

### آشنایی با چند کنترل جدید

در کتاب برنامه سازی ۲، به طور خاص روی پروژه های ویندوزی کار کردیم و با آنها واسط های کاربر گرافیکی تولید نمودیم. کنترل های زیادی هستند که می توانند این واسط های گرافیکی را از منظر کاربران محبوب تر و دلچسب تر نمایند. در این فصل با چند کنترل دیگر آشنا می شویم. در این گونه پروژه ها کنترل های ورود داده همیشه پر کاربرد و با اهمیت هستند. بنابراین در این فصل تمرین بیشتری با چند کنترل مهم ورود داده انجام می دهیم و با کنترل های جدیدی آشنا می شویم. در مثال های این فصل با فرم هایی کار می کنیم که داده ها را برای مقاصد مختلف از کاربر می گیرند.

پس از پایان این فصل انتظار می رود که فراگیر بتواند :

- ۱- فرم های ورود اطلاعات متفاوت تولید نماید.
- ۲- تفاوت و کاربرد کنترل دکمه انتخاب و کادر انتخاب را توضیح دهد و آن ها را در برنامه خود به کار بندد.
- ۳- کادر محاوره ای پیام را با دکمه های متنوع ایجاد نماید.
- ۴- کنترل NumericUpDown را در برنامه های خود استفاده کند.

## ۱-۱- ایجاد فرم ورود اطلاعات فردی

**مثال ۱-۱-۱-** برنامه‌ای بنویسید که بتواند یک فرم ورود اطلاعات مانند شکل ۱-۱ ایجاد نماید.

شکل ۱-۱-۱- فرم ورود اطلاعات

## الگوریتم یا روش انجام کار

**۱- / ایجاد یک پروژه ویندوزی:** در برنامه VS یک پروژه جدید از نوع Windows Form

Application با نام پروژه Entry Information در مسیر مشخصی بسازید.

**۲- / درج کنترل‌ها:** ساخت یک فرم ورود اطلاعات مانند شکل ۱-۱ ساده است و پیچیدگی

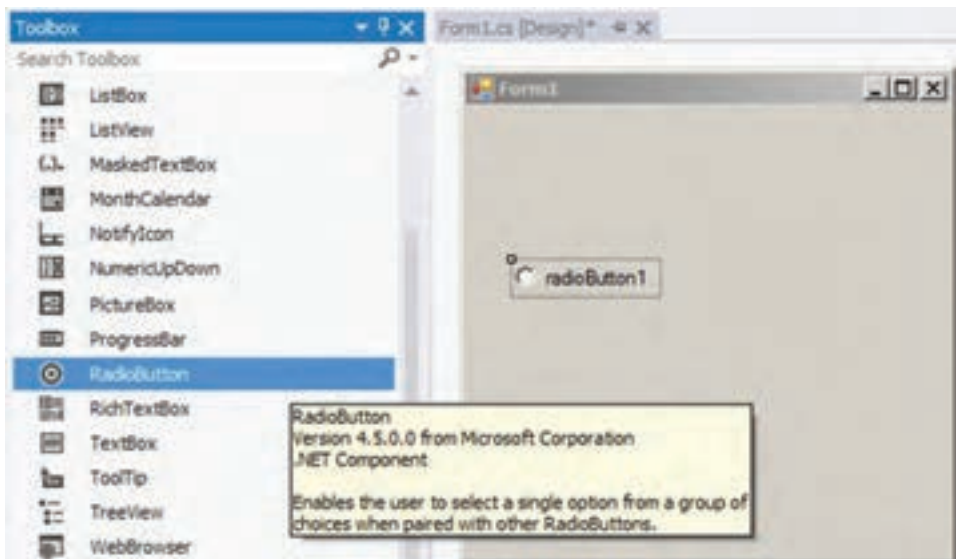
خاصی ندارد. تنها کنترلی که در فرم وجود دارد، که قبلاً با آن برخورد نکرده‌ایم، قسمت انتخاب

جنسیت است که می‌توان با کلیک بر روی دایره‌ها (دکمه‌ها)، جنسیت مرد یا زن را انتخاب کرد. این

دکمه‌ها، به دکمه رادیویی<sup>۱</sup> معروف هستند. چرا؟

در کنترل دکمه رادیویی، فقط یک گزینه حق انتخاب دارد.

در روی فرم‌ها نیز معمولاً برای انتخاب یک مورد از بین دو یا چند مورد، مانند انتخاب جنسیت (مرد<sup>۱</sup>، زن<sup>۲</sup>)، انتخاب رنگ زمینه (سبز، آبی، قرمز) و یا انتخاب نحوه پرداخت (نقدی<sup>۳</sup>، کارت<sup>۴</sup>)، چک از دکمه‌های رادیویی استفاده می‌کنیم. مثلاً برای انتخاب جنسیت نیاز به دو دکمه رادیویی است. برای اضافه کردن کنترل دکمه رادیویی، به جعبه ابزار Common Controls مراجعه و کنترل RadioButton را به فرم اضافه می‌کنیم. (شکل ۱-۲)



شکل ۱-۲

- ۱ \_ Male
- ۲ \_ Female
- ۳ \_ Cash
- ۴ \_ Debit Card

### ۳- تعیین ویژگی های کنترل ها

جدول ۱-۱- ویژگی های برجسب

Label						
ویژگی	مقدار	مقدار	مقدار	مقدار	مقدار	مقدار
Name	nameText	familyText	fatherText	telText	mobileText	emailText
Text	نام	نام خانوادگی	نام پدر	تلفن ثابت	تلفن همراه	پست الکترونیکی

جدول ۱-۲- ویژگی های کادر انتخاب

CheckBox	
ویژگی	مقدار
Name	acceptCheck
Checked	False
RightToLeft	Yes

جدول ۱-۳- ویژگی های دکمه انتخاب

RadioButton		
ویژگی	مقدار	مقدار
Name	maleRadio	femaleRadio
Checked	False	False
RightToLeft	Yes	Yes

برای دکمه های رادیویی و دکمه های انتخاب فارسی، ویژگی RightToLeft را Yes انتخاب می کنیم تا علامت دایره و مربع در سمت راست نوشته قرار گیرد.

