



پودمان پنجم

مدیریت پایگاه داده

مدیریت پرونده و پوشه‌های موجود در سیستم‌عامل به وسیلهٔ زبان برنامه‌نویسی موجب کارایی نرم‌افزارهای تولیدشده و استفاده از امکانات سیستم‌عامل در محیط برنامه‌نویسی می‌شود. در زبان برنامه‌نویسی C# کلاس‌های آماده‌ای وجود دارد که قابلیت دسترسی به پرونده، پوشه و دیسک‌گردان‌های دیسک سخت در رایانه را برای کاربر فراهم می‌کند. در محیط برنامه‌نویسی C# امکان اتصال برنامه‌های تولیدشده به بانک اطلاعاتی به وسیله کنترل‌های پیشرفته وجود دارد تا بتوان محتویات پایگاه داده را به وسیله برنامه مدیریت کرد. در این پودمان مفهوم و کاربرد انواع پرونده متنی و غیرمتنی و پوشه و متدهای ایجاد و دسترسی به پرونده‌ها و نحوه اتصال برنامه به پایگاه داده به وسیلهٔ واسط‌های برنامه‌نویسی را فراخواهید گرفت.

واحد یادگیری ۹

■ شایستگی مدیریت پایگاه داده

آیا تا به حال پی برده اید

- چگونه می‌توان اطلاعات را در یک پرونده ذخیره کرد؟
- اطلاعات یک پرونده متنی را چگونه می‌توان مشاهده و ویرایش کرد؟
- پرونده‌های تصویری چه تفاوتی با پرونده‌های متنی دارند؟
- چه نرم‌افزارهایی از پایگاه داده استفاده می‌کنند؟
- چگونه می‌توان از طریق کنترل‌ها به رکوردهای جدول پایگاه داده دسترسی داشت؟
- چگونه می‌توان قوانین یکپارچگی پایگاه داده را در برنامه مدیریت و کنترل کرد؟

هدف از این واحد شایستگی چگونه کار با پرونده‌ها و پایگاه داده در برنامه‌نویسی است.

استاندارد عملکرد

با استفاده از دانش کلاس‌های آماده پرونده و پوشه و شناخت دستورات کار با پایگاه داده، از پرونده‌های متنی و پایگاه داده در برنامه استفاده کند.

پرونده و انواع آن

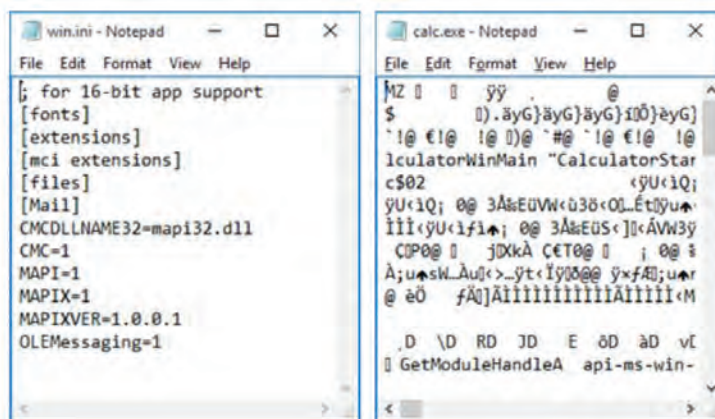
در هنگام کار با یک نرم‌افزار اداری، رایانه شما به‌طور اتفاقی خاموش می‌شود. اگر آخرین تغییرات برنامه را در یک پرونده ذخیره نکرده باشید، برای داده‌ها و اطلاعاتی که ذخیره نکرده‌اید، چه اتفاقی می‌افتد؟ متغیرها و آرایه‌هایی که برای نگهداری داده‌ها در برنامه تعریف شده‌اند، در حافظه اصلی سیستم ذخیره می‌شوند. این داده‌ها تا زمانی قابل استفاده هستند که برنامه در حال اجرا است و پس از خاتمه برنامه دیگر وجود نخواهند داشت. حافظه اصلی سیستم یک حافظه موقت است و داده‌های آن با قطع برق پاک می‌شود. برای حفظ و نگهداری داده‌های برنامه باید به روشی آنها را به‌صورت دائمی ذخیره کرد. استفاده از پرونده‌ها یک روش متداول برای ذخیره داده‌ها روی حافظه جانبی است.

پرونده دنباله‌ای از بایت‌ها است که روی حافظه جانبی تحت یک نام نگهداری می‌شود. برای دسترسی به پرونده‌ها از نام و نشانی آنها در حافظه جانبی استفاده می‌شود. برای ذخیره داده‌های مختلف، از انواع پرونده‌های متفاوتی استفاده می‌شود. سیستم عامل ویندوز نوع پرونده را از روی پسوند آن تشخیص می‌دهد. از نظر محتوا، پرونده‌ها به دو دسته تقسیم می‌شوند.

■ **پرونده‌های متنی (Text File):** پرونده‌ای که محتوای آن نویسه‌های چاپ‌شدنی است، **پرونده متنی** نامیده می‌شود.






■ **پرونده‌های دودویی (Binary File):** اگر داده‌های موجود در پرونده مانند اعداد و رشته‌ها به همان شکلی که در حافظه قرار دارند، در پرونده ذخیره شوند، پرونده را **دودویی** می‌گوییم.

محتوای پرونده‌های متنی با یک ویرایشگر ساده مانند Notepad و یا یک واژه‌پرداز مانند Word قابل مشاهده است. در حالی که محتوای پرونده‌های دودویی را نمی‌توان با برنامه‌های ویرایشگر به‌درستی مشاهده کرد (شکل ۱).



شکل ۱- محتوای یک پرونده دودویی (الف) و یک پرونده متنی (ب) در برنامه Notepad

در جدول زیر با کمک هم‌گروهی 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
.....	متنی

فعالیت گروهی



نوشتن در پرونده متنی

در کتابخانه NET، فضای نام System.IO دارای کلاس‌های مختلفی برای کار با پرونده‌ها و پوشه‌ها است. یکی از این کلاس‌ها، کلاس File است. با کمک متدهای موجود در کلاس File، می‌توان عملیات مختلفی نظیر ایجاد پرونده و نوشتن داده در آن، خواندن داده از پرونده، حذف پرونده و برخی کارهای دیگر را انجام داد. همان‌طور که شما برای نوشتن مطلبی در دفتر خود، ابتدا دفتر را باز کرده، یک صفحه سفید آن را می‌آورید، مطلب را نوشته، در پایان کار دفتر را می‌بندید، برای ساختن پرونده اطلاعاتی در رایانه نیز باید همین سه عمل را انجام دهید:

۱ باز کردن پرونده به منظور نوشتن اطلاعات

۲ نوشتن داده‌ها و اطلاعات درون پرونده باز شده

۳ بستن پرونده پس از اتمام کار

هر سه عملیات می‌تواند با متدهای مختلفی انجام شود. همچنین متدهایی وجود دارند که هر سه کار را پشت‌سرهم با یک دستور انجام می‌دهند.

کارگاه ۱ ایجاد پرونده و نوشتن در آن

می‌خواهیم برنامه‌ای بنویسیم که مشخصات هنرجو را از طریق کادرمتن دریافت کند و در یک پرونده متنی ذخیره کند.

در فضای نام System.IO متد WriteAllText از کلاس File، می‌تواند پرونده جدیدی را ایجاد کند و اطلاعات را درون آن ذخیره کرده، سپس پرونده را ببندد. اگر از قبل پرونده‌ای به همان نام وجود داشته باشد، محتوای قبلی پرونده پاک شده، متن جدید جایگزین می‌شود.

شکل کلی متد WriteAllText

System.IO.File.WriteAllText (عبارت رشته‌ای، نام و مسیر پرونده)

ورودی اول نام پرونده به همراه مسیر و ورودی دوم متنی است که در پرونده نوشته می‌شود.

۱ پروژه ویندوزی به نام SaveStudentInfo ایجاد کنید.

شکل ۲- نمونه برگ دریافت اطلاعات هنرجو

۲ فرم مناسب را طراحی کنید.

چهار کنترل کادرمتن را با نام‌های txtFirstName و txtLastName و txtFatherName و txtNationalCode و دو کنترل دکمه «ثبت» و «خروج» با نام‌های btnSave و btnExit به فرم اضافه کنید (شکل ۲).
 ۲ در متد رویداد کلیک دکمه «ثبت» کدهای زیر را بنویسید.

```
private void btnSave_Click(object sender, EventArgs e)
```

```
{
    string data = txtFirstName.Text;
    System.IO.File.WriteAllText("Info.txt", data);
}
```

برنامه را اجرا کنید، نام «علی» را در کادرمتن txtFirstName وارد کرده، دکمه «ثبت» را کلیک کنید.

۲ پرونده Info.txt را باز کنید.

پوشه Debug پروژه را باز کنید. آیا پرونده متنی Info.txt در آن پوشه قرار دارد؟ پرونده Info.txt را باز کنید. محتوای این پرونده متنی چیست؟

۵ بعد از اجرای مجدد برنامه، محتوای پرونده متنی Info.txt را بررسی کنید.

برنامه را اجرا کنید، نام «حسن» را در کادرمتن txtFirstName وارد کرده، دکمه «ثبت» را کلیک کنید. پرونده Info.txt را باز کنید. آیا نام «علی» در پرونده Info.txt وجود دارد؟

۶ دستور نوشتن در پرونده را به صورت زیر تغییر دهید.

```
System.IO.File.AppendAllText("Info.txt", data);
```

برنامه را اجرا کنید. نام‌های «حسن» و «علی» را وارد کنید. پرونده Info.txt را باز کنید. آیا هر دو نام در پرونده Info.txt ذخیره شده است؟

متدهای WriteAllText و AppendAllText هر دو رشته data را در پرونده می‌نویسند. این دو متد چه تفاوتی با هم دارند؟

۷ فضای نام System.IO را به برنامه معرفی کنید.

برای اختصار در کدنویسی و حذف کردن «فضای نام» از ابتدای کلاس‌ها، «فضای نام» را در ابتدای برنامه با عبارت using معرفی می‌کنیم. در ابتدای پرونده Form1.cs دستور زیر را اضافه کنید.

```
using System.IO;
```

۸ کدهای برنامه را به صورت زیر تغییر دهید.

```
string data = String.Format("{0}\t {1} \r\n", txtFirstName.Text, txtLastName.Text);
File.AppendAllText("Info.txt", data);
```

پرونده Info.txt را حذف کنید. چند نام دلخواه وارد و ثبت کنید. چه داده‌هایی در پرونده ذخیره می‌شوند؟ دنباله "\r\n" به عنوان نویسه خط جدید (New Line) در پرونده متنی عمل می‌کند. این رشته خط جدیدی به پرونده اضافه می‌کند و اطلاعات بعدی کاربر در ابتدای خط جدید نوشته می‌شود.

۹ برنامه را طوری تغییر دهید که نام پدر و کد ملی نیز در پرونده ذخیره شود.

کنجکاو



اگر بخواهیم اطلاعات در یک مسیر مشخص ذخیره شود، متدهای نوشتن در پرونده را چگونه فراخوانی می‌کنید؟

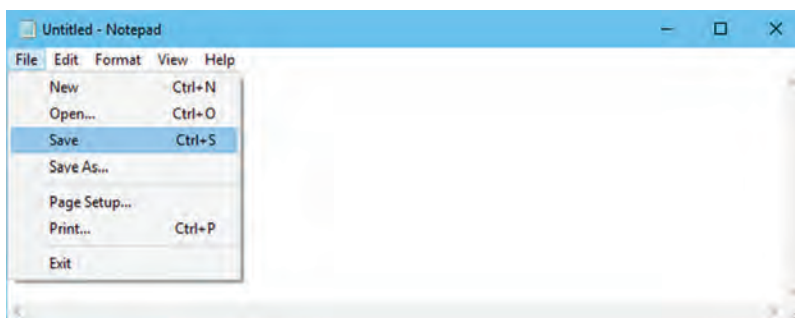
یادداشت



برای اضافه کردن اطلاعات به انتهای پرونده متنی و حفظ اطلاعات قبلی آن از متد AppendAllText استفاده می‌شود.

کارگاه ۲ پروژه Notepad

می‌خواهیم برنامه ایجاد و ذخیره پرونده در Notepad ویندوز را به کمک متدهای نوشتن در پرونده متنی پیاده‌سازی کنیم.



شکل ۳- فرم برنامه شبیه‌ساز Notepad

۱ پروژه ویندوزی با نام Notepad ایجاد کنید.

۲ فرم مناسب را طراحی کنید.

یک کنترل کادرتن با نام txtMainDocument به فرم اضافه کرده، ویژگی‌های آن را تعیین کنید (شکل ۳). یک کنترل MenuStrip به فرم اضافه کرده، با استفاده از گزینه Insert Standard Items گزینه‌های استاندارد را به منو اضافه کنید و گزینه‌های منو را به گونه‌ای تغییر دهید که منو، مشابه برنامه Notepad ویندوز شود.

۳ برای نگهداری نام و مسیر ذخیره آخرین پرونده، متغیر مناسب را اعلان کنید.

یک متغیر از نوع رشته‌ای با نام documentFilePath در بخش کلاس فرم تعریف کنید.

namespace notepad

```
{  
  
    public partial class MainForm : Form  
  
    {  
  
        string documentFilePath = "";
```

۴ یک کادر محاوره‌ای SaveFileDialog به فرم اضافه کنید.

ویژگی Filter این کادر محاوره‌ای را طوری تنظیم کنید که پرونده‌های متنی را نمایش دهد.

