



پودمان سوم

طراحی واسط گرافیکی

در هر نرم افزار کاربردی تولیدشده به وسیله محیط های برنامه نویسی، تعامل کاربر با محیط نرم افزار اهمیت ویژه ای دارد. واسط گرافیکی کاربر، بخش دیداری و قابل انتخاب است که به کاربر این امکان را می دهد به وسیله نمادهای گرافیکی و نشانگرهای دیداری، با دستگاه های الکترونیکی تعامل و ارتباط برقرار کند. میزان رضایت مندی کاربر و ارتباط مؤثر با محیط نرم افزار به طراحی واسط گرافیکی در محیط برنامه نویسی، وابستگی زیادی دارد. واسط گرافیکی کاربر، با دو بخش محیط نرم افزاری و اجزای سخت افزاری مرتبط است و سبب راهنمایی و کاربری مناسب استفاده کنندگان نرم افزار می شود. در نتیجه در مقایسه با نرم افزارهای مشابه می تواند عامل تعیین کننده برای انتخاب نرم افزار به وسیله کاربران به منظور برآورده کردن انتظارات کاری در زمینه کسب و کار باشد و زمینه رقابت تجاری را برای تولید نرم افزارهای مختلف فراهم آورد. زبان برنامه نویسی C# با داشتن امکانات وسیع و متنوع برای ایجاد واسط گرافیکی، می تواند برحسب نیازهای کاربر، واسط های گرافیکی مورد انتظار را طراحی کند. در این پودمان معرفی و کار با کنترل های ایجاد واسط گرافیکی کاربر و مدیریت رویدادهای مربوط به آنها را فرا خواهید گرفت.

واحد یادگیری ۵

■ شایستگی ایجاد واسط گرافیکی کاربر

آیا تا به حال پی برده اید

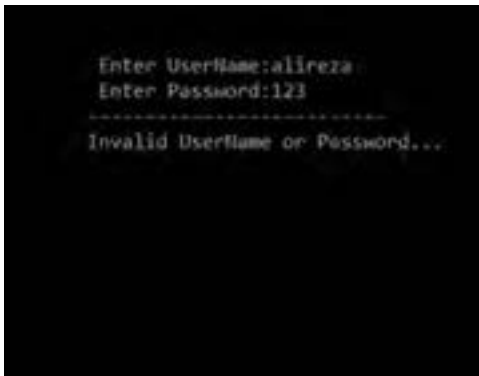
- چگونه می توان کادرهای محاوره ای انتخاب رنگ و قلم ویندوز را در برنامه فراخوانی کرد؟
 - چرا کاربران تمایل بیشتری برای کار با برنامه های گرافیکی دارند؟
 - از چه روشی برای دریافت نام کاربری و گذرواژه در برنامه استفاده کنیم؟
 - برای تغییر شکل ظاهری برنامه، چه کارهایی می توان انجام داد؟
 - بازی های ویندوزی چگونه طراحی می شوند؟
- هدف از این واحد شایستگی ایجاد واسط گرافیکی کاربر در پروژه ویندوزی است.

استاندارد عملکرد

با استفاده از IDE برنامه نویسی، یک واسط گرافیکی کاربر در پروژه ایجاد کرده و کد رویدادهای آن را بنویسد.

آشنایی با پروژه‌های ویندوزی

دو شکل زیر را با هم مقایسه کنید. چه تفاوت‌هایی بین آنها می‌بینید؟ کدام یک ظاهر زیباتری دارد؟ شما ترجیح می‌دهید از کدام شکل برای ورود نام و گذرواژه استفاده کنید؟