در صورتی که بخواهید دو دسته دکمه رادیویی مستقل از یکدیگر، روی فرم داشته باشید، باید هر دسته از دکمه‌ها را در داخل یک قاب مانند (GroupBox) قرار دهید.

**سؤال؟** برای اینکه جنسیت مرد، انتخابِ پیش فرض باشد، چه ویژگی‌ای مقاداردهی می‌شود؟

**۴- نوشتن کد برای دکمه «ذخیره»:** متد EH را برای دکمه ذخیره با رعایت شرایط زیر، می‌نویسیم:

الف) تمام اطلاعات خواسته شده که با علامت ستاره قرمز رنگ مشخص شده است اجباری است و کاربر باید وارد کند (کادرهای متنی نباید خالی باشد).

ب) با کلیک بر روی دکمه «ذخیره»، علاوه بر رعایت بند الف، باید کادر تیک مربوط به «قبول شرایط عضویت در سیستم» نیز تیک خورده باشد. اگر هر دو شرط برقرار بود، پیام «اطلاعات عضو جدید در سیستم ثبت شد» نشان داده شود و در غیر این صورت پیام خطای مناسبی ظاهر شود.

با استفاده از دستور if خالی نبودن کادرهای متنی و علامت دار بودن checkbox و انتخاب شدن یکی از RadioButtonها بررسی می‌شود.

```
if (name Text. Text != " " && family Text. Text != " "
    && father Text. Text != " " && telText. Text != " "
    && mobileText. Text != " " && emailText. Text != " "
    && acceptCheck. Checked
    && (maleRadio.Checked || femaleRadio.Checked))
{
}
}
```

بنابراین متد EH مربوط به دکمه «ذخیره» می‌تواند چنین باشد:

```

Private void saveButton_Click(object sender, EventArgs e)
{
    if (nameText.Text != "" && familyText.Text != ""
        && fatherText.Text != "" && telText.Text != ""
        && mobileText.Text != "" && emailText.Text != ""
        && acceptCheck.Checked
        && (maleRadio.Checked || femaleRadio.Checked))
    {
        string message = "اطلاعات عضو جدید در سیستم ثبت شد" + "\n نام : " + nameText.Text
            + "\n نام خانوادگی : " + familyText.Text + "\n نام پدر : " + fatherText.Text
            + "\n تلفن همراه : " + telText.Text + "\n تلفن ثابت : " + telText.Text
            + "\n پست الکترونیک : " + emailText.Text + "\n جنسیت : ";
        if (maleRadio.Checked)
            message += "مرد";
        else
            message += "زن";
        MessageBox.Show (message, "ورود اطلاعات کاربر");
    }
    else
    {
        MessageBox.Show ("ورود اطلاعات کاربر", "اطلاعات باید به طور کامل وارد شود");
    }
}

```

۵- نوشتن کد برای دکمه «انصراف»: متد EH را برای دکمه انصراف با رعایت شرایط

زیر، می نویسیم:

الف) با کلیک بر روی دکمه انصراف، یک پیام مبنی بر «اطمینان از انصراف» ظاهر شود

(شکل ۱-۳).



شکل ۱-۳- پیام هشدار خروج از برنامه

ب) اگر کاربر دکمه YES را انتخاب کرد فرم ورود اطلاعات بسته شود و اگر دکمه NO را انتخاب کرد، اتفاقی نیفتد.

بنابراین، متد EH مربوط به دکمه "انصراف" می تواند چنین باشد :

```
Private void cancelButton_Click(object sender, EventArgs e)
```

```
{  
    string message = "آیا قصد دارید از برنامه خارج شوید؟";  
    string caption = "هشدار";  
    DialogResult result = MessageBox.Show(message, caption,  
    MessageBoxButtons.YesNo);  
    if (result == DialogResult.Yes)  
        this.Close();  
}
```

از نوع داده شمارشی DialogResult برای بررسی مقدار برگشتی دکمه فشرده شده در کادر  
محواره ای استفاده می شود. مقادیر برگشتی DialogResult را در جدول ۴-۱ ببینید.

جدول ۴-۱ مقادیر DialogResult

مقدار برگشتی	شرح
None	از کادر محاوره ای چیزی برگشته است.
OK	دکمه ok فشرده شده است.
Cancel	دکمه Cancel فشرده شده است.
Abort	دکمه Abort فشرده شده است.
Retry	دکمه Retry فشرده شده است.
Ignore	دکمه Ignore فشرده شده است.
Yes	دکمه Yes فشرده شده است.
No	دکمه No فشرده شده است.

کلمه کلیدی this به نمونه ای از کلاس Form۱ اشاره می کند و منظور کلاسی است که رویداد  
برای آن نوشته شده است و متد Close برای بستن فرم استفاده می شود.

## توسعه و بهبود برنامه

تمرین: در رویداد دکمه ذخیره اطلاعات با توجه به خالی بودن هر یک از کادرهای متنی پیامی متناسب با آن نمایش داده شود. برای نمونه در صورت خالی بودن کادر متنی nameText پیام «لطفا نام را وارد کنید» نمایش داده شود.

## ۱-۲- کنترل عددی افزایشی – کاهش

علاوه بر کادر متنی، می‌توانید از کنترل عددی افزایشی – کاهش (NumericUpDown) برای دریافت داده‌های عددی، در یک محدوده مشخص، استفاده نمایید. در مثال زیر کنترل مذکور در قالب یک برنامه به نام سنجش مهارت عمل جمع، به کار رفته است.