۵ متد رویداد کلیک گزینه **Save As** منوی **File** را بنویسید.

```
private void saveAsToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        //به روزرسانی مسیر پرونده//
        documentFilePath = saveFileDialog1.FileName;
        //نوشتن متن در پرونده//
        File.WriteAllText(documentFilePath, txtMainDocument.Text);
    }
}
```

برنامه را اجرا کنید. متن دلخواهی بنویسید و در یک پرونده متنی جدید ذخیره کنید.
 ۶ برای گزینه‌های **Save** و **New** منوی **File** کدنویسی کنید.
 ۷ برنامه را توسعه دهید.

برنامه را توسعه دهید که هنگام ایجاد پرونده جدید و یا خروج از برنامه، در صورت نیاز به ذخیره پرونده جاری، تأیید کاربر را با نمایش پیام مناسب دریافت کند.

برنامه‌ای بنویسید که قابلیت ذخیره مخاطبان در دفترچه تلفن را داشته باشد (شکل ۴).

علی	علوی	35429090	نهران
بوس	محمدی	50901020	مسهد
رضا	فرهادی	43509070	نریر
محمد	محمدی	38904409	بوسهر
احمد	رضوی	22233390	اصفهان

نام: احمد
 نام خانوادگی: رضوی
 تلفن: 22233390
 نشانی: اصفهان
 اضافه به فهرست <<<

شکل ۴- فرم دفترچه تلفن

به کمک متد **AppendAllText** همه اطلاعات یک مخاطب را در یک خط به پرونده متنی **ContactList.txt** اضافه کنید.

فعالیت کارگاهی



بازی اعصاب‌سنج در فصل ۴ را توسعه دهید تا پس از پایان بازی نام کاربر و امتیازش را در یک پرونده متنی ذخیره کند.

فعالیت منزل





آنچه آموختم:

۱.
۲.
۳.

جدول ارزشیابی شایستگی‌های غیر فنی، ایمنی، بهداشت و توجهات زیست محیطی



نمره	استاندارد (شاخص‌ها/داوری/نمره‌دهی)	نتایج ممکن	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و...)	شایستگی‌ها
۲	توجه به مواردی که باعث بروز خطا در برنامه می‌شود و مدیریت آنها - توجه به نحوه چینش کنترل‌ها در فرم و تنظیم ویژگی‌های آنها از لحاظ دسترسی راحت به آنها و زیبایی ظاهر واسط کاربری	قابل قبول	تنظیم و اصلاح عملکردهای سیستم، عیب‌یابی مشکلات و نقصان سیستم - زبان فنی	شایستگی‌های غیر فنی
			رعایت ارگونومی	ایمنی و بهداشت
۱	توجه به ایمنی و بهداشت محیط کارگاه	غیر قابل قبول	نگهداری داده به صورت الکترونیکی کاهش مصرف کاغذ و نوشت افزار	توجهات زیست محیطی
			دقت در مدیریت خطاهای احتمالی - طراحی واسط گرافیکی کاربرپسند	نگرش

● این شایستگی‌ها در ارزشیابی پایانی واحد یادگیری باید مورد توجه قرار گیرند.

ارزشیابی مرحله ۱



نمره	استاندارد (شاخص‌ها/داوری/نمره‌دهی)	نتایج ممکن	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و...)	مراحل کار
۳	ایجاد پرونده متنی جدید در مسیر تعیین شده - افزودن داده به پرونده متنی موجود - رفع خطای برنامه	بالتر از حد انتظار	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۱۵ دقیقه	ایجاد پرونده متنی
۲	ایجاد پرونده متنی جدید در مسیر تعیین شده - افزودن داده به پرونده متنی موجود	در حد انتظار		
۱	ایجاد پرونده متنی در مسیر جاری	پایین‌تر از حد انتظار		

خواندن از پرونده متنی

همان طور که برای خواندن یک کتاب، ابتدا آن را باز کرده، شروع به خواندن صفحه‌ای می‌کنیم و در انتها کتاب را می‌بندیم، برای خواندن داده‌های یک پرونده نیز باید عملیات زیر را انجام دهیم:

۱ باز کردن پرونده به منظور خواندن داده‌ها

۲ خواندن داده‌های پرونده

۳ بستن پرونده پس از اتمام کار

در فضای نام System.IO، متدهای ReadAllText و ReadAllLines از کلاس File هر سه عملیات را انجام می‌دهند.

کارگاه ۳ خواندن پرونده متنی با متد ReadAllText

می‌خواهیم در برنامه Notepad محتویات یک پرونده متنی را با کلیک روی گزینه Open ببینیم.

۱ پروژه Notepad را باز کنید.

۲ یک کادر محاوره‌ای OpenFileDialog به فرم اضافه کنید.

ویژگی Filter این کنترل را برای نمایش پرونده‌های متنی تنظیم کنید.

۳ متد رویداد کلیک گزینه Open از منوی File را بنویسید.

```
private void openToolStripMenuItem_Click(object sender, EventArgs e)
```

```
{
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        documentFilePath = openFileDialog1.FileName;
        txtMainDocument.Text = File.ReadAllText(documentFilePath);
    }
}
```

برنامه را اجرا کرده، یک پرونده متنی دلخواه را باز کنید. عملکرد متد رویداد کلیک گزینه Open چیست؟

۴ برنامه را توسعه دهید.

کد مرحله ۳ را طوری تغییر دهید تا قبل از باز کردن پرونده جدید، کادر محاوره‌ای ذخیره پرونده باز شده قبلی، نمایش داده شود.

- با تنظیم ویژگی راست به چپ فرم و ترجمه گزینه‌های منو به زبان فارسی، برنامه Notepad را فارسی کنید.

- برای گزینه‌های منوی Edit برنامه Notepad کدنویسی کنید.

راهنمایی: از ویژگی‌های SelectedText، SelectionStart و SelectionLength کادر متن، برای تعیین متن انتخاب شده، ابتدای متن انتخاب شده و طول متن انتخاب شده استفاده کنید.

فعالیت منزل



می‌خواهیم با اجرای برنامه دفترچه تلفن، در ابتدای برنامه مخاطبان ثبت‌شده در پرونده متنی در فهرست نمایش داده شوند.

۱ پروژۀ دفترچه تلفن را باز کنید.

۲ متد رویداد Form1_Load را بنویسید.

```
private void Form1_Load(object sender, EventArgs e)
{
    string[] data;
    data = File.ReadAllLines ("ContactList.txt");
    lstContacts.Items.AddRange (data);
}
```

برنامه را اجرا کنید. چه اطلاعاتی را در فهرست مشاهده می‌کنید؟

با تعریف یک نقطه توقف (break point) روی دومین خط متد رویداد Load فرم، برنامه را با کلید F11 گام‌به‌گام اجرا کنید. رشته data حاوی چه مقادیری است؟ عملکرد متد ReadAllLines چیست؟ عملکرد متد AddRange چیست؟

۳ کد زیر را جایگزین کد رویداد Load فرم کنید.

```
string data;
data = File.ReadAllText("ContactList.txt");
lstContacts.Items.Add(data);
```

کد مرحله ۲ و ۳ را با هم مقایسه کنید.

برنامه را اجرا کنید. تفاوت متد ReadAllLines با متد ReadAllText در چیست؟

۴ پرونده ContactList.txt را از پوشه Debug برنامه حذف کنید.

برنامه را اجرا کنید. چه خطایی رخ می‌دهد؟ چرا؟

۵ کد زیر را جایگزین کد رویداد Load فرم کنید.

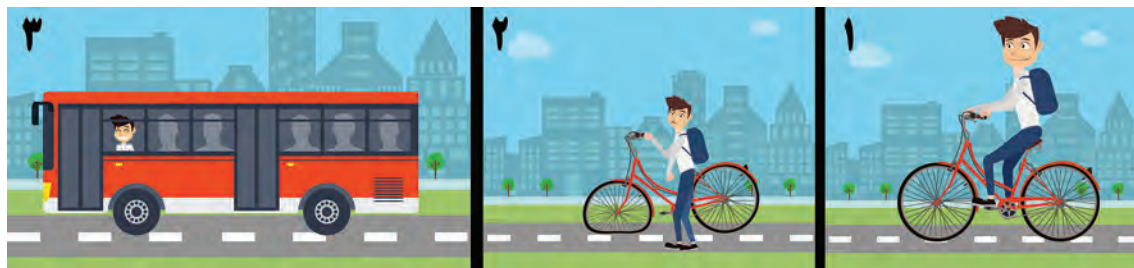
```
if (File.Exists("ContactList.txt"))
    lstContacts.Items.AddRange(File.ReadAllLines("ContactList.txt"));
else
```

```
    MessageBox.Show ("خطا", "پرونده مخاطبان موجود نیست",
    MessageBoxButtons.OK, MessageBoxIcon.Warning);
```

پیش از اینکه بخواهیم در یک پرونده چیزی بنویسیم و یا داده‌ای را از یک پرونده بخوانیم، باید از وجود پرونده اطمینان حاصل کنیم. با استفاده از متد Exists از کلاس File می‌توانیم این کار را انجام دهیم. برنامه را اجرا کنید. نوع خروجی متد Exists چیست؟

مدیریت خطا در برنامه

رضا هر روز شاد و خندان با دوچرخه به محل کار خود می‌رود. اگر یک روز دوچرخه رضا دچار مشکل فنی شود، آیا بروز این مشکل اجرای برنامه روزانه او را متوقف می‌کند؟ خیر، به عنوان یک راه‌حل جایگزین، او می‌تواند از اتوبوس یا مترو برای رسیدن به محل کار خود استفاده کند.



استثناها (Exceptions) خطاهایی هستند که هنگام اجرای برنامه ممکن است رخ دهند، مانند تقسیم یک عدد بر صفر و یا تبدیل یک رشته غیر عددی به عدد و در صورتی که برای روبه‌رو شدن و برخورد با آنها برنامه‌ای نداشته باشیم، باعث توقف روند عادی برنامه ما خواهند شد.

فیلم شماره ۱۱۱۱۰: مدیریت خطا در برنامه

فیلم



فیلم را مشاهده کرده، کارگاه ۵ را انجام دهید.

کارگاه ۵ مدیریت خطا با دستور try-catch

می‌خواهیم اطلاعات مخاطبان دفترچه تلفن را روی حافظه جانبی نظیر دیسک نرم یا حافظه فلش ذخیره کنیم و در صورت عدم وجود این حافظه‌ها عکس‌العمل مناسب از برنامه مشاهده کنیم. در زبان سی شارپ از دستورات try-catch برای مدیریت موارد استثنا استفاده می‌شود.

شکل ساده ساختار try-catch

```
try
{
    بخشی از کد که می‌خواهیم استثناها در آن کنترل شوند
}
catch
{
    دستوراتی که در صورت بروز یک استثنا اجرا می‌شوند
}
```

اگر در قسمت `try` یک حالت استثنا رخ دهد، کنترل برنامه به قسمت `catch` منتقل می‌شود. در این قسمت می‌توان خطای زمان اجرا را مدیریت کرده و پیام مناسبی به کاربر نشان داد.

۱ پروژۀ دفترچه تلفن را باز کنید.

۲ کد متد رویداد کلیک دکمه «افزودن به فهرست» را به صورت زیر تغییر دهید.

```
string contact = string.Format("{0}\t{1}\t{2}\t{3}\t", txtFirstName.Text  
txtLastName.Text, txtPhone.Text, txtAddress.Text);  
File.AppendAllText("a:\\ContactList.txt", contact);
```

برنامه را اجرا کنید. چه خطایی مشاهده می‌کنید؟

۳ با کمک هنرآموز خود پیام نشان داده شده در پنجره خطا را به فارسی معنی کنید.

۴ با استفاده از ساختار `try-catch` خطای رخ داده را مدیریت کنید.

```
try  
{  
    File.AppendAllText("a:\\ContactList.txt", contact);  
}  
catch  
{  
    MessageBox.Show("خطا", "اجرای عملیات امکان پذیر نیست. لطفاً یک بار دیگر سعی کنید",  
        MessageBoxButtons.OK, MessageBoxIcon.Warning);  
}
```

برنامه را اجرا کنید. اجرای برنامه در این حالت چه تفاوتی با حالت قبل دارد؟

در هنگام کار با متدهایی که روی پرونده‌ها کار می‌کنند، از ساختار `try-catch` برای مدیریت حالت‌های استثنا استفاده کنید.

یادداشت



ارزشیابی مرحله ۲



مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره‌دهی)	نمره
خواندن از پرونده متنی	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۲۰ دقیقه	بالاتر از حد انتظار	خواندن محتویات پرونده متنی - دسترسی به خطوط پرونده متنی - رفع خطای برنامه	۳
		در حد انتظار	خواندن محتویات پرونده متنی - دسترسی به خطوط پرونده متنی	۲
		پایین‌تر از حد انتظار	خواندن محتویات پرونده متنی	۱

مشاهده فهرست پرونده‌ها و پوشه‌ها

برنامه File Explorer ویندوز، ابزاری برای کاوش پرونده‌ها و پوشه‌های روی سیستم است. این برنامه به کاربر امکان می‌دهد که درایوهای سیستم را انتخاب کند، پوشه‌ها را باز کند و بتواند پرونده‌های درون پوشه‌ها را مشاهده کند. همچنین این برنامه امکان انجام عملیاتی نظیر ایجاد نسخه مشابه، حذف و جابه‌جا کردن پرونده‌ها و پوشه‌ها را برای کاربر فراهم می‌کند.

کلاس Directory

در فضای نام System.IO کلاس Directory برای کار با پوشه‌ها در نظر گرفته شده است. با کمک متدهای موجود در این کلاس می‌توان عملیاتی نظیر ایجاد، حذف، ایجاد نسخه مشابه، جابه‌جایی و برخی کارهای دیگر را روی پوشه‌ها انجام داد (جدول ۱).