شکل ۱- خروجی برنامه کنسول

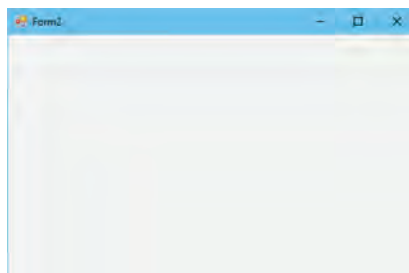
شکل ۲- خروجی برنامه ویندوزی

با ورود سیستم‌عامل‌های گرافیکی به دنیای رایانه، زبان‌های برنامه‌نویسی به وجود آمدند که در محیط گرافیکی این نوع سیستم‌عامل‌ها قادر به اجرا و فعالیت باشند. از جمله این زبان‌های برنامه‌نویسی می‌توان به، Microsoft Visual C++، Borland Delphi، Visual Basic.Net و C# اشاره کرد. خروجی این نوع برنامه‌ها در محیط ویندوز و با ظاهری زیبا و کاربرپسند قابل مشاهده است.

همه برنامه‌هایی که تا این بخش در C# نوشتید، با کاربر از طریق کنسول ارتباط برقرار می‌کرد، به همین دلیل به این برنامه‌ها، برنامه‌های کنسولی گفته می‌شود. اکنون که تجربه کافی در زمینه نوشتن برنامه‌های کنسولی را به دست آوردید، زمان آن است که با برنامه‌های ویندوزی آشنا شوید و بتوانید برنامه‌هایی بنویسید که ظاهر کاربردی‌تر و شبیه‌تر به برنامه‌های رایج امروزی دارند.

واسط گرافیکی کاربر

پروژه‌های کنسول با پروژه‌های ویندوزی تفاوت‌هایی دارند. در پروژه‌های ویندوزی، صفحه‌ای وجود دارد که روی آن ابزارهایی برای ورود اطلاعات، نمایش خروجی‌ها و کلیدهایی برای ثبت و کنترل پروژه قرار می‌گیرند (شکل ۲). ابزارهای طراحی و نمایش خروجی‌های پروژه ویندوزی روی صفحه‌ای قرار می‌گیرند که به آن فرم (Form) می‌گوییم (شکل ۳). فرم دربرگیرنده همه اجزای گرافیکی پروژه است و به عنوان یک واسط بین اجزای گرافیکی برنامه و کاربر عمل می‌کند. به همین دلیل به آن **واسط گرافیکی کاربر (GUI)** می‌گویند.



شکل ۳- فرم در پروژه ویندوزی

کنترل

به اجزای گرافیکی که در ساخت واسط گرافیکی کاربر به کار می‌روند و در محیط گرافیکی برنامه قابل مشاهده هستند، کنترل می‌گویند. حتی فرم نیز یک کنترل است. کنترل‌ها در دسته‌های مختلفی در جعبه ابزار قرار دارند. فرم، بستری برای دربرگرفتن کنترل‌ها است.

فیلم شماره ۱۱۰۵: آشنایی با IDE ویژوال استودیو در پروژه‌های ویندوزی

فیلم



فیلم را مشاهده کنید و فعالیت زیر را انجام دهید.