### کار در کارگاه ۲: سنجش مهارت عمل جمع

**مثال ۱-۲:** برنامه‌ای بنویسید که برای سنجش مهارت عمل جمع یک دانش‌آموز، مورد استفاده قرار گیرد. در این برنامه ابتدا فرمی مانند شکل ۱-۴ ظاهر می‌شود.



شکل ۱-۴

کاربر با کلیک بر روی دکمه «شروع»، باید به سؤالی که ظاهر می‌شود، در مدت زمان محدودی، پاسخ دهد. اعدادی که باید جمع زده شوند در محدوده یک تا ۵۰ به طور تصادفی انتخاب می‌گردند.



مثلاً در شکل ۵-۱ اعداد ۴۸ و ۳ باید با یکدیگر جمع زده شوند.



شکل ۵-۱- فرم برنامه سنجش مهارت جمع

کاربر می تواند عدد مورد نظر خود را در قسمت سفید رنگ تایپ کند و یا با استفاده از فلش های بالا و پایین، قادر است پاسخ خود را وارد کند. اگر کاربر بتواند در مدت تعیین شده پاسخ صحیح دهد، پیامی مانند شکل ۶-۱ ظاهر می شود :



شکل ۶-۱- نمایش پیام برای پاسخ درست

چنانچه کاربر نتواند در مدت زمان تعیین شده پاسخ صحیح دهد، پیامی مانند شکل ۱-۷ ظاهر

می شود :



شکل ۱-۷- نمایش پیام برای پاسخ اشتباه

پس از پایان مدت زمان تعیین شده و نمایش پیام‌های مربوطه، کاربر می‌تواند با کلیک بر روی دکمه «شروع» سؤال دیگری را مشاهده کند و در مدت زمان تعیین شده پاسخ دهد.

الگوریتم یا روش انجام کار : مراحل زیر را برای انجام این مثال دنبال می‌کنیم.

**۱- ایجاد یک پروژه ویندوزی :** در برنامه VS یک پروژه جدید از نوع Windows Form

Application با نام پروژه SimpleQuiz در مسیر مشخصی بسازید.

**۲- تعیین ویژگی‌های فرم :** یک فرم ورود اطلاعات مانند شکل ۱-۷ ساده است و پیچیدگی

خاصی ندارد. ویژگی‌های فرم را مانند جدول زیر تعیین کنید :

جدول ۱-۵- ویژگی‌های فرم

Form	
ویژگی	مقدار
Name	Form1
Text	سنجش مهارت جمع

### ۳- ایجاد کنترل‌های برچسب و تعیین ویژگی‌های آنها: در این پروژه نیاز به شش کنترل

برچسب داریم.

ابتدا در روی فرم، دو برچسب برای نمایش دو عددی که باید جمع زده شود، قرار دهید.

جدول ۷-۱- ویژگی‌های کنترل برچسب دوم

Label		
ویژگی	مقدار	
Name	leftLabel	
Text	?	
Font	Arial Size: 16 Bold	
Location	X	10
	Y	110

جدول ۶-۱- ویژگی‌های کنترل برچسب اول

Label		
ویژگی	مقدار	
Name	rightLabel	
Text	?	
Font	Arial Size: 16 Bold	
Location	X	120
	Y	110

#### نکته

اگر به جدول بالا دقت کنید مشاهده می‌کنید که اغلب ویژگی‌ها در دو برچسب مانند یکدیگر است. در این‌گونه موارد، می‌توانید یک برچسب را در روی فرم قرار دهید و ویژگی‌های آن را مطابق جدول تنظیم کنید، سپس آن را انتخاب کرده و عمل کپی و الحاق روی فرم را انجام دهید. در انتهای کار، فقط لازم است برخی از ویژگی‌های آن را مطابق با جدول تغییر دهید. استفاده از این روش، سرعت ساخت فرم را افزایش می‌دهد.

سپس در روی فرم، دو برچسب دیگر، برای نمایش نمادهای + و = قرار دهید.

جدول ۸-۱- ویژگی های کنترل برچسب سوم

Label		
ویژگی	مقدار	
Name	equalLabel	
Text	=	
Font	Arial Size: 16 Bold	
Location	X	170
	Y	110

جدول ۹-۱- ویژگی های کنترل برچسب چهارم

Label		
ویژگی	مقدار	
Name	sumLabel	
Text	+	
Font	Arial Size: 16 Bold	
Location	X	70
	Y	110

در روی فرم، دو برچسب دیگر برای نمایش پیام مربوط به زمان باقیمانده و اندازه ثانیه قرار دهید.

جدول ۱۰-۱- ویژگی های کنترل برچسب پنجم

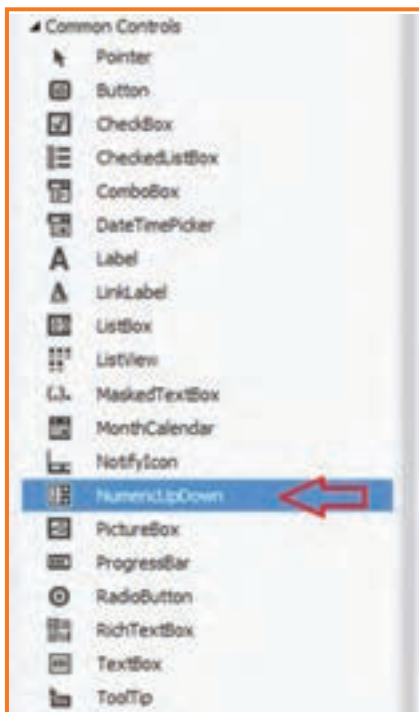
Label		
ویژگی	مقدار	
Name	timeLabel	
Text		
Visible	False	
Font	Arial Size: 16 Bold	
RightToLeft	No	
Location	X	60
	Y	25

جدول ۱۱-۱- ویژگی های کنترل برچسب ششم

Label		
ویژگی	مقدار	
Name	messageLabel	
Text	مدت زمان باقیمانده	
Visible	False	
Font	Arial Size: 16 Bold	
RightToLeft	Yes	
Location	X	120
	Y	30

#### ۴- ایجاد کنترل عددی افزایشی-کاهشی:

برای این منظور به جعبه ابزار Common Controls رفته و ابزار NumericUpDown را انتخاب و روی فرم قرار دهید.