جدول ۱- عملکرد برخی از متدهای کلاس Directory

نام متد	نوع ورودی	نوع خروجی	عملکرد
GetDirectories	رشته (مسیر)	آرایه رشته‌ای	
GetFiles			نام و مسیر همه پرونده‌های مسیر ورودی را در یک آرایه برمی‌گرداند.
Exists			

با استفاده از راهنمای مایکروسافت جدول ۱ را تکمیل کنید.

فعالیت کارگاهی



مسیر (path) رشته‌ای است که موقعیت یک پرونده یا پوشه را مشخص می‌کند. در فضای نام System.IO، کلاس Path برای انجام برخی کارها روی مسیرها مانند استخراج پسوند یا نام پرونده از مسیر در نظر گرفته شده است (جدول ۲).

جدول ۲- عملکرد برخی از متدهای کلاس Path

نام متد	نوع ورودی	نوع خروجی	عملکرد
GetFileName			نام و پسوند پرونده ورودی را برمی‌گرداند.
Combine	دو رشته (مسیر) یا بیشتر		

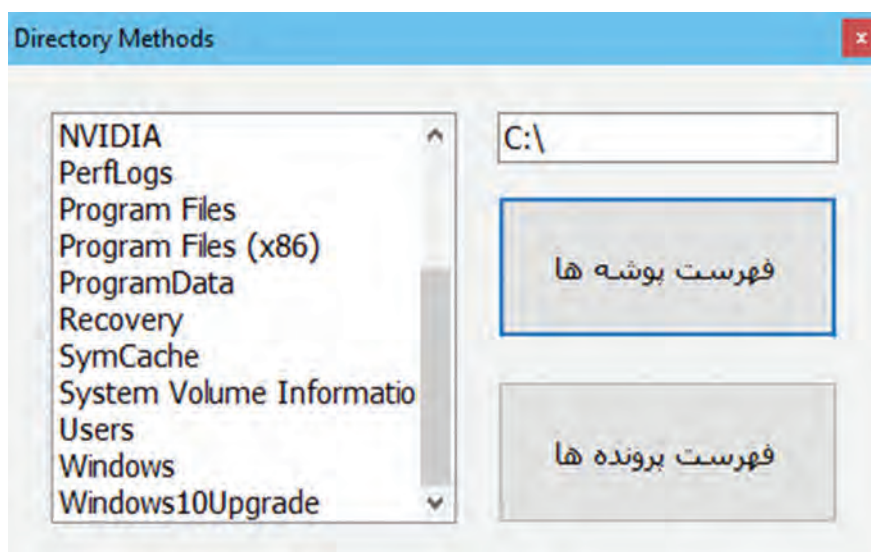
با استفاده از راهنمای میکروسافت جدول ۲ را تکمیل کنید.

فعالیت کارگاهی



کارگاه ۶ کاربرد کلاس Directory و Path

می‌خواهیم با یک برنامه نمونه، عملکرد برخی از متدهای کلاس Directory را ببینیم.



شکل ۵- فرم برنامه متدهای کلاس Directory

- ۱ پروژه ویندوزی به نام DirectoryMethods ایجاد کنید.
- ۲ برای برنامه فرم مناسب طراحی کنید (شکل ۵).

نام کنترل‌ها را مطابق جدول زیر تغییر دهید.

کنترل	دکمه «فهرست پرونده‌ها»	دکمه «فهرست پوشه‌ها»	کادر فهرست	کادر متن
Name	btnShowFiles	btnShowDirectoris	LstBoard	txtPath

۲ فضای نام **System.IO** را به برنامه معرفی کنید.

۴ متد رویداد کلیک دکمه «فهرست پوشه‌ها» را بنویسید.

```
private void btnShowDirectoris_Click(object sender, EventArgs e)
```

```
{
    string directory = @"C:\";
    lstBoard.Items.Clear();
    lstBoard.Items.AddRange(Directory.GetDirectories(directory));
}
```

برنامه را اجرا کنید و روی دکمه «فهرست پوشه‌ها» کلیک کنید. چه عناصری در فهرست مشاهده می‌کنید؟

۵ کد مرحله ۴ را تغییر دهید تا مسیر **directory** را از کادر متن دریافت کند.

۶ متد رویداد کلیک دکمه «فهرست پرونده‌ها» را بنویسید.

```
private void btnShowFiles_Click(object sender, EventArgs e)
```

```
{
    string directory = @"C:\";
    lstBoard.Items.Clear();
    lstBoard.Items.AddRange(Directory.GetDirectories(directory));
}
```

برنامه را اجرا کنید و روی دکمه «فهرست پرونده‌ها» کلیک کنید. چه عناصری در فهرست مشاهده می‌کنید؟

۷ کد مرحله ۶ را تغییر دهید تا مسیر **directory** را از کادر متن دریافت کند.

۸ متد رویداد کلیک دکمه «فهرست پوشه‌ها» را ویرایش کنید.

در خروجی متدهای **GetFiles** و **GetDirectories** نام زیرپوشه‌ها و پرونده‌ها به همراه مسیر مشاهده می‌شود.

برای اینکه فقط نام زیرپوشه و پرونده را مشاهده کنید، کد زیر را بنویسید.

```
private void btnShowDirectoris_Click(object sender, EventArgs e)
```

```
{
```

```

string directory = @"C:\";
lstBoard.Items.Clear();
string[] direcs = Directory.GetDirectories(directory);
foreach (string dir in direcs)
    lstBoard.Items.Add(Path.GetFileName(dir));
}

```

برنامه را اجرا کنید. کد مرحله ۴ و ۸ را با هم مقایسه کنید.
۹ متد رویداد کلیک دکمه «فهرست پرونده‌ها» را ویرایش کنید.
 کدهای زیر نام پرونده‌ها بدون مسیر را نمایش می‌دهد.

```

private void btnShowFiles_Click(object sender, EventArgs e)
{
    string directory = @"C:\";
    lstBoard.Items.Clear();
    string[] files = Directory.GetFiles(directory);
    foreach (string file in files)
        lstBoard.Items.Add(Path.GetFileName(file));
}

```


کاربرد پایگاه داده

به جز شماره تلفن‌های ضروری مانند اورژانس و یا آتش‌نشانی، این روزها کمتر کسی احتیاج پیدا می‌کند که شماره تلفنی را به خاطر بسپارد. به یاری تلفن‌های همراه، کافی است برای تماس یا ارسال پیامک، در فهرست مخاطب‌های تلفن با یک جست‌وجوی ساده به اطلاعات فرد موردنظر دسترسی پیدا کنید. برنامه فهرست مخاطبان در گوشی‌های تلفن همراه، برنامه‌ای است که امکان ایجاد، ذخیره، حذف، ویرایش و جست‌وجوی اطلاعات را در اختیار کاربر قرار می‌دهد. برای تهیه چنین برنامه‌ای ابتدا باید پایگاه‌داده‌ای ایجاد کرد تا بتوان اطلاعات مخاطبان (Contacts) را در آن ذخیره کرد.



به کمک هم‌گروهی خود یک پایگاه‌داده برای ذخیره مخاطبان ایجاد کنید و به سؤالات زیر پاسخ دهید.

- چگونه پایگاه‌داده را در Access ایجاد می‌کنید؟
- چگونه در پایگاه‌داده، جدول مخاطبان را ایجاد می‌کنید؟
- چه فیلدهایی را برای جدول مخاطبان تعریف می‌کنید؟ نوع داده این فیلدها چیست؟
- چگونه رکورد جدید در جدول مخاطبان وارد می‌کنید؟
- چگونه یک رکورد را حذف می‌کنید؟

فعالیت گروهی



خیلی از نرم‌افزارهایی که ما در محیط پیرامون خود با آنها سروکار داریم، عملیات ذخیره و بازیابی اطلاعات را انجام می‌دهند. نرم‌افزارهایی نظیر برنامه فروشگاه‌ها، بانک‌ها و کتابخانه‌ها و برخی از نرم‌افزارهای بازی برای ذخیره مشخصات، سطح و امتیاز کاربران از پایگاه‌داده استفاده می‌کنند. برخی از برنامه‌ها و بازی‌های تلفن‌های هوشمند نیز از پایگاه‌داده برای ذخیره و بازیابی اطلاعات استفاده می‌کنند. در ساخت این برنامه‌ها ابتدا پایگاه‌داده مربوط به محیط عملیاتی طراحی می‌شود سپس به کمک زبان‌های برنامه‌سازی نظیر سی‌شارپ، ویژوال بیسیک، جاوا و php، عملیات ارتباط با پایگاه‌داده و عملیات مبنایی روی پایگاه‌داده انجام می‌شود.

کنجکاوی



- چند نرم‌افزار پیرامون خود را نام ببرید که از پایگاه‌داده برای ذخیره و بازیابی اطلاعات استفاده می‌کنند.
- نام بازی‌هایی را بنویسید که از پایگاه‌داده برای ذخیره اطلاعات بازی استفاده می‌کنند.
- چند نرم‌افزار فارسی تلفن هوشمند را نام ببرید که از پایگاه‌داده استفاده می‌کنند.
- آیا امکانات موجود در نرم‌افزارهای مدیریت پایگاه‌داده نظیر Access برای تولید یک نرم‌افزار که کار ذخیره و بازیابی اطلاعات را انجام دهد، کافی نیست؟ چرا؟
- یک بازی برخط (online) را بررسی کنید که مشخصات، سطح و امتیاز کاربران در آن ذخیره می‌شود. فکر می‌کنید پایگاه‌داده این بازی از چه جدول‌هایی تشکیل شده است؟

ایجاد پایگاه داده

طرح مسئله: مدیر فروشگاه فرهنگیان یک شهر کوچک به هنرستان جابر بن حیان آمده است تا از هنرآموز رشته شبکه و نرم افزار رایانه برای تولید یک برنامه رایانه ای کمک بگیرد. بعد از طرح مسئله به وسیله مدیر فروشگاه، هنرآموز این مسئله را در بین هنرجویان این گونه مطرح کرد:

فرهنگیان می توانند کالاهای خود را به صورت اعتباری از این فروشگاه خریداری کنند. مسئول فروش، اطلاعات خرید را به شکل زیر در یک جدول Excel یادداشت می کند:

نام	نام خانوادگی	تاریخ	مبلغ خرید (تومان)
محمد	محمدی	۱۳۹۶/۲/۱	۵۲۰۰۰
علی	غلامی	۱۳۹۶/۲/۱	۱۲۰۰۰۰
فرهاد	سهرابی	۱۳۹۶/۲/۲	۳۰۰۰۰
محمد	محمدی	۱۳۹۶/۲/۳	۴۵۰۰۰

جدول بالا را در Access پیاده سازی کرده، 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 می آید؟ چرا؟ مشکل را چگونه حل می کنید؟

فعالیت کارگاهی



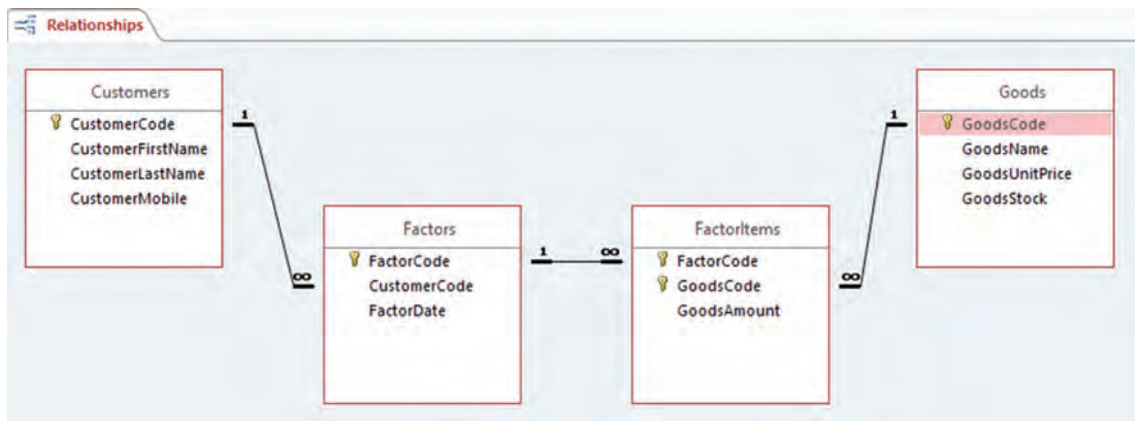
در پایان ماه مدیر، فهرست فرهنگیان را به حسابداری اداره آموزش و پرورش شهرستان ارسال می کند تا از حقوق آنها کسر شود. برخی از فرهنگیان هنگام خرید درخواست برگه خرید دارند تا بدانند در پایان ماه چه مقدار از حقوقشان کسر می شود.

کنجکاوی



- برگه خرید شامل چه داده هایی می شود؟
- برای نمایش کالاها در برگه خرید نیاز به «جدول کالا» داریم، این جدول شامل چه فیلدهایی می شود؟
- برای نمایش اطلاعات مربوط به کارمند فرهنگی در برگه خرید نیاز به «جدول مشتری» داریم، این جدول شامل چه فیلدهایی می شود؟
- برای ایجاد هر برگه خرید نیاز به «جدول برگه خرید» داریم که مشخص کند مشتری در چه تاریخی چه کالاهایی خریده است. این جدول شامل چه فیلدهایی می شود؟
- مشتری در هر بار مراجعه ممکن است چندین قلم کالا خریداری کند؛ بنابراین نمی توان کالاهای خریداری شده را در «جدول کالا» ذخیره کرد، به همین دلیل، جدول دیگری به نام «جدول ثبت برگه خرید» ایجاد می کنیم. این جدول شامل چه فیلدهایی می شود؟

شکل ۶ ساختار کلی پایگاه داده سیستم فروشگاه فرهنگیان را به صورت ساده نمایش می دهد.