فعالیت کارگاهی

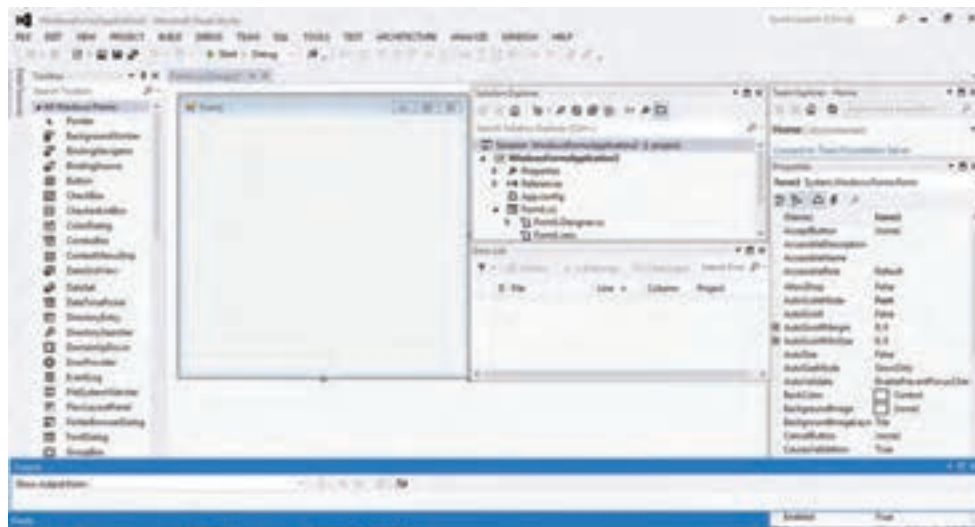


- جدول ۱ را کامل کنید.

جدول ۱- اجزای IDE پروژه‌های ویندوزی

ردیف	نام پنجره	کاربرد
۱	Toolbox	
۲		فهرستی از پرونده‌های تشکیل‌دهنده پروژه را نمایش می‌دهد.
۳		
۴	Form	
۵		
۶		

- یک پروژه جدید با نام FirstAPP در مسیر دلخواه ایجاد کرده، ابتدا جعبه ابزار را در صفحه ثابت کنید و سپس ببندید.
- پنجره ویژگی‌ها (Properties) را مخفی کنید و بعد از حالت مخفی خارج کنید.
- چیدمان IDE پروژه را شبیه شکل ۴ تنظیم کنید.



شکل ۴- چیدمان اجزای IDE پروژه ویندوز فرم

ویژگی‌های فرم

با شناخت ویژگی‌های یک کنترل و تنظیم آنها می‌توانید ظاهری زیبا برای پروژه طراحی کنید. برای مقداردهی ویژگی‌های یک کنترل در VS از پنجره ویژگی‌ها استفاده می‌کنیم.

کارگاه ۱ | تنظیم ویژگی‌های فرم


می‌خواهیم فرمی برای مصرف بهینه آب طراحی کنیم (شکل ۵).



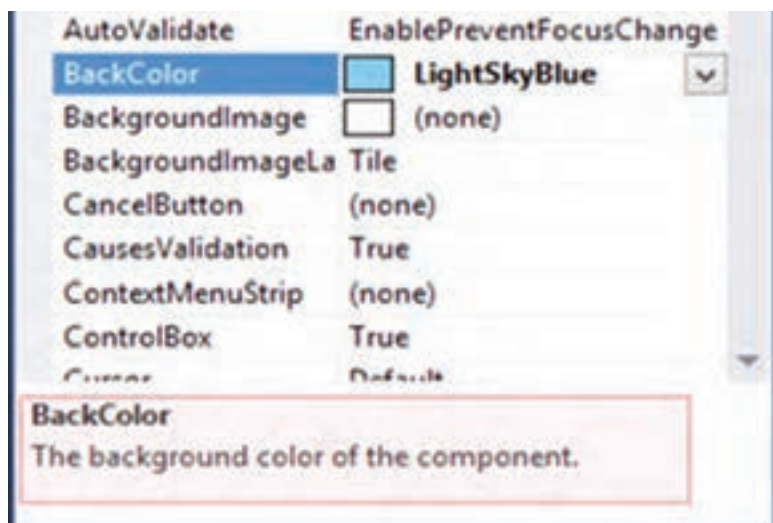
شکل ۵- فرم مصرف بهینه آب

۱ | پروژه ویندوزی با نام **WaterConsumption** ایجاد کنید.

۲ | رنگ زمینه فرم را تغییر دهید.

روی فرم راست کلیک کرده، گزینه **Properties** را انتخاب کنید. در سمت راست فرم در پنجره ویژگی‌های فرم، ویژگی **BackColor** را پیدا کرده، آن را انتخاب کنید. روی علامت  کلیک کنید تا فهرست کشویی باز شود و از زبانه **Web** رنگ **LightSkyBlue** (آبی روشن) را انتخاب کنید. رنگ‌های دیگر را آزمایش کنید.