شکل ۸-۱

#### ۵- تعیین ویژگی‌های کنترل عددی افزایشی-کاهشی: ویژگی‌های این کنترل را مطابق

با جدول زیر تنظیم کنید.

جدول ۱۲-۱- ویژگی‌های کنترل عددی افزایشی-کاهشی

NumericUpDown	
ویژگی	مقدار
Name	numericUpDown1
Maximum	100
Minimum	0
Visible	False
Font	Arial Size: 16 Bold

حداقل و حداکثر مقداری که می‌توان در کنترل عددی وارد کرد، به وسیله دو ویژگی Minimum و Maximum معین می‌شود. با توجه به اینکه، دو عدد تصادفی بین ۰ تا ۵۰ ظاهر می‌شوند، بنابراین مجموع آنها، عددی بین ۰ تا ۱۰۰ است. از این رو مقدار ویژگی‌های Minimum و Maximum کنترل عددی را به ترتیب برابر صفر و ۱۰۰ مقداردهی می‌کنیم.

**۶- ایجاد کنترل دکمه و تعیین ویژگی‌های آن:** یک دکمه نیز به فرم اضافه کنید.

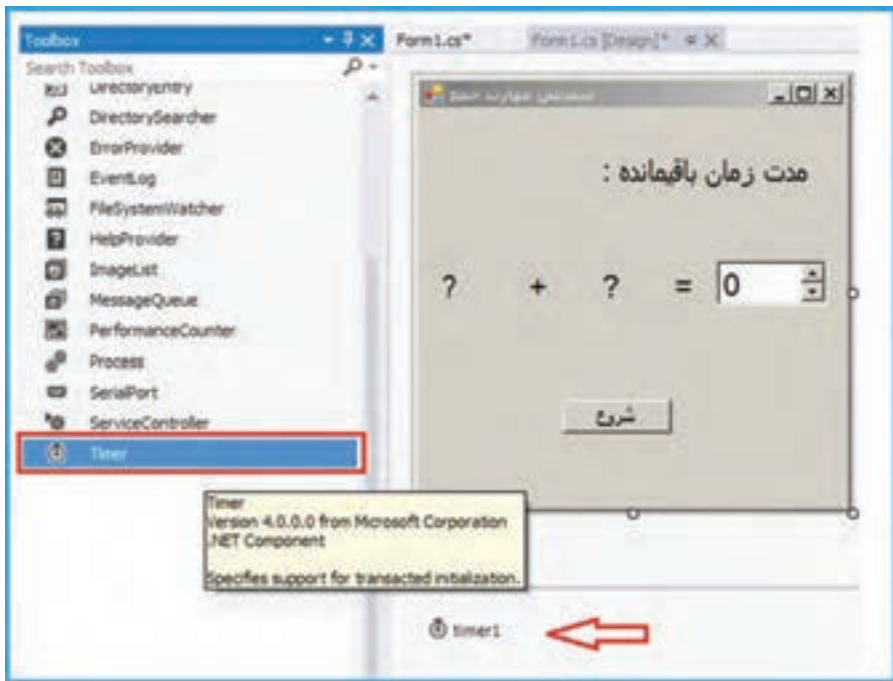
جدول ۱۳-۱ ویژگی‌های دکمه شروع

Button		
ویژگی	مقدار	
Name	start	
Text	شروع	
Location	X	وسط صفحه قرار گیرد
	Y	200

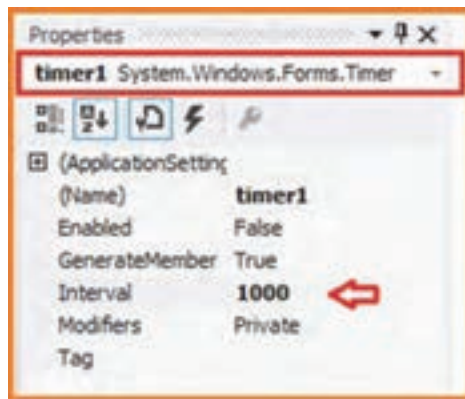
**۷- ایجاد زمان‌سنج (Timer):** با توجه به اینکه برای پاسخ به سؤال، محدودیت زمانی داریم و باید در ظرف مدت ۵ ثانیه به سؤال پاسخ داده شود، لذا از یک زمانی استفاده می‌کنیم. برای این منظور به سراغ پنجره جعبه ابزار رفته و در جعبه ابزار Component، یک شیء Timer را به برنامه اضافه کنید. توجه داشته باشید که این شیء در روی فرم نشان داده نمی‌شود بلکه در زیر فرم قرار می‌گیرد (شکل ۱-۹).

**۸- تعیین ویژگی کنترل زمانی:** با اضافه کردن شیء تایمر در برنامه، ویژگی‌های آن در پنجره Properties نشان داده می‌شود. ویژگی Interval را برابر ۱۰۰۰ قرار دهید تا هر ۱ ثانیه معادل ۱۰۰۰ میلی ثانیه، یک رویداد زمانی رخ دهد که در آن صورت متد EH مربوط به Timer به طور خودکار اجرا می‌شود (شکل ۱-۱۰).

