات داده شده کد رخداد کلیک دکمه "ورود" را بنویسید.

(۱) در پنجره MainForm و در صفحه "حساب کاربری من" اطلاعات کاربر باید نمایش داده شود.

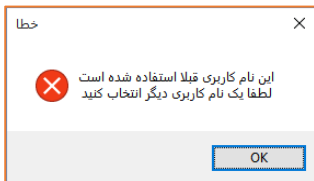
برای این منظور نام کاربری را از صفحه LoginForm دریافت کرده و در یک متغیر به نام userId در کلاس MainForm ذخیره کنید. سپس با فراخوانی متد لازم از Table Adapter جدول Users اطلاعات کاربر را به او نشان دهید.

(۲) در پنجره ورود با زدن دکمه "قبلا ثبت نام نکرده اید؟" پنجره ثبت نام برای کاربر نمایش داده می شود (RegisterForm). در این پنجره اطلاعات کاربر جدید وارد می شود.

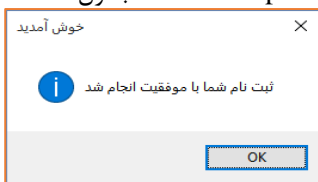
(۳) با زدن دکمه ثبت نام، ابتدا بررسی می‌شود که فیلدهای ستاره‌دار خالی نباشند و گذرواژه و تکرار آن با هم برابر باشند. در صورت خالی بودن هر یک از فیلدهای ستاره‌دار و یا برابر نبودن گذرواژه و تکرار آن پیام مناسب برای کاربر نمایش داده شود. با توجه به این توضیحات کد رخداد کلیک دکمه "ثبت نام" را بنویسید.



(۴) در صورتی که مقادیر فیلدها به درستی وارد شده بود، باید بررسی شود که نام کاربری وارد شده در جدول Users موجود نباشد. برای این منظور ابتدا متد لازم را به Table Adapter جدول Users اضافه کنید و سپس این متد را در برنامه فراخوانی کنید.



(۵) اگر نام کاربری معتبر بود آنگاه اطلاعات کاربر جدید به جدول Users اضافه شده و پیام خوش آمدید به کاربر نشان داده می‌شود. برای این منظور ابتدا متد لازم برای افزودن کاربر جدید را به Table Adapter جدول Users اضافه کنید، سپس این متد را در برنامه فراخوانی کنید. در صورت بروز استثناء در هنگام ورود رکورد جدید، پیام مناسبی باید به کاربر نشان داده شود.



بخش ب: پیاده‌سازی واحد نمایش، ثبت و ویرایش آگهی

(۱) پروژه جدیدی با نام Avderts در C# ایجاد کنید و نام فرم اصلی برنامه را MainForm بگذارید. یک TabControl با سه صفحه به برنامه اضافه کنید. صفحه فهرست آگهی‌ها را مطابق شکل زیر طراحی کنید.

آگهی املاک

فهرست آگهی های من حساب کاربری من

جستجو

آدرس

تعداد خواب

| نوع ملک | متر از | آدرس | تعداد خواب | قیمت | تاریخ ثبت آگهی | توضیحات |
|----------|--------|--------|------------|------------|----------------|---------------------|
| آپارتمان | 90 | شهدا | 2 | 3000000000 | 20/06/1397 | پارکینگ و انباری |
| آپارتمان | 100 | انقلاب | 5 | 4000000000 | 20/06/1397 | بدون پارکینگ |
| تجاری | 80 | آزادی | 2 | 3000000000 | 17/08/1396 | پارکینگ بدون انباری |
| ویلايي | 300 | رسالت | 3 | 9000000000 | 17/08/1396 | پارکینگ و انباری |
| آپارتمان | 120 | کارنگر | 3 | 5000000000 | 20/06/1397 | پارکینگ و انباری |
| آپارتمان | 120 | کارنگر | 3 | 5000000000 | 21/06/1397 | پارکینگ و انباری |
| آپارتمان | 90 | شهدا | 2 | 3000000000 | 21/06/1397 | پارکینگ و انباری |
| آپارتمان | 90 | شهدا | 2 | 3000000000 | 21/06/1397 | پارکینگ و انباری |
| آپارتمان | 90 | شهدا | 2 | 3000000000 | 21/06/1397 | پارکینگ و انباری |
| آپارتمان | 90 | شهدا | 2 | 3000000000 | 21/06/1397 | پارکینگ و انباری |
| ویلايي | 300 | رسالت | 3 | 9000000000 | 21/06/1397 | پارکینگ و انباری |
| ویلايي | 300 | رسالت | 3 | 9000000000 | 21/06/1397 | پارکینگ و انباری |
| تجاری | 80 | آزادی | 2 | 3000000000 | 21/06/1397 | پارکینگ بدون انباری |
| تجاری | 80 | آزادی | 2 | 3000000000 | 21/06/1397 | پارکینگ بدون انباری |
| تعمیر | 90 | آزادی | 2 | 3000000000 | 21/06/1397 | پارکینگ بدون انباری |