با انتخاب هر ویژگی در پنجره Properties، شرح آن ویژگی در بخش پایین پنجره نمایش داده می‌شود (شکل ۶).



شکل ۶- پنجره Properties

۳ عنوان فرم را به «مصرف بهینه» تغییر دهید.

از پنجره ویژگی‌های فرم، مقدار ویژگی Text را به «مصرف بهینه» تغییر دهید. با تغییر مقدار ویژگی RightToLeft به Yes چه تغییری در عنوان فرم مشاهده می‌کنید؟

۴ برای فرم، تصویر زمینه قرار دهید به صورتی که تمام فرم را پر کند.

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

جدول ۲- تأثیر مقادیر ویژگی BackgroundImageLayout روی فرم

نتیجه	مقدار

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

جدول ۳- تأثیر مقادیر ویژگی **FormBorderStyle** روی فرم

مقدار	نتیجه

۶ ویژگی **Right To Left Layout** فرم را **true** کنید. در ظاهر فرم چه تغییری ایجاد می شود؟

با کمک هم گروهی خود جدول ۴ را کامل کنید.

جدول ۴- عملکرد برخی ویژگی های فرم

ویژگی	عملکرد
Font	
Size (Width , Height)	
Locked	
Text	
Icon	
ControlBox	

فعالیت گروهی



آنچه آموختم:

برداشت



۱.
۲.
۳.

کارگاه ۲ | استفاده از کنترل برچسب در واسط کاربری

همه ما بارها در برنامه‌های مختلف با راهنمایی و توضیحات برنامه، اطلاعاتی را تکمیل کرده‌ایم. این توضیحات می‌تواند متن‌هایی برای راهنمایی کاربر هنگام ورود اطلاعات یا نمایش یک پیام مهم برای جلب توجه کاربر باشد. در این کارگاه و کارگاه بعد می‌خواهیم با استفاده از کنترل‌های برچسب (Label) و کادر متن (TextBox) و دکمه (Button) یک نمونه برگ ورود کاربر بسازیم.



شکل ۷- فرم ورود کاربر

۱ پروژه ویندوزی به نام Login ایجاد کنید.

۲ ویژگی‌های فرم را تنظیم کنید (شکل ۷).

ویژگی FormBorderStyle فرم را None کنید و اندازه فرم را مانند شکل ۷ تغییر دهید. رنگ زمینه فرم را به رنگ LightSkyBlue تغییر دهید و جهت نمایش فرم را از راست به چپ کنید. قلم فرم را به Tahoma و اندازه ۱۲ تغییر دهید.

۳ کنترل برچسب را به فرم اضافه کنید.

واژه Label را در قسمت Search ToolBox جست‌وجو کنید و با دابل کلیک روی کنترل آن را به فرم اضافه کنید. از این برچسب برای نمایش عنوان برنامه استفاده می‌شود. با اضافه کردن یک کنترل به فرم، ویژگی‌های آن کنترل به عنوان کنترل فعال، در پنجره ویژگی‌های نمایش داده می‌شود.

۴ اندازه برچسب را تنظیم کنید.

با ماوس عرض (Width) برچسب را به اندازه عرض فرم تغییر دهید. عرض برچسب تغییر نمی‌کند. چرا؟ ویژگی AutoSize را false کنید و دوباره عرض برچسب را تغییر دهید.

۵ نام برچسب را تغییر دهید.

نام هر کنترل در ویژگی Name آن نگهداری می‌شود. با اضافه کردن هر کنترل به پروژه، VS نام پیش فرضی برای آن در نظر می‌گیرد که از نام کنترل و یک عدد تشکیل شده است. به عنوان مثال نام پیش فرض برچسبی که به فرم اضافه کردید Label1 است. اگر برچسب دوم را به فرم اضافه کنید، نام آن Label2 خواهد بود. در پنجره ویژگی‌ها، نام برچسب را تغییر دهید. تغییر نام هیچ تأثیری در ظاهر کنترل ندارد.