شکل ۶- ساختار کلی پایگاه داده سیستم فروشگاه فرهنگیان

پایگاه داده MyShop را با توجه به شکل ۶ در Access ایجاد کرده، داده مناسب را در جدول ها وارد کنید. جدول Goods کالا، Customers جدول مشتری، Factors جدول برگه خرید و FactorItems جدول ثبت برگه خرید است.

فعالیت کارگاهی



- فرض کنید در فروشگاه فرهنگیان بیش از یک فروشنده کار فروش کالا را انجام می دهند، اگر قرار باشد در برگ خرید، نام فروشنده نیز قید شود، چه تغییری باید در ساختار پایگاه داده فروشگاه صورت گیرد؟
- در یک بازی می توان با ایجاد یک نام کاربری وارد بازی شد. هر کاربر با کسب حداقل امتیاز به سطح بالاتر می رود و با توجه به امتیاز خود ستاره کسب می کند. پایگاه داده مناسب را برای این بازی طراحی کنید تا نشان دهد که هر کاربر در هر سطح چند امتیاز و چند ستاره دارد.

فعالیت منزل



ارتباط با پایگاه داده

اولین گام برای مدیریت پایگاه داده در زبان سی شارپ ایجاد یک ارتباط بین پایگاه داده و برنامه است. ساده ترین راه برای ایجاد ارتباط استفاده از wizard است. برای نمایش اطلاعات یک جدول پایگاه داده روی فرم، از کنترل DataGridView استفاده می شود.

کارگاه ۷ ارتباط با پایگاه داده از طریق wizard

می‌خواهیم برنامه‌ای بنویسیم که واژه‌های فنی رایانه به همراه توضیح آنها را در قالب یک جدول نمایش دهد. هدف از این برنامه نمایش اطلاعات یک جدول پایگاه داده روی فرم برنامه است.

نام واژه	معنی واژه
abs	عملکردی که مقدار قدر مطلق را در زبان بیس...
absolute address	آدرس مطلق - مکان مشخصی در حافظه...
absolute URL	یو آر ال مطلق - نوعی یو آر ال که حاوی آ...
abstract	خلاصه . چکیده . انتزاعی . خلاصه ای از...
accelerator	شتاب دهنده - دستگاهی که موجب ...
accente	تلفظ ها . تکیه های صدا - علامتهایی ...
acceptable-use policy	سیاستی که توسط مالک یک سیستم...
access provider	شرکتی که دسترسی به شبکه اینترنت...
access time	مقدار زمان لازم برای آنکه یک دستگاه ح...
accounting software	نرم افزار حسابداری - نرم افزاری که برا...

۱ پایگاه داده جدیدی به نام dictionary در Access با قالب mdb ایجاد کنید.

۲ جدول Computer را با فیلدهای زیر ایجاد کنید.

Field Name	Data Type	
WordCode	AutoNumber	کد واژه
WordName	Short Text	نام واژه
WordMeans	Long Text	مفهوم واژه

فیلد WordName باید یکتا باشد. چگونه از تکراری بودن آن جلوگیری می‌کنید؟

۳ داده‌های مناسب را در جدول Computer وارد کنید.

۴ یک پروژه ویندوزی به نام ComputerDictionary در VS ایجاد کنید.

۵ کنترل DataGridView را از جعبه ابزار به فرم اضافه کنید.

کنترل DataGridView برای نمایش داده‌ها در قالب یک جدول به کار می‌رود.

۶ کنترل DataGridView را به منبع اطلاعات متصل کنید.

فیلم شماره ۱۱۱۱: نمایش داده‌های جدول در DataGridView

فیلم



فعالیت کارگاهی



پس از مشاهده فیلم، کنترل DataGridView را به بانک اطلاعاتی متصل کنید.

- چه کنترل‌هایی به فرم برنامه اضافه شده است؟

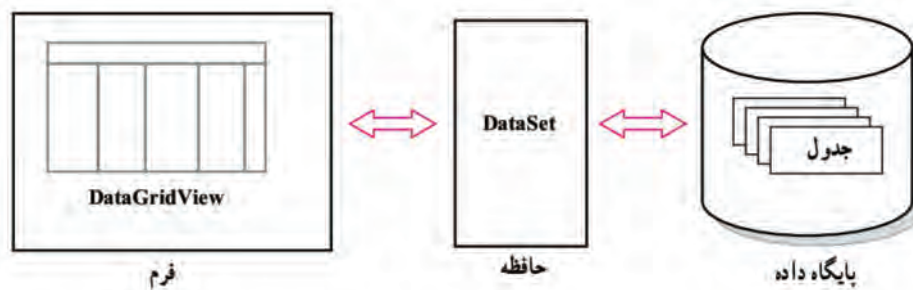
- در بخش Solution Explorer چه تغییری رخ داده است؟ چه پرونده‌ای به این بخش اضافه شده است؟

- کلید F7 را فشار دهید. آیا کدی به برنامه اضافه شده است؟

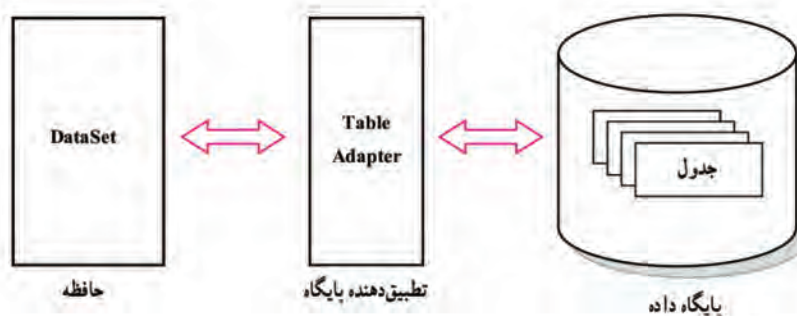


برای ارتباط پروژه ویندوزی VS2012 با پایگاه داده Access با قالب accdb (Access2013 به بالا) باید بسته نرم‌افزاری Access Database Engine را متناسب با سیستم‌عامل خود از تارنمای مایکروسافت بارگیری و نصب کنید.

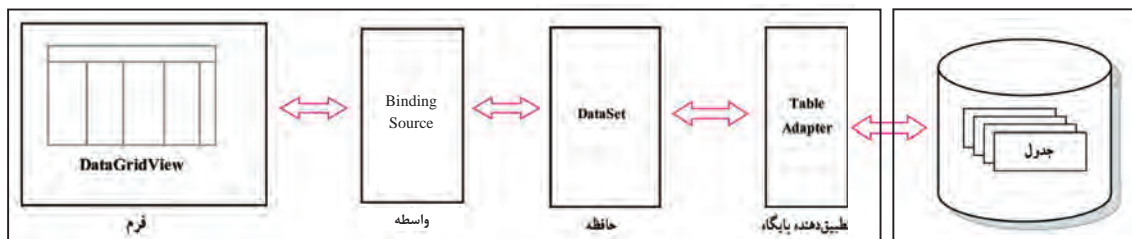
با ایجاد ارتباط برنامه به بانک اشیای زیر به صورت خودکار به فرم پروژه اضافه می‌شود:
الف) DictionaryDataSet که از نوع کلاس DataSet است و برای ذخیره داده‌های پایگاه داده در حافظه به کار می‌رود و می‌تواند اطلاعات یک یا چند جدول را ذخیره کند.



ب) ComputerTableAdapter ارتباط بین پایگاه داده و نرم‌افزار کاربردی را مهیا می‌کند.



ج) ComputerBindingSource که از نوع کلاس BindingSource است. برای نمایش داده‌ها روی فرم از واسط دیگری به نام BindingSource استفاده می‌شود. به کمک این کنترل، دسترسی کنترل‌های نمایش داده نظیر DataGridView به بخشی از تصویر اطلاعات موجود در DataSet فراهم می‌شود.



پروژه (برنامه به زبان C#)

پایگاه داده

در کتابخانه .NET. مجموعه‌ای از کلاس‌های قدرتمند برای اتصال به پایگاه داده و انجام عملیات مختلف روی داده، تحت عنوان ADO.NET تعریف شده است.

کارگاه ۸ سفارشی کردن کنترل DataGridView

می‌خواهیم با تغییر برخی از ویژگی‌ها، نمایش داده را در کنترل DataGridView سفارشی کنیم. با اجرای برنامه کارگاه ۷ مشاهده کردید که نام سرستون‌های (Headers) جدول در کنترل DataGridView فارسی نیست و اندازه پهنای ستون‌ها تنظیم نیست.

۱ روی کنترل DataGridView راست کلیک کنید و گزینه Edit Columns را انتخاب کنید.

۲ فیلدهایی را که می‌خواهید نمایش داده نشود، حذف کنید.

کدام فیلد نباید در واژه‌نامه نمایش داده شود؟ بعد از انتخاب فیلد با کلیک روی دکمه Remove، فیلد را حذف کنید.

۳ ویژگی‌های ستون‌ها را تنظیم کنید.

با کلیک روی هر فیلد می‌توان ویژگی‌های مربوط به آن فیلد را تغییر داد. ویژگی HeaderText فیلد عنوان سرستون را نشان می‌دهد، با تغییر این ویژگی عنوان سرستون‌ها را به زبان فارسی بنویسید و دکمه OK را انتخاب کنید. در کنترل DataGridView چه تغییری اتفاق می‌افتد؟

۴ ویژگی Dock کنترل را به Fill تغییر دهید. چه تغییری مشاهده می‌کنید؟

۵ ویژگی AutoSizeColumnsMode را به Fill تغییر دهید. چه تغییری مشاهده می‌کنید؟

۶ مقدار ویژگی DataSource را None کنید.

چه تغییری مشاهده می‌کنید؟ مقدار قبلی را در DataSource قرار دهید. چه تغییری مشاهده می‌کنید؟ ویژگی DataSource کنترل DataGridView، نام همان BindingSource است که به فرم اضافه شده است.

۷ رنگ قلم و زمینه کنترل را تغییر دهید.

برای قالب‌بندی کنترل DataGridView چگونه ویژگی DefaultCellStyle را تغییر می‌دهید؟

برنامه را اجرا کنید. روی سرستون «واژه» کلیک کنید. چه تغییری در نمایش داده‌ها اتفاق می‌افتد؟

آیا می‌توانید از طریق این کنترل رکوردی را حذف یا اضافه کنید؟ چه ویژگی را باید تغییر دهید؟

جدول ۳ برخی از ویژگی‌های کنترل DataGridView را نمایش می‌دهد. این جدول را تکمیل کنید.

جدول ۳- ویژگی‌های کنترل DataGridView

نام ویژگی	شرح
Dock	مشخص می‌کند که کدام لبه از کنترل به فرم محدود شود.
AutoSizeColumnsMode	وضعیت اندازه خودکار برای ستون را مشخص می‌کند.
DefaultCellStyle	
DataSource	
HeaderText	
ColumnHeadersDefaultCellStyle	

فعالیت کارگاهی





- جدول مخاطبان تلفن همراه را روی فرم نمایش دهید.
- از پایگاه داده فروشگاه، جدول کالا را روی فرم نمایش دهید.

در مورد ویژگی‌های دیگر کنترل DataGridView تحقیق کنید.

کارگاه ۹ مقیدسازی کنترل‌ها

در کارگاه ۷ به کمک wizard داده‌های جدول را در کنترل DataGridView نمایش دادیم. آیا می‌توان مقدار هر فیلد را جداگانه در کنترل‌هایی نظیر کادرمتن یا کادرفهرست نمایش داد؟ می‌خواهیم در برنامه واژه‌نامه فنی با کلیک روی هر رکورد در کنترل DataGridView واژه و معنی آن در دو کادرمتن جداگانه نمایش داده شود.



شکل ۷- فرم پروژه

۱ پروژه ویندوزی ComputerDictionary را باز کنید.

۲ فرم پروژه را مطابق شکل ۷ طراحی کنید.

دو کنترل کادرمتن با نام‌های txtWordName و txtWordMeans به فرم پروژه اضافه کنید. برای فراهم کردن امکان نمایش چند خط در کادرمتن txtWordMeans، کدام ویژگی کادرمتن را باید تغییر داد؟

۳ کادرهای متن را به فیلد موردنظر مقید کنید.

ویژگی DataBindings کنترل txtWordName، را باز کنید. روی بخش Advanced کلیک کنید. فهرست کشویی Binding را باز کنید و از منبع داده (computerBindingSource) فیلد موردنظر (WordName) را انتخاب کنید. برنامه را اجرا کنید و روی یک سطر دلخواه در DataGridView کلیک کنید. چه مقداری در txtWordName قرار می‌گیرد؟

در این حالت می‌گوییم کنترل txtWordName مقید به فیلد WordName شده است. کنترل کادرمتن txtWordMeans را نیز مقید به فیلد WordMeans کنید و برنامه را اجرا کنید.

۴ مقیدسازی یک کنترل کادر فهرست (ListBox) را بررسی کنید.

یک کنترل کادر فهرست به فرم اضافه کنید و ویژگی DataSource و DisplayMember آن را تغییر دهید. برای مقیدسازی کنترل کادر فهرست چگونه ویژگی DataSource و DisplayMember را تغییر می‌دهید؟

۵ یک کنترل کادر ترکیبی (ComboBox) را مقید به فیلد WordName کنید.

یک کنترل کادر ترکیبی را به فرم اضافه کنید و آن را مقید به فیلد WordName کنید.

۶ عملکرد کنترل BindingNavigator را بررسی کنید.

کنترل BindingNavigator را به فرم اضافه کنید و ویژگی DataSource آن را تنظیم کنید. سپس برنامه را اجرا کنید.