**سؤال:** ویژگی Enabled برای زمان‌سنج به طور پیش فرض چه مقداری است؟



شکل ۹-۱- اضافه کردن زمان سنج



شکل ۱۰-۱- تعیین بازه زمانی زمان سنج

**۹- تعریف متغیرهای مورد نیاز برنامه:** قبل از تولید اعداد، دو متغیر برای نگهداری اعداد تولید شده تعریف می‌کنیم. این دو متغیر را در خط‌های ابتدایی کلاس Form1 ایجاد می‌کنیم. همچنین در همین مکان یک متغیر با نام timeLeft برای نگهداری زمان باقیمانده تعریف می‌کنیم. با

این کار در همه متدها و رویدادهای کلاس می‌توانیم از این سه متغیر استفاده کنیم.

```
public partial class Form1 : Form
```

```
{  
    int leftNumber;  
    int rightNumber;  
    int timeLeft;  
}
```

۱۰- نوشتن رویداد *Click* برای دکمه: اکنون نوبت به نوشتن متد EH رویداد کلیک دکمه «شروع» می‌رسد. کارهایی که باید در این متد انجام شود عبارت‌اند از:

الف) تولید دو عدد تصادفی در محدوده یک تا پنجاه: برای تولید اعداد تصادفی صحیح، از کلاس Random استفاده می‌شود. برای استفاده از آن، کد زیر را درون دکمه شروع می‌نویسیم:

```
Random randomNumber = new Random();
```

اکنون اگر از دستور `randomNumber.Next()` استفاده کنیم یک عدد صحیح تصادفی خواهیم داشت. اما می‌خواهیم دامنه اعدادی که بدست می‌آوریم محدود باشد. یک راه این است که از متد `Random.Next(maxValue)` استفاده کنیم. این متد عددی بزرگ‌تر یا مساوی صفر و کوچک‌تر از `maxValue` به ما می‌دهد. توجه داشته باشید که خود عدد `maxValue` جزو محدوده نیست. بنابراین اگر از دستور `randomNumber.Next(۵۰)` استفاده کنیم عددی بین ۰ تا ۴۹ خواهیم داشت. پس با انتخاب حد بالایی ۵۱، به عددی بین ۱ تا ۵۰ دست پیدا می‌کنیم. اما روش دیگر آن است که از متد `Random.Next(minValue, maxValue)` استفاده کنیم. این متد، محدوده مورد نظر را مشخص می‌کند پس کافی است دستور `randomNumber.Next(1, 51)` را برای رسیدن به مطلوب بنویسیم.

بنابراین دستورهای ما به این صورت نوشته می‌شوند:

```
leftNumber = randomNumber.Next(1, 51);  
rightNumber = randomNumber.Next(1, 51);
```

ب) قرار دادن اعداد تصادفی تولید شده در کنترل‌های برچسب:

```
leftLabel.Text = leftNumber.ToString ();  
rightLabel.Text = rightNumber.ToString ();
```



سؤال: چرا از متد ToString استفاده شده است؟

ج) مقداردهی زمان باقی مانده و نمایش پیام :

```
messageLabel.Visible = true;  
timeLabel.Text = "5";  
timeLeft = 5;  
timeLabel.Visible = true;
```

د) فعال کردن زمان سنج : زمان سنج را برای شروع سنجش، فعال کرده و دکمه «شروع» را برای جلوگیری از رخ دادن رویداد کلیک دکمه تا قبل از پایان این مرحله غیرفعال می کنیم :

```
timer1.Enabled = true;  
start.Enabled = false;
```

ه) نمایش و مقدار دهی کنترل NumericUpDown1 :

```
numericUpDown1.Value = 0;  
numericUpDown1.Visible = true;
```

متد کامل EH دکمه «شروع» چنین خواهد بود :

```
private void start_Click(object sender, EventArgs e)  
{  
    Random randomNumber = new Random();  
    leftNumber = randomNumber.Next(1, 51);  
    rightNumber = randomNumber.Next(1, 51);  
    leftLabel.Text = leftNumber.ToString();  
    rightLabel.Text = rightNumber.ToString();  
    messageLabel.Visible = true;  
    timeLabel.Text = "5";  
    timeLeft = 5;  
    timeLabel.Visible = true;  
    numericUpDown1.Value = 0;
```

```

numericUpDown1.Visible = true;
timer1.Enabled = true;
start.Enabled = false;
}

```

## ۱۱- نوشتن رویداد Tick برای کنترل زمانی: اکنون باید متد EH رویداد زمانی را

بنویسیم. متد مربوطه باید عملیات زیر را انجام دهد:

الف) درستی پاسخ وارد شده به وسیله کاربر را بررسی کند.

ب) اگر پاسخ کاربر صحیح باشد، پیامی نمایش دهد که «پاسخ شما صحیح است» و سپس رویداد زمانی را از کار باندازد و دکمه «شروع» را برای سنجش بعدی فعال کند.

ج) اگر پاسخ کاربر نادرست باشد، آنگاه زمان باقیمانده را بررسی کند اگر زمان به پایان رسیده بود، آنگاه پیامی نمایش دهد که «پاسخ شما اشتباه است» و سپس رویداد زمانی را از کار باندازد و دکمه «شروع» را برای سنجش بعدی فعال کند.

د) اگر پاسخ کاربر نادرست باشد، ولی مدت پاسخ‌گویی به سؤال تمام نشده است، آنگاه یک ثانیه از زمان باقیمانده کم کند و مدت باقیمانده را روی صفحه نمایش دهد.

```
private void timer1_Tick(object sender, EventArgs e)
```

```

{
    if (leftNumber + rightNumber == numericUpDown1.Value)
    {
        timer1.Enabled = false;
        timeLeft = 5;
        MessageBox.Show(
            "سنجش مهارت عمل جمع", "آفرین. شما به این سؤال درست پاسخ دادید");
        start.Enabled = true;
    }
    if (timeLeft > 0)
    {
        timeLeft -- ;
    }
}

```

```

timeLabel.Text = timeLeft.ToString();
}
else
{
    timer1.Enabled = false;
    MessageBox.Show(
        "سنجش مهارت عمل جمع", "پاسخ اشتباه! متأسفانه وقت شما برای پاسخ گویی به این سؤال تمام شد");
    start.Enabled = true;
}
}
}

```

برای انجام عملیات بالا، بر روی زمان‌سنج، دو بار کلیک کرده تا متد EH مربوطه ظاهر شود و سپس در داخل آن، دستورات زیر را تایپ می‌کنیم.