↓ ↑ ↶ ↷

در این صفحه رکوردهای جدول Adverts به صورت جدول بندی شده در یک کنترل data grid view نمایش داده می شوند. عناوین تمامی فیلدها را به فارسی نمایش دهید. چهار کنترل button با عنوان های بعدی، قبلی، ابتدا و انتها برای پیمایش رکوردها روی فرم قرار دهید و کدهای مناسب را برای رخداد کلیک آنها بنویسید.

۲) امکان جست و جوی آگهی بر اساس آدرس و یا تعداد اتاق خواب را به برنامه اضافه کنید. با زدن دکمه جستجو، رکوردهایی که شرایط مشخص شده را دارند فیلتر می شوند.

۳) امکان جست و جوی بر اساس آدرس را به گونه ای انجام دهید که با وارد کردن حروف آدرس، همه آدرس های مشابه آن فیلتر شود.

۴) صفحه فهرست آگهی ها را مطابق شکل زیر طراحی کنید. فهرست باز شونده را به فیلد نام کاربری از جدول Users مقید کنید. با انتخاب نام کاربری از فهرست باز شونده، فهرست تمام آگهی های آن کاربر در جدول نمایش داده می شود.

آگهی املاک

فهرست آگهی های من حساب کاربری من

انتخاب نام کاربری:

فهرست آگهی ها

| حذف | توضیحات | تاریخ ثبت آگهی | قیمت | تعداد خواب | آدرس | مترای | نوع ملک |
|-----|------------------|----------------|------------|------------|--------|-------|----------|
| حذف | پارکینگ و انباری | 20/06/1397 | 3000000000 | 2 | شهدا | 90 | آپارتمان |
| حذف | بدون پارکینگ | 20/06/1397 | 4000000000 | 5 | انقلاب | 100 | آپارتمان |
| حذف | پارکینگ و انباری | 20/06/1397 | 5000000000 | 3 | کارنگر | 120 | آپارتمان |
| حذف | پارکینگ و انباری | 21/06/1397 | 5000000000 | 3 | کارنگر | 120 | آپارتمان |
| حذف | پارکینگ و انباری | 21/06/1397 | 3000000000 | 2 | شهدا | 90 | آپارتمان |
| حذف | پارکینگ و انباری | 21/06/1397 | 3000000000 | 2 | شهدا | 90 | آپارتمان |

جزئیات آگهی

کد آگهی:

نوع ملک:

مترای:

آدرس:

تعداد خواب:

قیمت:

توضیحات:

ثبت آگهی جدید

ویرایش آگهی

- ۵) یک ستون برای حذف سطر به جدول اضافه کنید و کد رخداد کلیک بر روی آن را بنویسید.
- ۶) با انتخاب هر سطر از جدول، داده‌های آن سطر در کادرهای متنی بخش "جزئیات آگهی" نشان داده می‌شود. با زدن دکمه ویرایش آگهی، رکورد مشخص شده با کادر متنی "کد آگهی" باید با داده‌های کادرهای متنی به روزرسانی شود.
- ۷) با زدن دکمه ثبت آگهی جدید، یک رکورد جدید با داده‌های کادرهای متنی به جدول آگهی‌ها افزوده می‌شود.