کنترل کادرمتن با کدنویسی به صورت زیر مقید می‌شود:

```
txtWordName.DataBindings.Add("Text", computerBindingSource, "WordName");
```

و کنترل کادر فهرست به صورت زیر مقید می‌شود:

```
listBox1.DataSource = computerBindingSource;
```

```
listBox1.DisplayMember = "WordName";
```

- چه کنترل‌هایی را می‌توان مقید کرد؟ چند کنترل را نام ببرید.
- چگونه با کدنویسی می‌توان یک کنترل را نامقید کرد؟

کنجکاوی



فعالیت کارگاهی



پس از طراحی پایگاه داده فروشگاه و ورود داده‌های مناسب در جدول‌ها، فرم برنامه فروشگاه را به صورت شکل ۸ طراحی کنید.

کد کالا	نام کالا	قیمت واحد	موجودی
1	شیر	2000	100
2	ماست	1500	50
3	برنج	30000	20
4	تخم مرغ	500	3000
5	نخود	2500	20
6	کنسرو لوبیا	3000	50
7	شکر	3000	100
8	فند	2500	60
9	دستمال کاغذ...	2000	30
10	بنبر خامه ای	3200	15
11	کلوچه	500	100
12	کیک	500	200
13	خامه	3000	30
14	آرد گندم	2000	100
15	ناگت مرغ	6000	20

شکل ۸- فرم برنامه فروشگاه

کارگاه ۱۰ بارگذاری داده و پیمایش رکوردها در DataGridView

می‌خواهیم با کدنویسی DataGridView فرم کالا را پر کنیم و برای دکمه‌های پیمایش کدنویسی کنیم.

۱ پروژه فروشگاه را که در فعالیت کارگاهی فرم آن را طراحی کردید، باز کنید.

نام کنترل DataGridView را dgvGoods، نام فرم را MainForm و نام دکمه‌های پیمایش را مطابق جدول زیر قرار دهید.

نام کنترل	btnLast	btnNext	btnPrev	btnFirst
عملکرد	رفتن به آخرین رکورد	رفتن به رکورد بعدی	رفتن به رکورد قبلی	رفتن به اولین رکورد
Text	>>	>	<	<<

۲ یک ارتباط برای دسترسی به تمام جدول‌های پایگاه‌داده MyShop ایجاد کنید.

۳ رویداد MainForm_Load را برای بارگذاری داده‌های جدول کالا در کنترل dgvGoods بنویسید.

```
private void MainForm_Load(object sender, EventArgs e)
```

```
{
    this.goodsTableAdapter.Fill(this.myShopDataSet.Goods);
    dgvGoods.DataSource = goodsBindingSource;
}
```

TableAdapter دارای متد Fill است که یک DataSet را از داده‌های جدول پر می‌کند.

myShopDataSet شامل چه جدول‌های دیگری است؟

کنجکاوی



برنامه را اجرا کنید. آیا می‌توان برای مقیدسازی کنترل dgvGoods دستور دیگری نوشت؟

۴ برای رفتن به اولین رکورد کد زیر را بنویسید.

```
private void btnFirst_Click(object sender, EventArgs e)
```

```
{
    goodsBindingSource.MoveFirst();
}
```

برنامه را اجرا کنید. برای رفتن به آخرین رکورد چه دستوری می‌نویسید؟

۵ برای رفتن به رکورد قبلی کد زیر را بنویسید.

```
private void btnPrev_Click(object sender, EventArgs e)
```

```
{
    goodsBindingSource.MovePrevious();
}
```

برنامه را اجرا کنید. برای رفتن به رکورد بعدی چه کدی می‌نویسید؟



در سربرگ مشتری، فرمی همانند فرم کالا ایجاد کنید و از طریق کدنویسی بارگذاری داده در کنترل DataGridView را انجام دهید و دکمه‌های پیمایش رکوردها را ایجاد کنید.

کارگاه ۱۱ دسترسی به خانه‌های DataGridView

می‌خواهیم در برنامه فروشگاه، بدون مقیدسازی کنترل کادرمتن با کلیک روی کنترل dgvGoods، مقادیر فیلدهای رکورد در کادرمتن متناظر قرار گیرد.

۱ پروژۀ فروشگاه را باز کنید.

۲ نام چهار کنترل کادرمتن در پنل را به txtGoodsCode و txtGoodsName و txtUnitPrice و txtStock تغییر دهید.

۳ متد رویداد CellClick کنترل dgvGoods را بنویسید.

```
private void dgvGoods_CellClick(object sender, DataGridViewCellEventArgs e)
```

```
{
    int index = dgvGoods.CurrentRow.Index;
    txtGoodsCode.Text = dgvGoods.Rows[index].Cells[0].Value.ToString();
    txtGoodsName.Text = dgvGoods.Rows[index].Cells[1].Value.ToString();
    txtUnitPrice.Text = dgvGoods.Rows[index].Cells[2].Value.ToString()
    txtStock.Text = dgvGoods.Rows[index].Cells[3].Value.ToString();
}
```

رویداد CellClick زمانی رخ می‌دهد که روی یک خانه دلخواه DataGridView کلیک کنید.

شکل کلی دسترسی به شماره رکورد جاری

DataGridView.CurrentRow.Index نام کنترل

کنترل DataGridView همانند یک جدول دارای سطر و ستون است. برای دسترسی به خانه‌های یک رکورد ابتدا اندیس سطر (Rows) و سپس اندیس ستون (Cell) را مشخص می‌کنیم.

شکل کلی دسترسی به مقدار یک سلول DataGridView

DataGridView.Rows [شماره سطر].Cells [شماره ستون].Value نام کنترل

مقدار اندیس (شماره) اولین رکورد چیست؟



برنامه را اجرا کنید و با کلیک روی یک سطر دلخواه، نتیجه کار را ببینید.

برای سربرگ مشتری برنامه فروشگاه، عملیات بالا را انجام دهید.

فعالیت کارگاهی



برداشت



آنچه آموختم:

۱.
۲.
۳.

ارزشیابی مرحله ۳



نمره	استاندارد (شاخص‌ها/داوری/نمره‌دهی)	نتایج ممکن	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	مراحل کار
۳	طراحی واسط کاربری و ایجاد ارتباط واسط کاربری با پایگاه داده - مقیدسازی کنترل - پیمایش رکوردها - رفع خطا	بالاتر از حد انتظار	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۲۰ دقیقه	مقیدسازی واسط گرافیکی کاربر
۲	طراحی واسط کاربری و ایجاد ارتباط واسط کاربری با پایگاه داده - مقیدسازی کنترل - پیمایش رکوردها	در حد انتظار		
۱	طراحی واسط کاربری و ایجاد ارتباط واسط کاربری با پایگاه داده	پایین‌تر از حد انتظار		

عملیات در پایگاه‌داده

جست‌وجو، درج، حذف و ویرایش رکوردها از عملیات پراهمیت در پایگاه‌داده است. پس از ایجاد ارتباط اولیه پایگاه‌داده MyShop با برنامه، یک پرونده جدید به نام myShopDataSet.xsd در بخش Solution Explorer ایجاد می‌شود که با دابل کلیک روی آن می‌توان جدول‌ها، نام فیلدهایشان، TableAdapter و متدهای مربوط به آنها را مشاهده کرد. با راست کلیک روی هر کدام از این جدول‌ها و انتخاب گزینه Query از منوی Add می‌توان یک پرس‌وجو برای انتخاب داده‌ها (جست‌وجو) و یا پرس‌وجو برای عملیات درج، حذف و ویرایش ایجاد کرد.

کارگاه ۱۲ جست‌وجو در جدول

می‌خواهیم قابلیت جست‌وجوی واژه را به برنامه «واژه‌نامه فنی رایانه» اضافه کنیم.

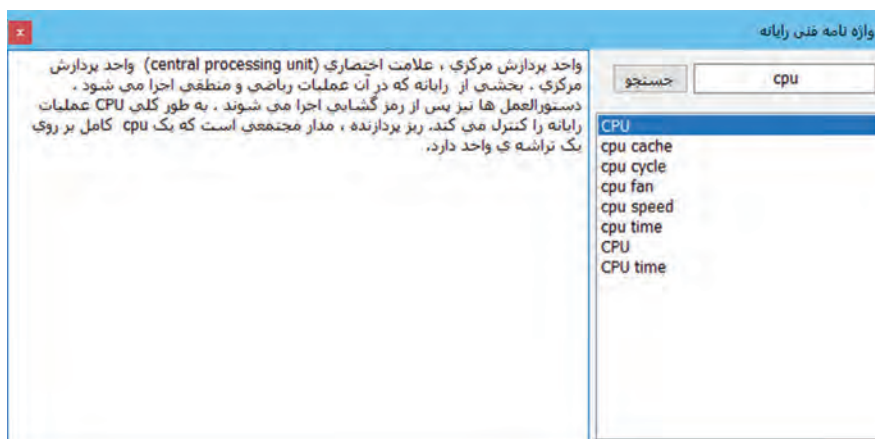
۱ پروژه ویندوزی به نام Dictionary ایجاد کنید.

۲ یک کنترل SplitContainer به فرم برنامه اضافه کنید.

کنترل SplitContainer ناحیه نمایش یک ظرف (Container) را به دو بخش تقسیم می‌کند. این کنترل شامل یک نوار متحرک است که با جابه‌جایی آن می‌توان اندازه بخش‌ها را تغییر داد.

روی چه کنترل‌هایی می‌توان کنترل SplitContainer را قرار داد؟

کنجکاوی



شکل ۹- فرم جست‌وجوی واژه

۳ پنل سمت راست را به صورت افقی به دو بخش تقسیم کنید.

پنل سمت راست را انتخاب کرده، یک کنترل SplitContainer دیگر روی آن قرار دهید. ویژگی Orientation آن را Horizontal قرار دهید. در این حالت چه تغییری مشاهده می‌کنید؟

۴ کنترل کادرفهرست را به پنل پایین در سمت راست اضافه کنید.

پنل پایین در سمت راست فرم را انتخاب کرده، یک کنترل کادرفهرست روی آن قرار داده و ویژگی Dock آن را در حالت Fill قرار دهید. در این حالت چه تغییری مشاهده می‌کنید؟ این کنترل را به فیلد WordName از جدول Computer مقید کنید.

۵ یک کنترل کادرمتن به پنل سمت چپ فرم اضافه کنید.

پنل سمت چپ فرم را انتخاب کرده، یک کنترل کادرمتن با نام txtWordMeans روی آن قرار دهید. ویژگی MultiLine را true کنید. ویژگی Dock کنترل کادرمتن را در حالت Fill قرار دهید.

این کنترل را به فیلد WordMeans از جدول Computer مقید کنید.

۶ یک کنترل کادرمتن دیگر با نام txtSearchWord برای جست‌وجوی واژه در پنل بالای فرم قرار دهید.

۷ متدی برای پرس‌وجو در جدول Computer براساس فیلد WordName ایجاد کنید.

از پنجره Solution Explorer روی پرونده dictionaryDataSet.xsd دابل کلیک کرده، آن را باز کنید. با راست کلیک روی جدول Computer، انتخاب گزینه Add و سپس گزینه Query پنجره wizard پیکربندی پرس و جو باز می شود.

گزینه Use SQL statement را انتخاب کنید، سپس دکمه Next را کلیک کنید. در ادامه گزینه SELECT which returns rows را انتخاب کرده، دکمه Next را کلیک کنید.

در این پنجره یک پرس و جوی آماده SELECT برای انتخاب همه داده های جدول مشاهده می کنید. برای اینکه بتوان یک واژه را در جدول جست و جو کرد، از طریق دکمه Query Builder پرس و جو را تغییر دهید. در پنجره Query Builder در ستون Filter در جلوی فیلد WordName علامت ؟ (علامت سؤال انگلیسی) قرار دهید. دستور SQL به صورت زیر تغییر می کند:

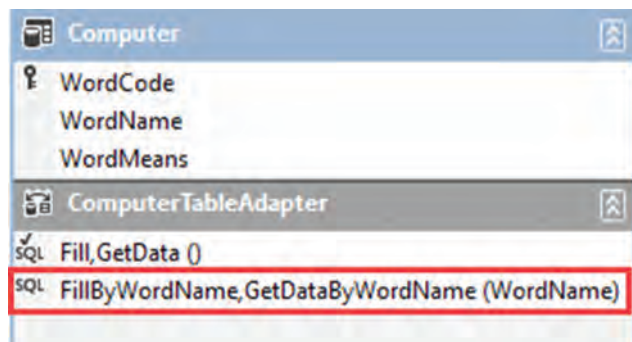
SELECT WordCode, WordName, WordMeans FROM Computer WHERE (WordName = ?)
دکمه OK و سپس دکمه Next را کلیک کنید.

عملکرد دستور SELECT چیست؟

کنجکاوی



در پنجره بعدی باید یک نام مناسب برای متد در نظر گرفته شود. در این پنجره دو نوع متد برای جست و جو می توان ایجاد کرد. متد اول یک جدول داده (DataTable) را پر می کند و خروجی ندارد. نام این متد را در کادر متن اول FillByWordName قرار دهید. متد دوم دارای یک خروجی از نوع جدول داده است. نام متد دوم را GetDataByWordName قرار دهید. دکمه Finish را کلیک کنید. متد FillByWordName با دو آرگومان ورودی برای computerTableAdapter ایجاد می شود.



علامت «سؤال» در پرس و جو نشانه ورودی های متد است.