این نام به صورت پیش فرض، هم نام با متغیری است که برای دسترسی به این کنترل در برنامه، از آن استفاده می‌شود.

۶ متن داخل برچسب را تغییر دهید.

متن داخل برچسب با نام پیش فرض آن یکسان است. برای تغییر آن ویژگی Text برچسب را «نمون برگ ورود کاربر» قرار دهید. ویژگی RightToLeft برچسب را به Yes تغییر داده، نتیجه را بررسی کنید.

۷ رنگ قلم و زمینه برچسب را تغییر دهید.

ویژگی ForeColor را به White و ویژگی BackColor را به Black تغییر دهید.

۸ متن داخل برچسب را به صورت عمودی و افقی وسط چین کنید.

ویژگی TextAlign را MiddleCenter قرار دهید. این ویژگی را با مقادیر دیگر مقداردهی کرده، نتیجه را روی برچسب بررسی کنید.

۹ برچسب را به بالای فرم قفل کنید.

مقدار ویژگی Dock برچسب را Top قرار دهید تا با تغییر اندازه فرم، برچسب هم متناظر با آن تغییر کند.

۱۰ دو کنترل برچسب برای نمایش متن «نام کاربری» و «گذرواژه» به فرم اضافه کنید.

ویژگی RightToLeft فرم را به Yes تغییر دهید. ویژگی RightToLeft برچسب‌های جدید چه مقداری دارند؟

کنجکاوی



کارگاه ۳ استفاده از کادر متن و دکمه در واسط کاربری

۱ یک کنترل کادر متن (TextBox) به فرم اضافه کنید.

برای دریافت نام کاربری یک کنترل کادر متن به فرم اضافه کنید. ویژگی Name کادر متن چه مقداری دارد؟

۲ نام کادر متن را تغییر دهید.

بهتر است نامی که برای کنترل انتخاب می‌کنید، مانند متغیرها متناسب با عملکرد آن در برنامه و نشان‌دهنده نوع کنترل باشد. برای نام‌گذاری کنترل، مخفف نوع کنترل نظیر lbl برای برچسب، txt برای کادر متن، btn برای دکمه را به عملکرد کنترل اضافه کنید.

۳ اندازه کادر متن را تغییر دهید.

عرض کادر متن را مانند شکل ۷ تنظیم کنید. برای این کار علاوه بر استفاده از ماوس می‌توانید از ویژگی Size نیز استفاده کنید.

۴ ویژگی BorderStyle کادر متن را FixedSingle قرار دهید و نتیجه را بررسی کنید.

۵ ویژگی Text کادر متن را بررسی کنید.

ویژگی Text، متنی که زمان اجرا در کادر متن وارد می‌شود را نگهداری می‌کند. مقدار پیش‌فرض این ویژگی چیست؟ مقدار ویژگی Text کادر متن را تغییر دهید و با اجرای برنامه نتیجه را بررسی کنید. آیا کادر متن فقط برای دریافت اطلاعات به کار می‌رود؟

کنجکاوی



۶ کادر متن جدیدی برای دریافت گذرواژه به فرم اضافه کنید.

بعد از انتخاب کادر متن ورود نام کاربر، با فشردن کلید CTRL و کشیدن هم‌زمان ماوس، کادر متن جدیدی ایجاد کنید. آیا مقدار ویژگی‌های کادر متن جدید با کادر متن قبلی تفاوت دارد؟ کدام ویژگی کادر متن جدید با قبلی متفاوت است؟

۷ تنظیمی انجام دهید که گذرواژه وارد شده در این کادر متن مشاهده نشود.

ویژگی PasswordChar کادر متن را * قرار دهید. برنامه را اجرا کنید و در کادر متن تایپ کنید.

آیا می‌توان بیش از یک نویسه را در ویژگی PasswordChar قرار داد؟ نوع داده این ویژگی چیست؟

کنجکاوی



۸ تعداد نویسه‌های کادر متن گذرواژه را محدود کنید.