### ۳-۱- کادر محاوره‌ای پیام

برای بررسی پاسخ کاربر، مجموع دو عدد با پاسخ وارد شده به وسیله کاربر، مقایسه می‌شود، اگر پاسخ کاربر صحیح باشد، پیام درست بودن پاسخ نمایش داده شده، زمان‌سنج غیرفعال می‌شود و دکمه شروع برای عملیات مجدد فعال می‌گردد. در صورت اشتباه بودن پاسخ کاربر از زمان باقی مانده یک ثانیه کم شده و برحسب‌ها متناسب با آن به روز می‌شوند و در صورت پایان یافتن زمان کاربر پیام مناسب نمایش داده شده، تایمر غیرفعال می‌شود و دکمه شروع برای عملیات مجدد فعال می‌گردد.

در این برنامه برای نمایش پیام از کادر محاوره‌ای MessageBox استفاده شده است. این کادر محاوره‌ای، با استفاده از متد Show() پیامی را در یک پنجره جداگانه نمایش می‌دهد و منتظر می‌شود

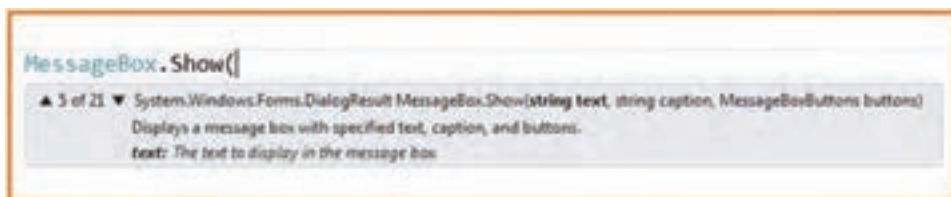
تا کاربر بر روی دکمه‌ای کلیک کند. شکل ۱۱-۱ پیام مربوط به پاسخ صحیح کاربر را نشان می‌دهد. به وسیله پارامترهایی که به متد Show() ارسال می‌کنیم، می‌توانیم پیام، عنوان پنجره و دکمه‌های مورد نظر در کادر محاوره‌ای را تعیین کنیم.



شکل ۱۱-۱- کادر محاوره‌ای پیام

;(دکمه پیش فرض. آیکون. دکمه‌ها. عنوان پنجره. پیام.) `MessageBox.Show()`

متد `Show()` دارای شکل‌های مختلفی است که شکل ۱۲-۱ یک نمونه استفاده از متد `Show()` با سه پارامتر را نشان می‌دهد.



شکل ۱۲-۱

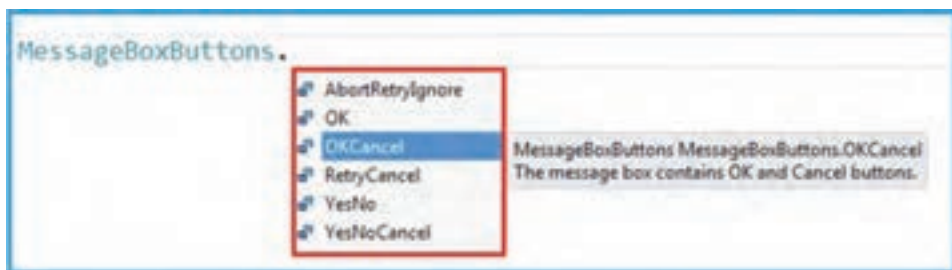
اگر فقط یک پارامتر رشته‌ای به متد `Show()` ارسال کنید، این رشته در پنجره‌ای بدون عنوان و تنها با یک دکمه "OK" نمایش داده می‌شود.

دکمه‌های پنجره پیام `MessageBox` را به وسیله نوع داده شمارشی `MessageBoxButtons` تعیین می‌کنیم که در فضای نام `System.Windows.Forms` قرار دارد و اعضای آن در جدول ۱۴-۱ آمده است.

جدول ۱۴-۱

نام اعضا	شرح
AbortRetryIgnore	کادر پیام شامل دکمه‌های Abort, Retry و Ignore است.
OK	کادر پیام شامل دکمه OK است.
OKCancel	کادر پیام شامل دکمه‌های OK و Cancel است.
RetryCancel	کادر پیام شامل دکمه‌های Retry و Cancel است.
YesNo	کادر پیام شامل دکمه‌های Yes و No است.
YesNoCancel	کادر پیام شامل دکمه‌های Yes, No و Cancel است.

شکل ۱۳-۱ اعضای این داده شمارشی را نشان می‌دهد.



شکل ۱۳-۱

مثلاً دستور زیر سبب نمایش کادر محاوره‌ای با دو دکمه "yes" و "No" می‌شود.

`MessageBox.Show("آیا می‌خواهید ادامه دهید؟", MessageBoxButtons.YesNo);`

اکنون برنامه را اجرا و خطاهای زمان اجرای آن را بررسی کنید.

### توسعه و بهبود برنامه

تمرین : در این قسمت می‌خواهیم برنامه سنجش عمل جمع را توسعه دهیم. برای این منظور عملیات زیر را دنبال می‌کنیم :

الف) توسعه برنامه با نمایش پاسخ صحیح در کنترل `NumericUpDown` در پایان سنجش :  
دستوری به متد `EH` رویداد زمانی اضافه کنید که در صورت پایان مدت زمان سنجش، پاسخ صحیح را نیز در کنترل عددی افزایشی - کاهش‌ی نمایش دهد.

ب) توسعه برنامه با دریافت نام کاربر :

۱- در پنجره طراحی فرم، یک کادر متنی (`Text Box`) و یک برچسب اضافه نمایید تا کاربر نام خود را قبل از شروع سنجش وارد نماید.