فهرست واریسی راهنمای تصحیح آزمون

| مرحله نمره | مرحله ۱ (نوشتن در پرونده متنی) | مرحله ۲ (خواندن از پرونده متنی) | مرحله ۳ (مقیدسازی واسط گرافیکی کاربری) | مرحله ۴ (انجام عملیات روی پایگاه داده) |
|--|--|--|---|--|
| ۱ | <input type="checkbox"/> ایجاد پرونده متنی در مسیر جاری <input type="checkbox"/> | خواندن محتویات پرونده متنی <input type="checkbox"/> دسترسی به خطوط پرونده متنی <input type="checkbox"/> رفع خطای برنامه <input type="checkbox"/> | طراحی واسط کاربری و ایجاد ارتباط واسط کاربری با پایگاه داده <input type="checkbox"/> | نوشتن کد برای درج رکورد <input type="checkbox"/> |
| ۲ | ایجاد پرونده متنی جدید در مسیر تعیین شده <input type="checkbox"/> افزودن داده به پرونده متنی موجود <input type="checkbox"/> | خواندن محتویات پرونده متنی <input type="checkbox"/> دسترسی به خطوط پرونده متنی <input type="checkbox"/> | طراحی واسط کاربری و ایجاد ارتباط واسط کاربری با پایگاه داده <input type="checkbox"/> مقیدسازی کنترل <input type="checkbox"/> پیمایش رکوردها <input type="checkbox"/> | نوشتن کد برای درج رکورد، ویرایش و حذف و جست و جوی رکورد <input type="checkbox"/> |
| ۳ | ایجاد پرونده متنی جدید در مسیر تعیین شده <input type="checkbox"/> افزودن داده به پرونده متنی موجود <input type="checkbox"/> رفع خطای برنامه <input type="checkbox"/> | خواندن محتویات پرونده متنی <input type="checkbox"/> دسترسی به خطوط پرونده متنی <input type="checkbox"/> رفع خطای برنامه <input type="checkbox"/> | طراحی واسط کاربری و ایجاد ارتباط واسط کاربری با پایگاه داده <input type="checkbox"/> مقیدسازی کنترل <input type="checkbox"/> پیمایش رکوردها <input type="checkbox"/> رفع خطا <input type="checkbox"/> | نوشتن کد برای درج رکورد، ویرایش و حذف و جست و جوی رکورد <input type="checkbox"/> مدیریت و رفع خطای برنامه <input type="checkbox"/> |
| حداقل نمره قبولی از ۳ | ۲ | ۱ | ۱ | ۲ |
| نمره هنرجو | | | | |
| نمره نهایی آزمون ارزشیابی پایانی واحد یادگیری شماره ۹ - پودمان پنجم کتاب توسعه برنامه سازی و پایگاه داده | | | | |

منابع

- (۱) برنامه درسی رشته شبکه و نرم‌افزار رایانه. دفتر تالیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش (۱۳۹۴)
- (۲) استاندارد ارزشیابی حرفه فناوری اطلاعات و ارتباطات. دفتر تالیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش (۱۳۹۳)
- (۳) استاندارد شایستگی حرفه فناوری اطلاعات و ارتباطات. دفتر تالیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش (۱۳۹۲)
- (۴) شیوه‌نامه نحوه ارزشیابی دروس شایستگی‌های فنی و غیرفنی شاخه‌های فنی و حرفه‌ای و کاردانش شماره ۴۰۰/۲۱۱۴۸۲ مورخ ۹۵/۱۱/۳۰
- (۵) کربلایی، مجید. (۱۳۹۵). برنامه‌سازی ۳ کد ۳-۴۵۱/۵. تهران: شرکت چاپ و نشر کتاب‌های درسی ایران
- (۶) جعفرنژاد قمی، عین‌الله. (۱۳۹۳). آموزش گام به گام برنامه‌سازی به زبان C#، بابل: انتشارات علوم رایانه
- (۷) جعفری امین. (۱۳۹۲). آموزش گام به گام C#.NET 2010. تهران: انتشارات صفار
- (۸) احمدزاده، اسلام. کتاب الکترونیکی آموزش جامع برنامه‌نویسی C #
- (۹) آریانیان، احسان. احمدی، محمدرضا. ملکی، داوود. (۱۳۹۲). اصول مجازی‌سازی و رایانش ابری، تهران: انتشارات نیاز دانش
- 10) Agarwal, V. V. (2012). Beginning C# 5.0 databases. 2nd ed. (NY): Apress.
- 11) Stellman, A., & Greene, J. (2010). Head First C#. 2nd ed. (NY): O'Reilly.
- 12) Ray, E. (2003). Learning XML. Beijing Cambridge, Mass: O'Reilly
- 13) C# Programming Guide. <https://msdn.microsoft.com/>
- 14) All C# Language Topics. <https://stackoverflow.com/documentation/c/topics>
- 15) /c/topics