یادداشت



▲ برای جست و جوی واژه، متد رویداد TextChanged کادر متن جست و جو (txtSearchWord) را بنویسید.
private void txtSearchWord_TextChanged(object sender, EventArgs e)

```
{
    if (txtSearchWord.TextLength > 0)
```

```
computerTableAdapter.FillByWordName(dictionaryDataSet.Computer, txtSearchWord.  
Text);
```

else

```
computerTableAdapter.Fill(dictionaryDataSet.Computer);
```

```
}
```

متد `FillByWordName`، محتوای کادرمتن را در فیلد `WordName` جدول جست‌وجو می‌کند و جدول داده `Computer` را با رکوردهایی که پیدا کرده، پر می‌کند. هم‌زمان کادرفهرست نیز واژه مورد نظر و کادرمتن `txtWordMeans` معنی واژه را نمایش می‌دهند. عملکرد دستور `if` چیست؟ برنامه را اجرا کنید. یک واژه دلخواه را جست‌وجو کنید.

۹ برای استفاده از متد `GetDataByWordName` دستور `if` را به صورت زیر ویرایش کنید.

```
if (txtSearchWord.Text.Length > 0)
```

```
computerBindingSource.DataSource =
```

```
computerTableAdapter.GetDataByWordName(txtSearchWord.Text);
```

else

```
computerTableAdapter.Fill(dictionaryDataSet.Computer);
```

این کد را با کد مرحله ۸ مقایسه کنید و تفاوت دو متد `FillByWordName` و `GetDataByWordName` را بنویسید.

نوع ویژگی `DataSource` کنترل `computerBindingSource` چیست؟

کنجکاوی



۱۰ پرس‌وجوی قبلی را ویرایش کنید.

برای جست‌وجوی کلمات مشابه روی متد `FillByWordName` راست کلیک کنید و گزینه `Configure` را انتخاب کنید تا امکان ویرایش پرس‌وجوی قبلی فراهم شود. دستور را به صورت زیر ویرایش کنید:

```
SELECT WordCode, WordName, WordMeans FROM Computer
```

```
WHERE (WordName Like ?+'%')
```

برنامه را دوباره اجرا کنید. عملکرد دستور `Like` چیست؟

- قابلیت جست‌وجو براساس فیلد نام و کد کالا را برای سربرگ کالا ایجاد کنید.
- قابلیت جست‌وجو براساس نام و شماره همراه را برای جدول مخاطبان ایجاد کنید.

فعالیت منزل



کارگاه ۱۳ جست و جوی ترکیبی در جدول

معمولاً در اکثر برنامه‌ها برای ورود کاربر به برنامه باید هویت کاربر تأیید شود. این تأیید هویت معمولاً با نام کاربری و گذرواژه صورت می‌گیرد. به برنامه فروشگاه جدول جدیدی به نام جدول کاربران (Users) اضافه می‌کنیم.

Field Name	Data Type	
UserName	Short Text	نام کاربری
Password	Short Text	گذرواژه
UserFirstName	Short Text	نام کاربر
UserLastName	Short Text	نام خانوادگی کاربر
TypeOfAccess	Number	نوع دسترسی
LastEntryDate	Date/Time	تاریخ آخرین ورود کاربر
LastExitDate	Date/Time	تاریخ آخرین خروج کاربر
PicturePath	Short Text	مسیر تصویر کاربر

می‌خواهیم برنامه‌ای بنویسیم که هویت کاربر را بررسی کند و اگر مورد تأیید بود، کاربر را به فرم اصلی (MainForm) برنامه فروشگاه منتقل کند.

۱ پروژه فروشگاه را باز کنید.

شکل ۱۰- نمونه برگ ورود کاربر

۲ یک فرم جدید با نام **LoginForm** به پروژه اضافه کنید.

۳ فرم پروژه را طراحی کنید (شکل ۱۰).

دو کنترل کادرمتن با نام‌های `txtUserName` و `txtPassword` و یک دکمه با نام `btnLogin` به فرم اضافه کنید.

فیلم شماره ۱۱۱۱۳: اتصال به پایگاه داده

فیلم



پس از مشاهده فیلم component های مورد نیاز را برای استفاده از جدول users پایگاه داده MyShop به فرم اضافه کنید و نام آنها را به myShopDataSet, usersTableAdapter, usersBindingSource تغییر دهید.

۴ یک متد جدید با نام **FillByUserNameAndPass** برای **usersTableAdapter** ایجاد کنید.

در قسمت ایجاد پرس و جو در کادر Query Builder، در ستون Filter، روی فیلد UserName و Password علامت ؟ قرار دهید. پرس و جوی ایجاد شده به صورت زیر خواهد بود:

```
SELECT UserName, Password, UserFirstName, UserLastName, TypeOfAccess, LastEntryDate, LastExitDate
FROM Users WHERE (UserName = ?) AND (Password = ?)
```

۵ در **LoginForm** فیلدی برای اعتبارسنجی کاربر به نام **succeeded** تعریف کنید.

از این فیلد در فرم MainForm استفاده خواهد شد. MainForm مقدار فیلد را از LoginForm دریافت می کند.

```
public partial class LoginForm : Form
```

```
{
    public bool succeeded;
    متد رویداد کلیک دکمه btnLogin را بنویسید.
```

```
private void btnLogin_Click(object sender, EventArgs e)
```

```
{
    usersTableAdapter.FillByUserNameAndPass(myShopDataSet.Users,
    txtUserName.Text, txtPassword.Text);
    if (myShopDataSet.Users.Rows.Count > 0)
    {
        succeeded = true;
        this.Close();
    }
    else
        MessageBox.Show("خطا", "نام کاربری یا گذرواژه معتبر نیست", MessageBoxButtons.OK, MessageBoxIcon.Error);
}
```

کلاس myShopDataSet دارای جدول هایی است که ویژگی Rows.Count، تعداد سطرهای این جدول را مشخص می کند.

مفهوم شرط دستور if چیست؟

برنامه را اجرا کنید. عملکرد متد btnLogin_Click چیست؟

۷ متد رویداد **Load** فرم اصلی را بنویسید.

```
private void MainForm_Load(object sender, EventArgs e)
```

```
{
    LoginForm frmLogin= new LoginForm();
    frmLogin.ShowDialog();
    if (!frmLogin.succeeded)
    {
        this.Close();
    }
}
```


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.

فعالیت منزل



با بازکردن جدول کاربران در پایگاه داده، هر فردی می تواند نام کاربری و گذرواژه همه کاربران را ببیند و به برنامه فروشگاه وارد شود. برای امنیت بیشتر برنامه چه کار باید کرد؟

پژوهش



آنچه آموختم:

برداشت



۱.
۲.
۳.

کارگاه ۱۴ درج یک رکورد در جدول

می خواهیم برنامه ای بنویسیم که با کلیک روی دکمه «ثبت»، رکورد جدیدی به جدول «کالا» اضافه شود.

۱ پروژه فروشگاه را باز کنید.

۲ متد «اضافه کردن کالا» را ایجاد کنید.

برای ایجاد این متد روی فایل myShopDataSet.xsd دابل کلیک کنید. چه جدول هایی مشاهده می کنید؟ روی جدول Goods راست کلیک کنید. گزینه Query از منوی Add را انتخاب کنید و سپس گزینه Use SQL statement را انتخاب و دکمه Next را کلیک کنید. در ادامه گزینه INSERT را انتخاب و دکمه Next را کلیک کنید. در پنجره بعدی یک پرس و جوی اضافه کردن رکورد برای جدول Goods ایجاد شده که می توان با دکمه Query Builder آن را تغییر داد. علامت «سؤال» در پرس و جو نشانه ورودی های متد است. برای جدول کالا چهار ورودی داریم. بدون ایجاد تغییر دکمه Next را کلیک کنید.

ساختار دستور INSERT INTO در زبان SQL چگونه است؟

کنجکاوی



در کادر محاوره‌ای باز شده، باید نام متد پرس‌وجو را تعیین کرد که پیش‌فرض آن InsertQuery است. بدون تغییر دکمه Next سپس Finish را کلیک کنید.

۳ متد رویداد کلیک دکمه «ثبت» را بنویسید.

```
private void btnAddGoods_Click(object sender, EventArgs e)
{
    if (txtGoodsCode.TextLength > 0 && txtGoodsName.TextLength > 0
        && txtUnitPrice.TextLength > 0)
    {
        goodsTableAdapter.InsertQuery(int.Parse(txtGoodsCode.Text),
            txtGoodsName.Text, int.Parse(txtUnitPrice.Text), int.Parse(txtStock.Text));
        goodsTableAdapter.Fill(myShopDataSet.Goods);
        MessageBox.Show ("با موفقیت ثبت شد");
    }
}
```

در این کد فراخوانی متد Fill برای به‌روزرسانی داده‌های dgvGoods است. متد InsertQuery دارای چند آرگومان ورودی است؟ چرا برخی از ورودی‌ها عددی است؟ چرا از دستور if در برنامه استفاده شده است؟ عملکرد متد رویداد btnAddGoods_Click چیست؟ عملکرد دستور goodsTableAdapter.InsertQuery چیست؟ برنامه را اجرا کنید و یک رکورد جدید را در کادرهای متن وارد کنید. و سپس دکمه «ثبت» را کلیک کنید. آیا رکورد جدید در dgvGoods نمایش داده می‌شود؟ چرا؟ برنامه را ببندید و اجرا کنید. آیا رکورد اضافه‌شده در dgvGoods وجود دارد؟ جدول Goods پایگاه‌داده MyShop را باز کنید. آیا رکورد اضافه‌شده در جدول وجود دارد؟

۴ ویژگی Copy to Output Directory پایگاه‌داده را تنظیم کنید.

برای اینکه ثبت رکورد در جدول کالا به صورت دائمی باشد، پنجره ویژگی‌های پایگاه‌داده MyShop در Solution Explorer را باز کنید و ویژگی Copy to Output Directory را به Copy if newer تغییر دهید. دوباره برنامه را اجرا کنید و یک رکورد جدید را در کادرهای متن وارد کرده، دکمه «ثبت» را کلیک کنید و برنامه را ببندید. دوباره برنامه را اجرا کنید. آیا در جدول Goods این کالا اضافه شده است؟

- برای جدول مشتری برنامه درج رکورد را بنویسید.
- برای جدول مخاطبان برنامه ثبت مخاطب جدید را بنویسید.

فعالیت منزل



کارگاه ۱۵ حذف رکورد از جدول

می‌خواهیم برنامه‌ای بنویسیم که با کلیک روی دکمه «حذف» رکورد انتخابی از جدول کالا حذف شود.

۱ پروژه فروشگاه را باز کنید.

۲ متدی برای حذف رکورد ایجاد کنید.

همانند کارگاه «درج رکورد»، روی جدول راست کلیک کرده، گزینه Query منوی Add، سپس گزینه Use SQL statement را انتخاب کنید و سپس گزینه DELETE را انتخاب کرده، دکمه Next را کلیک کنید. در پنجره پیکربندی پرس‌وجو، یک پرس‌وجوی «حذف» رکورد ایجاد شده که باید ویرایش شود.

شکل کلی دستور SQL حذف رکورد از جدول

شرایط WHERE نام جدول DELETE FROM

مثال: برای حذف یک کالا با کد ۱ دستور حذف به صورت زیر است:

```
DELETE FROM Goods WHERE (GoodsCode = 1)
```

روی دکمه Query Builder کلیک کنید. در بخش Design شرط‌های اضافی را حذف کنید تا شرط زیر را داشته باشید:

```
DELETE FROM Goods WHERE (GoodsCode = ?)
```

پس از ویرایش پرس‌وجو، دکمه OK را کلیک کنید. بقیه مراحل را همانند کارگاه «درج» انجام دهید و نام پیش‌فرض را برای متد حذف بپذیرید.

۳ متد رویداد کلیک دکمه «حذف» را بنویسید.

```
private void btnDeleteGoods_Click(object sender, EventArgs e)
```

```
{
    if (txtGoodsCode.Text.Length > 0)
    {
        goodsTableAdapter.DeleteQuery(int.Parse(txtGoodsCode.Text));
        goodsTableAdapter.Fill(myShopDataSet.Goods);
        MessageBox.Show("با موفقیت حذف شد");
    }
}
```

برنامه را اجرا کرده، یک رکورد دلخواه را حذف کنید.

۴ دستور را تغییر دهید تا حذف براساس نام کالا باشد.

۵ دستور را برای حذف تمامی رکوردها تغییر دهید.



- برای جدول مشتری برنامه حذف رکورد را بنویسید.
- برای جدول مخاطبان برنامه حذف مخاطب را بنویسید.

کارگاه ۱۶ ویرایش رکورد در جدول

می‌خواهیم برنامه‌ای بنویسیم که با کلیک روی دکمه «ویرایش» رکورد انتخابی در جدول کالا ویرایش شود.

۱ پروژه فروشگاه را باز کنید.

۲ متدی برای ویرایش رکورد ایجاد کنید.

همانند کارگاه درج و حذف رکورد، روی جدول راست کلیک کرده، گزینه Query منوی Add، سپس گزینه UseSQLstatement را انتخاب کنید و در ادامه گزینه UPDATE را انتخاب کرده، دکمه Next را کلیک کنید. در پنجره پیکربندی پرس‌وجو، یک پرس‌وجوی «ویرایش» رکورد، ایجاد شده که باید ویرایش شود.

شکل کلی دستور SQL ویرایش رکورد

نام جدول UPDATE

SET

, مقدار ۱ = نام فیلد ۱

, مقدار ۲ = نام فیلد ۲

.

.

WHERE (شرایط)

با استفاده از دکمه Query Builder دستور SQL ایجادشده را طوری تغییر دهید که ویرایش رکورد براساس کد کالا انجام شود.