۲- در دستور `MessageBox` که پیام مربوط به پاسخ درست و همچنین پیام پاسخ نادرست را نشان می‌دهد، نام کاربر نیز اضافه شود.

ج) توسعه برنامه با اضافه کردن عملیات دیگر به فرم :

علاوه بر سنجش عمل جمع، عملیات دیگر نظیر تفریق، ضرب و تقسیم نیز اضافه کنید.  
راهنمایی : برای عمل تفریق نیز چهار برچسب و یک کنترل عددی افزایشی - کاهش‌ی به فرم

اضافه کنید. دو عدد تصادفی نیز تولید کنید. برای اینکه عمل تفریق ساده شود عدد تصادفی دوم همواره باید کمتر یا برابر عدد تصادفی اولی باشد. چرا؟  
برای این منظور از دستورات زیر استفاده کنید که این ویژگی را تضمین می کند.

```
leftNumber = randomNumber.Next(1, 51);
```

```
rightNumber = randomNumber.Next(1, leftNumber+1);
```

د) توسعه و بهبود برنامه با نمایش اطلاعاتی از نتیجه سنجش، شامل تعداد سؤالات، تعداد پاسخ صحیح و تعداد پاسخ غلط :

۱- دو فیلد عددی در داخل کلاس فرم برای نگهداری تعداد سؤالات و تعداد پاسخ صحیح با مقدار اولیه صفر تعریف کنید (برای نام متغیرها از totalQuiz و correctQuiz استفاده کنید).  
۲- در متد EH، دکمه «شروع» مقدار متغیر مربوط به تعداد سؤالات را یک واحد افزایش دهید.

۳- در متد EH رویداد زمانی، اگر پس از پایان مدت سنجش، پاسخ کاربر غلط بود، یک واحد به متغیر مربوط به تعداد پاسخ غلط اضافه نمایید.

۴- در دستورات MessageBox، باید تغییری دهید که علاوه بر پیام قبلی، تعداد سؤالات و پاسخ‌های درست و نادرست (مقدار متغیرها) را نیز نشان دهد.

## خودآزمایی فصل اول

الف) درستی یا نادرستی هر عبارت را تعیین کنید.

۱- برای دکمه‌های رادیویی اگر ویژگی `RightToLeft` را `Yes` انتخاب کنیم، نوشته در سمت راست دایره قرار می‌گیرد.

۲- برای انتخاب دکمه رادیویی به عنوان پیش فرض، ویژگی `Enabled` آن را `True` قرار می‌دهیم.

۳- کنترل عددی افزایشی-کاهشی (`NumericUpDown`) برای دریافت داده‌های اعشاری هم به کار می‌رود.

۴- در متد `Random.Next(maxValue)` مقدار `maxValue` می‌تواند منفی باشد.

۵- از دکمه‌های رادیویی یک `groupBox` فقط یکی در حالت انتخاب قرار می‌گیرد. (ب) جاهای خالی را با عبارت مناسب پر کنید.

۶- از نوع داده شمارشی `MessageBoxButtons` برای تعیین نوع `MessageBox` ..... استفاده می‌شود.

۷- برای ایجاد دو دسته دکمه رادیویی مستقل از یکدیگر، باید هر دسته از دکمه‌ها را در داخل ..... قرار دهید.

۸- برای بستن فرم از متد ..... استفاده می‌شود.

۹- متد `Random.Next(maxValue)` عددی بزرگ‌تر یا مساوی ..... و کوچک‌تر از ..... به ما می‌دهد.

۱۰- برای نمایش پیام با استفاده از `MessageBox`، متد ..... فراخوانی می‌شود. (ج) به سؤالات زیر پاسخ دهید.

۱۱- تفاوت کنترل دکمه رادیویی و دکمه انتخاب را بنویسید.

۱۲- چرا در فرم‌ها برای تعیین جنسیت از دکمه رادیویی استفاده می‌شود؟

۱۳- بررسی کنید اگر در کنترل `NumericUpDown` ویژگی `Increment` را با عدد  $\frac{2}{5}$  مقداردهی کنیم، هر بار عدد داخل کنترل چه مقدار افزایش می‌یابد.

برای مقداردهی با اعداد  $\frac{2}{1}$  و  $\frac{2}{7}$  و  $\frac{3}{5}$  هم نتایج را بررسی کنید.

۱۴- از نوع داده شمارشی `DialogResult` به چه منظور استفاده می‌شود؟

۱۵- دستوری بنویسید که عدد تصادفی بین ۵ تا ۱۵ تولید کند.

۱۶- دستوری بنویسید که عدد تصادفی بین  $10^0$  تا  $10^1$  تولید کند.

## تمرینات برنامه‌نویسی فصل اول

۱- برنامه ویندوزی بسازید که یک فرم ورود به سیستم (Login) مانند شکل روبرو ایجاد کند. (دکمه‌ها فاقد متد EH هستند).



۲- در تمرین قبل، برای دکمه «ورود» یک متد EH بنویسید. یک نام کاربری و یک رمز عبور دلخواه در برنامه در نظر بگیرید. چنانچه کاربر نام کاربری و کلمه عبور را درست وارد کرد پیام «خوش آمدید» و در غیر این صورت پیام «نام کاربری یا کلمه عبور اشتباه است» با استفاده از کادر محاوره‌ای MessageBox نمایش داده شود. (دکمه «ثبت نام» فاقد متد EH است).

۳- یک فرم برای ثبت نام یک کاربر جدید مانند شکل روبرو طراحی کنید. (دکمه‌ها فاقد متد EH هستند).



۴- برای تمرین قبل شرایط زیر را در نظر بگیرید:

(الف) نام کاربری حداقل ۵ حرف و حداکثر ۳۰ حرف باشد.

(ب) کلمه عبور حداقل ۶ کاراکتر و حساس به حروف کوچک و بزرگ باشد.

(ج) کلمه عبور و تکرار باید دقیقاً یکسان باشند.