۳ متد رویداد کلیک دکمه «ویرایش» را بنویسید.

```
private void btnUpdateGoods_Click(object sender, EventArgs e)
```

```
{
```

```
if (txtGoodsCode.Text.Length > 0 && txtGoodsName.Text.Length > 0 &&
    txtUnitPrice.Text.Length > 0)
```

```
{
```

```
goodsTableAdapter.UpdateQuery(int.Parse(txtGoodsCode.Text),
    txtGoodsName.Text,int.Parse(txtUnitPrice.Text),
```

```
int.Parse(txtStock.Text), int.Parse(txtGoodsCode.Text));
goodsTableAdapter.Fill(myShopDataSet.Goods);
MessageBox.Show ("با موفقیت ویرایش شد");
}
}
```

برنامه را اجرا کنید و با انتخاب یک رکورد عمل ویرایش را انجام دهید.

- برنامه درج کالای جدید را طوری تغییر دهید که اگر کالا در جدول موجود بود، فقط موجودی کالا را افزایش دهد.
- برای جدول مشتری برنامه ویرایش رکورد را بنویسید.
- برای جدول مخاطبان برنامه ویرایش مخاطب را بنویسید.

فعالیت منزل



کارگاه ۱۷ درج و حذف سطر از DataGridView

می‌خواهیم در برگه خرید با انتخاب کالا، آن را در یک DataGridView اضافه کنیم و مبلغ نهایی برگه خرید را محاسبه کنیم. امکان حذف کالا از DataGridView وجود دارد.

فروشگاه

برگه خرید کالا مشتری گزارش

شماره برگه خرید: 1 تاریخ: 1396/11/29

کد مشتری: 1300

نام مشتری: رضا محمدی تلفن: نشانی:

کد کالا: 10 تعداد: 2

نام کالا: بنیر خامه ای قیمت: 3200

موجودی کالا: 15 درج در سبد خرید <<

مبلغ قابل پرداخت: 24900 تومان

سبد خرید:

کد کالا	نام کالا	تعداد	قیمت واحد	قیمت کل
1	شیر	3	2000	6000
11	کلوچه	10	500	5000
4	تخم مرغ	15	500	7500
10	بنیر خامه ...	2	3200	6400

پاک کردن چاپ ثبت برگه خرید

۱ پروژه فروشگاه را باز کنید.

۲ در سربرگ «برگه خرید» یک کنترل **DataGridView** به نام **dgvFactors** اضافه کنید.

این کنترل را همانند شکل سفارشی کنید. برای اضافه کردن ستون علامت حذف سطر، روی کنترل **dgvFactors** راست کلیک کنید. گزینه **Edit Column** را کلیک کرده، برای اضافه کردن ستون جدید روی دکمه **Add** کلیک کنید. از فهرست کشویی **Type**، عنصر **DataGridViewImageColumn** را انتخاب کنید و کادرمتن **HeaderText** را خالی کنید. با کلیک روی ستون ایجادشده در سمت راست ویژگی **Image** را به دلخواه تنظیم کنید.

ویژگی‌های **ImageLayout** و **FillWeight** را به دلخواه تغییر دهید. کاربرد این ویژگی‌ها چیست؟

۳ نام کادر متن **کد کالا** و **تعداد کالا** را **txtFGoodsCode** و **txtFAmount** قرار دهید.

۴ متد **FillByGoodsCode** را برای پرس‌وجو براساس **کد کالا** ایجاد کنید.

۵ متد رویداد **txtFGoodsCode_KeyDown** را بنویسید.

با ورود **کد کالا** در کادرمتن **txtFGoodsCode** و فشردن کلید **Enter** نام کالا در کنترل برجسب قرمز رنگ **lblGoodsName** و قیمت کالا در کنترل برجسب قرمز رنگ **lblPrice** قرار می‌گیرد.

```
private void txtFGoodsCode_KeyDown(object sender, KeyEventArgs e)
```

```
{  
    lblPrice.Text=" ";  
    lblGoodsName.Text = " ";  
    if (e.KeyCode==Keys.Enter)  
    if (txtFGoodsCode.Text.Length > 0)  
    {  
        goodsTableAdapter.FillByGoodsCode(myShopDataSet.Goods,  
            int.Parse(txtFGoodsCode.Text));  
        if (myShopDataSet.Goods.Rows.Count > 0)  
        {  
            lblGoodsName.Text =  
                myShopDataSet.Goods.Rows[0][ "GoodsName"].ToString();  
            lblPrice.Text =  
                myShopDataSet.Goods.Rows[0][ "UnitPrice"].ToString();  
        }  
    }  
}
```

نتیجه اجرای متد FillByGoodsCode جدول داده‌ای با حداقل یک رکورد است. مقادیر فیلدهای جدول Goods به صورت زیر است:

myShopDataSet.Goods.Rows [شماره سطر] ["نام فیلد"]

۶ متد رویداد کلیک دکمه انتقال به سربرگ کالا را بنویسید.

این دکمه (btnGoodsTab) که در کنار کادرمتن کد کالا قرار دارد، با فعال کردن سربرگ کالا، امکان جست‌وجو براساس نام در سربرگ کالا را فراهم می‌کند.

`private void btnGoodsTab_Click(object sender, EventArgs e)`

```
{
    tabShop.SelectedTab = tabPageGoods; // فعال شدن سربرگ کالا
    txtSGoodsName.Focus();
}
```

۷ متد رویداد کلیک دکمه «درج در برگه خرید» سربرگ کالا را بنویسید.

در سربرگ کالا با کلیک دکمه «درج در برگه خرید» اطلاعات مربوط به کالای انتخابی در سربرگ «برگه خرید» قرار می‌گیرد.

`private void btnInsertGoodsFactor_Click(object sender, EventArgs e)`

```
{
    tabShop.SelectedTab = tabPageFactor; // فعال شدن سربرگ برگه خرید
    txtFGoodsCode.Text = txtGoodsCode.Text;
    lblGoodsName.Text = txtGoodsName.Text;
    lblPrice.Text = txtUnitPrice.Text;
    txtFGoodsCode.Focus();
}
```

۸ متد رویداد کلیک دکمه جست‌وجوی مشتری (btnCustomerTab) در سربرگ «برگه خرید» را بنویسید.

برای تعیین مشخصات مشتری براساس نام مشتری، باید به سربرگ مشتری برویم. این رویداد سربرگ مشتری را فعال می‌کند.

۹ متد رویداد کلیک دکمه «درج در برگه خرید» در سربرگ مشتری را بنویسید.

در این رویداد باید مشخصات مشتری انتخاب شده به کنترل‌های سربرگ «برگه خرید» منتقل شود.

۱۰ متد رویداد کلیک دکمه «درج در سبد خرید» را بنویسید.

نام دکمه «درج در سبد خرید» را btnInsertGoods قرار دهید.

`private void btnInsertGoods_Click(object sender, EventArgs e)`

```
{
    int n=dgvFactors.Rows.Count - 1;
    dgvFactors.Rows.Add();
    dgvFactors.Rows[n].Cells[0].Value = txtFGoodsCode.Text;
```

```

dgvFactors.Rows[n].Cells[1].Value = lblGoodsName.Text;
dgvFactors.Rows[n].Cells[2].Value = txtFAmount.Text;
dgvFactors.Rows[n].Cells[3].Value = lblPrice.Text;
dgvFactors.Rows[n].Cells[4].Value =
    int.Parse(txtFAmount.Text) * int.Parse(lblPrice.Text);
}

```

چگونه می‌توان تعداد سطرهای یک DataGridView را تعیین کرد؟ برنامه را اجرا کنید و با جست‌وجوی یک کالا، آن را به جدول کالاها در برگه خرید اضافه کنید. عملکرد متد Add کنترل dgvFactors چیست؟

قیمت کل در کدام خانه dgvFactors قرار می‌گیرد و چگونه محاسبه می‌شود؟

۱۱ کد مرحله ۱۰ را تغییر دهید تا در صورت خالی بودن کادرمتن کد کالا، خطا رخ ندهد.

اگر کادرمتن txtFGoodsCode خالی باشد و دکمه «درج در سبد خرید» را کلیک کنیم، خطا رخ می‌دهد. با یک دستور if خطا را کنترل کنید.

۱۲ به متد رویداد کلیک دکمه «درج در سبد خرید» محاسبه مبلغ قابل پرداخت را اضافه کنید.

قطعه کدی بنویسید که مبلغ قابل پرداخت را محاسبه کرده، در کنترل برجسب lblAllCost قرار دهد.

۱۳ متد حذف سطر را برای کلیک روی علامت حذف سطر بنویسید.

برای حذف سطر انتخابی در جدول برگه خرید، متد رویداد CellContentClick کنترل dgvFactors را ایجاد کنید.

```

private void dgvFactors_CellContentClick(object sender,
DataGridViewCellEventArgs e)

```

```

{
    if (dgvFactors.Columns[e.ColumnIndex] is
        DataGridViewImageColumn && e.RowIndex >= 0)
        dgvFactors.Rows.RemoveAt(e.RowIndex);
}

```

ویژگی e.ColumnIndex شماره ستون و ویژگی e.RowIndex شماره سطر انتخابی را مشخص می‌کند. با کلیک روی هر خانه کنترل DataGridView رویداد CellContentClick رخ می‌دهد و اگر کلیک روی ستونی حاوی تصویر رخ داده باشد، با متد RemoveAt سطر جاری را حذف می‌کند.

کلمه کلیدی is در زبان سی شارپ برای ارزیابی سازگاری نوع در زمان اجرا به کار می‌رود. با استفاده از is می‌توان تعیین کرد که آیا یک نمونه شیء یا نتیجه یک عبارت می‌تواند به یک نوع مشخص تبدیل شود یا خیر؟

شکل کلی استفاده از is

```
expr is type
```

expr عبارتی است که می‌خواهیم امکان تبدیل شدن آن به یک نمونه از نوع type را بررسی کنیم. اگر expr غیر null و قابل تبدیل شدن به type باشد، حاصل عبارت true است، در غیر این صورت مقدار false برگردانده می‌شود.

به عنوان مثال، کد زیر تعیین می‌کند که آیا obj می‌تواند به یک نمونه از نوع TextBox تبدیل شود یا خیر.

```
if(obj is TextBox)
{
    ((TextBox)obj).Clear();
}
```

برای اینکه بتوان سطر جدیدی را با متد Add به DataGridView اضافه کرد این کنترل نباید مقید باشد.

یادداشت



کارگاه ۱۸ مدیریت خطا در برنامه پایگاه داده

می‌خواهیم برنامه فروشگاه را طوری تغییر دهیم که هنگام عملیات درج، حذف و ویرایش در سربرگ کالا، مدیریت خطا نیز صورت گیرد.

۱ برنامه فروشگاه را اجرا کنید.

۲ در سربرگ کالا، کالایی را ثبت کنید که کد کالای آن قبلاً ثبت شده باشد.

با درج کد تکراری چه اتفاقی رخ می‌دهد؟

در برخی مواقع با انجام عملیات مبنایی در جدول‌ها خطایی رخ می‌دهد که اگر مدیریت نشود، برنامه با ارسال یک پیام خطا متوقف می‌شود. برای جلوگیری از این امر باید دستوراتی که احتمال می‌رود در آنجا خطا رخ دهد را با ساختار دستوری try-catch مدیریت کرد.

برای اینکه بدانیم هنگام عملیات مبنایی چه خطاهایی ممکن است رخ دهد، جدول زیر را کامل کنید.

نام عمل	فهرست خطاها
درج	خطای ناشی از ساختار نادرست دستور درج -
حذف	
ویرایش	خطای ناشی از خالی ماندن فیلدی که نباید Null باشد-

برای مدیریت خطا در برنامه، دستور یا دستوراتی را که احتمال دارد، در هنگام اجرای آنها خطا رخ دهد، در بلاک try قرار دهید.

۳ روی دکمه ثبت در سربرگ «کالا» دابل کلیک کنید و کدها را به صورت زیر ویرایش کنید.

```
private void btnAddGoods_Click(object sender, EventArgs e)
{
    if (txtGoodsCode.Text.Length > 0 && txtGoodsName.Text.Length > 0 &&
        txtUnitPrice.Text.Length > 0)
    {
        try
        {
            goodsTableAdapter.InsertQuery(int.Parse(txtGoodsCode.Text),
```

```

txtGoodsName.Text, int.Parse(txtUnitPrice.Text),
int.Parse(txtStock.Text));
goodsTableAdapter.Fill(myShopDataSet.Goods);
MessageBox.Show ("با موفقیت ثبت شد");
}
catch
{
MessageBox.Show ("خطا در ثبت کالا");
}
}
}

```

برنامه را اجرا کنید و کالایی را ثبت کنید که کد کالای آن تکراری باشد، چه اتفاقی رخ می‌دهد؟
۴ مدیریت خطا را برای عملیات دیگری در سربرگ کالا انجام دهید.
 آیا دستور `if` در برنامه عمل کنترل خطا را انجام می‌دهد؟
 اگر کد کالا غیر عددی نوشته شود، آیا خطایی رخ می‌دهد؟

مدیریت خطا را برای کدهای سربرگ مشتری با استفاده از متد `TryParse` انجام دهید.

فعالیت منزل



کارگاه ۱۹ توسعه پروژه فروشگاه

فیلم شماره ۱۱۱۱۲: ذخیره برگه خرید در برنامه فروشگاه

فیلم



فعالیت کارگاهی



پس از مشاهده فیلم، برنامه جست‌وجوی مشتری و کالا و درج در سبد خرید و ثبت برگه خرید در سربرگ را بنویسید.

فعالیت منزل



- سربرگ جدیدی به نام کاربران به برنامه فروشگاه اضافه کنید که فقط مدیر فروشگاه به آن دسترسی داشته باشد و بتواند عملیات ایجاد، حذف و ویرایش کاربر را انجام دهد.