(د) در صورت کلیک بر روی دکمه «ذخیره»، شرایط بالا بررسی شده و در صورت عدم رعایت شرایط مذکور، پیام خطای مناسبی نمایش داده شود و در صورت رعایت تمام موارد، پیام «کاربر جدید با موفقیت به سیستم اضافه شد» با استفاده از کادر محاوره‌ای (MessageBox) نمایش داده شود.



ه) در صورت کلیک بر روی دکمه «پاک کردن»، محتوای کادرهای متنی پاک شد و علامت چشمک زن<sup>۱</sup> در داخل کادر متنی نام کاربری قرار گیرد.

۵- برنامه ویندوزی بسازید که یک فرم «ارتباط با ما» مانند شکل روبرو ایجاد کند. (دکمه‌ها فاقد متد EH هستند.)



۶- در تمرین قبل، برای دکمه‌های فرم، متدهای EH مناسب بنویسید به طوری که:  
الف) با کلیک بر روی دکمه «ارسال» در صورت تکمیل بودن فرم، پیام «ارسال موفقیت‌آمیز» نمایش داده شود.

ب) با کلیک بر روی دکمه «انصراف» و در صورت تأیید کاربر، فرم بسته شود.

ج) دکمه «خالی کردن فرم» نیز برای پاک کردن محتوای کادرهای متنی استفاده می‌شود. قبل از پاک کردن، تأیید کاربر توسط یک پیام مناسب گرفته شود.

---

۱- راهنمایی: برای قرار گرفتن علامت چشمک زن در کادر متنی مورد نظر، از متد (Focus) کادر متنی استفاده نمایید.

### پروژه

در این بخش از فعالیت، برای تمرین و یادگیری بهتر مطالب مطرح شده در کتاب، پروژه‌ای را ایجاد کرده و در هر فصل متناسب با محتوا، برای تکمیل پروژه یک گام به جلو برمی‌داریم. به این ترتیب در پایان کتاب با توجه به دستورات آموزش داده شده، پروژه تکمیل خواهد شد. به این منظور پروژه واژه‌نامه یا فرهنگ لغات معین را انتخاب کرده‌ایم.

در این فصل، فایل پروژه را ایجاد می‌کنیم و آن را برای تکمیل، متناسب با محتوای فصل‌های دیگر، ذخیره می‌کنیم.

### مرحله اول

امکان جستجوی معنی واژه، مهم‌ترین امکان هر واژه‌نامه است. در مرحله اول قصد داریم قسمت جستجوی این واژه‌نامه را با توجه به کنترل‌های ورود اطلاعات و آرایه پیاده‌سازی کنیم.

۱- پروژه‌ای از نوع WFA با نام Dictionary ایجاد کنید.

۲- با توجه به شکل زیر کنترل‌های لازم را روی فرم قرار داده و ویژگی‌های آنها را تنظیم نمایید.



کنترل‌های فرم اولیه پروژه واژه‌نامه

توجه کنید که کادر معنی، غیرفعال است و امکان نوشتن متن در آن وجود ندارد.

۳- برای ایجاد واژه‌نامه باید واژگان و معنی آنها را در بخشی از حافظه ذخیره کنیم، با توجه به انواع داده‌هایی که تاکنون با آن آشنا شده‌ایم می‌توان واژگان و معنی آنها را در آرایه‌ای از نوع String ذخیره کرد. برای سادگی و مطابقت با آنچه تاکنون آموخته‌اید، واژگان و معانی در دو آرایه ۳ عضوی پیاده‌سازی می‌شوند و آرایه در کلاس فرم به صورت زیر تعریف کنید:

```
string[] words = new string [] { "کنکاش", "تدبیر", "کرامت" };
string[] meanings = new [] { "مشورت، شور، کنکاش هم گفته می‌شود" (مص م. به پایان کاری) - ۱، "مشورت، شور، کنکاش هم گفته می‌شود" (مص ل. سخاوت)، "اندیشیدن. ۲- اندیشیدن. ۳- مشورت کردن. ۴- (مص.) پایان بینی. ۵- شور،
string
(مص ل. سخاوت)، "اندیشیدن. ۲- اندیشیدن. ۳- مشورت کردن. ۴- (مص.) پایان بینی. ۵- شور،
";
};
```

شما می‌توانید از کلمات دیگری به انتخاب خود، استفاده نمایید.

با توجه به تعریف آرایه، مشخص است که عناصر آرایه words واژگان واژه‌نامه و عناصر متناظر آن در آرایه meaning معنی آن واژگان می‌باشد.

۴- کد کلیک دکمه جستجو:

```
private void search_Click(object sender, EventArgs e)
{
    bool flag = false;
    for (int i = 0; i < words.Length; i++)
        if (words[i] == searchWord.Text)
        {
            description.Text = meanings[i];
            flag = true;
            break;
        }
    if (!flag)
        MessageBox.Show("پیدا نشد");
}
```

متغیر منطقی flag تعیین می‌کند واژه در واژه‌نامه وجود دارد یا خیر. در صورت وجود واژه، مقدار این متغیر True می‌شود. در پایان جستجو، در صورت false بودن این متغیر، پیام «پیدا نشد» نمایش داده می‌شود.

برنامه را اجرا کنید و یکی از ۳ واژه موجود مانند «کرامت» را جستجو و نتیجه را مشاهده نمایید. اکنون جستجو را برای واژه «مروت» تکرار کنید.

## واژگان و اصطلاحات انگلیسی فصل اول

ردیف	واژه انگلیسی	معنی به فارسی
۱	Cash	
۲	Debit card	
۳	Down	
۴	Entry	
۵	Equal	
۶	Female	
۷	Increment	
۸	Information	
۹	Leave	
۱۰	Male	
۱۱	Maximum	
۱۲	Minimum	
۱۳	Radiobutton	
۱۴	Random	
۱۵	Skill	
۱۶	Sum	
۱۷	Up	
۱۸	Value	