فاز ۵:

- در این پودمان آرایه‌های ایجاد شده برای نگهداری اطلاعات را حذف کرده و اطلاعات را از پایگاه داده دریافت کرده و در پایگاه داده ذخیره می‌کنیم.
- پروژه را به پایگاه داده کتابخانه که در پودمان ۱ ایجاد کردید، متصل کنید.
 - موارد زیر را به پروژه اضافه کنید:
 - ثبت زمان login کردن مدیر در یک پرونده به نام Log.txt همراه با مدیریت خطا.
 - اضافه کردن دکمه «دریافت پرونده» و کادر متنی برای دریافت مسیر ذخیره سازی پرونده
 - در صورت کلیک روی دکمه «دریافت پرونده» پرونده Log.txt را در مسیر تعیین شده در کادر متن ذخیره کند.
 - اضافه کردن دو سربرگ نمایش اعضا و نمایش کتاب‌ها که داده‌های جدول اعضا و کتاب‌ها در این سربرگ مشاهده شود.
 - فراهم کردن امکان حذف اعضا و ویرایش مشخصات آنها
 - فراهم کردن امکان حذف کتاب و ویرایش عنوان آنها

ارزشیابی مرحله ۴



مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره‌دهی)	نمره
انجام عملیات روی پایگاه داده	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۲۰ دقیقه	بالاتر از حد انتظار	نوشتن کد برای درج رکورد، ویرایش و حذف و جست‌وجوی رکورد - مدیریت و رفع خطای برنامه	۳
		در حد انتظار	نوشتن کد برای درج رکورد، ویرایش و حذف و جست‌وجوی رکورد	۲
		پایین‌تر از حد انتظار	نوشتن کد برای درج رکورد	۱
<p>معیار شایستگی انجام کار:</p> <p>کسب حداقل نمره ۲ از مراحل نوشتن در پرونده متنی و انجام عملیات روی پایگاه داده</p> <p>کسب حداقل نمره ۲ از بخش شایستگی‌های غیرفنی، ایمنی، بهداشت، توجهات زیست‌محیطی و نگرش</p> <p>کسب حداقل میانگین ۲ از مراحل کار</p>				

جدول ارزشیابی پایانی

شرح کار:

- ۱ نوشتن در پرونده متنی
- ۲ خواندن از پرونده متنی
- ۳ مقیدسازی واسط گرافیکی کاربری
- ۴ انجام عملیات روی پایگاه داده

استاندارد عملکرد:

با استفاده از دانش کلاس‌های آماده پرونده و پوشه و شناخت دستورات کار با پایگاه داده، از پرونده‌های متنی و پایگاه داده در برنامه استفاده کند.

شاخص‌ها:

شماره مرحله کار	شاخص‌های مرحله کار
۱	درج داده در پرونده متنی جدید یا موجود بر اساس نیاز- رفع خطای برنامه
۲	دسترسی به محتویات پرونده متنی بر اساس نیاز- رفع خطای برنامه
۳	ایجاد ارتباط واسط کاربری با پایگاه داده - مقیدسازی کنترل‌ها - رفع خطای برنامه
۴	نوشتن کد برای درج، ویرایش، حذف و جست‌وجوی رکورد بر اساس نیاز- مدیریت و رفع خطای برنامه

شرایط انجام کار و ابزار و تجهیزات:

مکان: کارگاه رایانه مطابق استاندارد تجهیزات هنرستان‌ها

تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است.

زمان: ۷۵ دقیقه (نوشتن در پرونده متنی ۱۵ دقیقه - خواندن از پرونده متنی ۲۰ دقیقه - مقیدسازی واسط گرافیکی کاربری ۲۰ دقیقه - انجام عملیات روی پایگاه داده ۲۰ دقیقه)

معیار شایستگی:

ردیف	مرحله کار	حداقل نمره قبولی از ۳	نمره هنرجو
۱	نوشتن در پرونده متنی	۲	
۲	خواندن از پرونده متنی	۱	
۳	مقیدسازی واسط گرافیکی کاربری	۱	
۴	انجام عملیات روی پایگاه داده	۲	
	شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش: تنظیم و اصلاح عملکردهای سیستم، عیب‌یابی مشکلات و نقصان سیستم - زبان فنی رعایت ارگونومی نگهداری داده به صورت الکترونیکی کاهش مصرف کاغذ و نوشتافزار طراحی واسط گرافیکی کاربرپسند - دقت در مدیریت خطاهای احتمالی		
	میانگین نمرات		
			*

* حداقل میانگین نمرات هنرجو برای قبولی و کسب شایستگی، ۲ است.

منابع فارسی

- ۱ شورای برنامه‌ریزی درسی رشته شبکه و نرم‌افزار رایانه (۱۳۹۴). برنامه درسی رشته شبکه و نرم‌افزار رایانه. سازمان پژوهش و برنامه‌ریزی آموزشی. دفتر برنامه‌ریزی و تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش.
- ۲ شورای برنامه‌ریزی درسی رشته شبکه و نرم‌افزار رایانه (۱۳۹۳). استاندارد ارزشیابی حرفه فناوری اطلاعات و ارتباطات. سازمان پژوهش و برنامه‌ریزی آموزشی. دفتر برنامه‌ریزی و تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش.
- ۳ شورای برنامه‌ریزی درسی رشته شبکه و نرم‌افزار رایانه (۱۳۹۲). استاندارد شایستگی حرفه فناوری اطلاعات و ارتباطات. سازمان پژوهش و برنامه‌ریزی آموزشی. دفتر برنامه‌ریزی و تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش.
- ۴ کربلایی، مجید، (۱۳۹۵)، برنامه‌سازی ۲ کد ۴۵۱، تهران: شرکت چاپ و نشر کتاب‌های درسی ایران
- ۵ کربلایی، مجید، (۱۳۹۵)، برنامه‌سازی ۳ کد ۳-۴۵۱/۵، تهران: شرکت چاپ و نشر کتاب‌های درسی ایران
- ۶ راعی، اکرم. (۱۳۹۵). بانک اطلاعاتی Access 2007 کد ۳۱۰۱۸۰. (چاپ اول). تهران: شرکت چاپ و نشر کتاب‌های درسی ایران
- ۷ کارکن، مهناز و دیگران، (۱۳۹۴)، کتاب معلم راهنمای تدریس برنامه‌سازی یک سی شارپ، تهران: شرکت چاپ و نشر کتاب‌های درسی ایران
- ۸ یمقانی، محمدرضا. (۱۳۹۴). بانک اطلاعاتی کد ۴۵۱/۷. (چاپ دهم). تهران: شرکت چاپ و نشر کتاب‌های درسی ایران
- ۹ ابراهیمی، یونس (۱۳۹۴)، آموزش C# به زبان ساده، تهران: انتشارات نبض دانش
- ۱۰ کربلایی، مجید، (۱۳۹۳)، برنامه‌سازی ۱ کد ۳۵۸/۷۰، تهران: شرکت چاپ و نشر کتاب‌های درسی ایران.
- ۱۱ جعفرنژاد قمی، عین‌الله. (۱۳۹۳). آموزش گام به گام برنامه‌سازی به زبان C#، بابل: انتشارات علوم رایانه
- ۱۲ جعفری، امین، (۱۳۹۲)، آموزش گام به گام C#.NET 2010، تهران: انتشارات صفار
- ۱۳ روحانی رانکوهی، محمد تقی. (۱۳۸۶). مفاهیم بنیادی پایگاه داده‌ها. (چاپ ششم). تهران: انتشارات جلوه

منابع انگلیسی

- ۱۴ Morgan, C. 2017. The History of Data Storage and Backup Part One: The Writing on the Wall. <http://www.storagecraft.com/blog/history-of-data-storage-and-backup-part-1-the-writing-on-the-wall>
- ۱۵ Date, C. J. (2016). The New Relational Database Dictionary. United States of America: O'Reilly Media, Inc
- ۱۶ Alexander, M., & Kusleika, D. (2016). Access 2016 Bible. United States of America: John Wiley & Sons, Inc
- ۱۷ Hernandez, M. J. (2013). Database Design for Mere Mortals. 3rd ed. United States of America: Addison-Wesley
- ۱۸ Gaddis Tony (2013) 'Starting out with Visual C# 2012 (3rd Edition)', Addison Wesley
- ۱۹ Sharma, S. (2012). A Tutorial Approach for Teaching Database Concepts. Master of

Science. North Dakota State University of Agriculture and Applied Science

୨୦ Saini, S. K. 2012. Importance of Means of Communication. <http://www.studyvillage.com/resources/ViewResource.aspx?ResourceId=3252>

୨୧ Agarwal, V. V. (2012). Beginning C# 5.0 databases. 2nd ed. (NY): Apress.

୨୨ Azuma, S., & Ltd. Trend-Pro Co. (2009). The Manga Guide to Databases. 1st ed. San Francisco: No Starch Press

୨୩ Churcher, C. (2007). Beginning Database Design from Novice to Professional. United States of America: Apress

୨୪ Takahashi, M., & Azuma, S., & Ltd. Trend-Pro Co. (2009). The Manga Guide to Databases. 1st ed. San Francisco: No Starch Press

୨୫ Stellman, A., & Greene, J. (2010). Head First C#. 2nd ed. (NY): O'Reilly

୨୬ Date, C. J. (2005). Database in Depth. United States of America: John Wiley & Sons, Inc

୨୭ Silberschatz, A., & Korth, H. F., & Sudarshan, S. (2011). Database System Concepts, 6th ed. United States of America: The McGraw-Hill

୨୮ Oppel, A. J. (2004). Databases Demystified, United States of America: The McGraw-Hill

୨୯ Barrows, A., & Levine Young, M., & Stockman, J. C. (2010). Access 2010 All-in-One For Dummies, United States of America: Wiley Publishing, Inc

୩୦ Elmasri, R., & Navathe, S. B. (2000). Fundamentals of Database Systems. 3rd ed. United States of America: Addison-Wesley

୩୧ Petersen, J. V. (2002). Absolute Beginner's Guide to Databases. 1st ed. United States of America: Que Publishing

୩୨ Vrat Agarwal, Vidya , Beginning C# 5.0 Databases,Apress23(Second Edition)

୩୩ The world's largest web developer site .(www.w3schools.com)

୩୪ C# Programming Guide. <https://msdn.microsoft.com/>

୩୫ All C# Language Topics. <https://stackoverflow.com/documentation/c%23/topics>



سازمان پژوهش و برنامه‌ریزی آموزشی جهت ایفای نقش خطیر خود در اجرای سند تحول بنیادین در آموزش و پرورش و برنامه درسی ملی جمهوری اسلامی ایران، مشارکت معلمان را به‌عنوان یک سیاست اجرایی مهم دنبال می‌کند. برای تحقق این امر در اقدامی نوآورانه سامانه تعاملی بر خط اعتبارسنجی کتاب‌های درسی راه‌اندازی شد تا با دریافت نظرات معلمان درباره کتاب‌های درسی نونگاشت، کتاب‌های درسی را در اولین سال چاپ، با کمترین اشکال به دانش‌آموزان و معلمان ارجمند تقدیم نماید. در انجام مطلوب این فرایند، همکاران گروه تحلیل محتوای آموزشی و پرورشی استان‌ها، گروه‌های آموزشی و دبیرخانه راهبری دروس و مدیریت محترم پروژه آقای محسن باهو نقش سازنده‌ای را بر عهده داشتند. ضمن ارج نهادن به تلاش تمامی این همکاران، اسامی دبیران و هنرآموزانی که تلاش مضاعفی را در این زمینه داشته و با ارائه نظرات خود سازمان را در بهبود محتوای این کتاب یاری کرده‌اند به شرح زیر اعلام می‌شود.

کتاب توسعه برنامه سازی و پایگاه داده‌ها با کد ۲۱۱۲۸۷

ردیف	نام و نام خانوادگی	استان محل خدمت	ردیف	نام و نام خانوادگی	استان محل خدمت
۱	پرستو کفیل	آذربایجان شرقی	۱۵	محمد مهدی آهنگری	شهرستان‌های تهران
۲	محمود باقری	گیلان	۱۶	مجید باقری	قزوین
۳	رضا روح الهی	کرمان	۱۷	محمدصادق شایسته منش	فارس
۴	مریم شفیعی	شهر تهران	۱۸	هوشنگ ابراهیمی	سیستان و بلوچستان
۵	بتول نخعی	کرمان	۱۹	حسین چاکرال‌حسینی	یزد
۶	صفیه غایی	اصفهان	۲۰	علی برزگری	خراسان جنوبی
۷	عابدین قربانی	قم	۲۱	محبوبه خوانی	سمنان
۸	زینب عباسی	قم	۲۲	بتول حجتی	گلستان
۹	زهرا مرادی نژاد	البرز	۲۳	سیده زهرا وزیر	یزد
۱۰	ایران حکمتیان	خوزستان	۲۴	پرهام محمدیان صبور	خراسان شمالی
۱۱	فاطمه بلالی دهکردی	چهارمحال و بختیاری	۲۵	تاجما روزخوش	کهگیلویه و بویراحمد
۱۲	فرحروز رزمی	اردبیل	۲۶	نازلی سیدانزایی نژاد	خراسان رضوی
۱۳	مهناز درویشی	البرز	۲۷	بهرروز نورایی نژاد	ایلام
۱۴	اسماعیل وفانور	هرمزگان	۲۸	فرحناز نادریگی	همدان

هنرآموزان محترم، هنرجویان عزیز و اولیای آنان می‌توانند نظریه‌های اصلاحی خود را درباره‌ی مطالب این کتاب از طریق نامه

به نشانی تهران - صندوق پستی ۴۸۷۴ / ۱۵۸۷۵ - گروه درسی مربوط و یا پیام‌نگار tvoccd@roshd.ir ارسال نمایند.

وب‌گاه: www.tvoccd.medu.ir

دفترتالیف کتاب‌های درسی فنی و حرفه‌ای و کارداش