می‌خواهیم کاربر نتواند گذرواژه‌ای بیشتر از ۸ نویسه وارد کند. ویژگی `MaxLength` را برابر ۸ قرار دهید. برنامه را اجرا کنید و سعی کنید بیش از ۸ نویسه در کادر متن گذرواژه بنویسید. نتیجه چیست؟

۹ کنترل دکمه (Button) را به فرم اضافه کنید.

از کنترل دکمه برای اجرای دستورات استفاده می‌شود.

۱۰ ویژگی‌های دکمه را تنظیم کنید.

ویژگی `Text` دکمه را «ورود» و ویژگی `Cursor` آن را `Hand` قرار داده، برنامه را اجرا کنید. با قرار گرفتن ماوس روی دکمه، اشاره‌گر ماوس چه تغییری می‌کند؟

۱۱ ویژگی `Enabled` دکمه را بررسی کنید.

ویژگی `Enabled` دکمه را `false` کنید. برنامه را اجرا کرده، وضعیت دکمه را بررسی کنید. این ویژگی را برای کادر متن و برچسب نیز تغییر داده، برنامه را اجرا کنید.

۱۲ ویژگی `Visible` کنترل‌ها را بررسی کنید.

مقدار این ویژگی را برای یکی از کنترل‌ها `false` قرار دهید و برنامه را اجرا کنید.

آیا می‌توان یک ویژگی چند کنترل را هم‌زمان تغییر داد؟

کنجکاو



فعالیت گروهی



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

نام ویژگی	عملکرد	کنترل
Name	نام کنترل	همه کنترل‌ها
BackColor	رنگ زمینه	Label-TextBox-Button-Form
ForeColor		
Text		
Enabled		
Visible		
AutoSize		Label-Form-Button
TextAlign		
Size		
Location		
Cursor		
Dock		
RightToLeft		



فرم مشخصات کاربر را ایجاد کنید (شکل ۸).

- ویژگی Name کنترل های کادر متن و دکمه را به صورت زیر قرار دهید.

نام کاربر	نام خانوادگی	نام کاربری	گذرواژه	تکرار گذرواژه	نشانی	ثبت	حذف	جستجو
txtFirstName	txtLastName	txtUserName	txtPassword	txtRePassword	txtAddress	btnSave	btnDelete	btnSearch



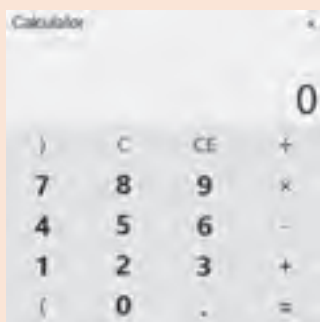
- برای کادر متن «نشانی» ویژگی Multiline را true قرار دهید.

- کادر متن «نام کاربری» حداکثر ۱۰ نویسه باشد.

شکل ۸- فرم مشخصات کاربر



ماشین حساب مشابه ویندوز ۱۰ را طراحی کنید (شکل ۹).



رنگ زمینه فرم را (۲۴۲, ۲۴۲, ۲۴۲) قرار دهید. ویژگی FlatStyle دکمه را Flat و ویژگی FlatAppearance.BorderSize را صفر قرار دهید.

شکل ۹- فرم ماشین حساب

جدول ارزشیابی شایستگی های غیر فنی، ایمنی و بهداشت و توجهات زیست محیطی

شایستگی ها	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص ها/ داوری/ نمره دهی)	نمره
شایستگی های غیر فنی	مدیریت کیفیت، پایش شاخص های کیفیت- زبان فنی	قابل قبول	انتخاب کنترل های کاربر پسند - توجه به نحوه چینش کنترل ها در فرم و تنظیم ویژگی آنها از لحاظ دسترسی راحت به آنها و زیبایی ظاهری واسط کاربری	۲
ایمنی و بهداشت	رعایت ارگونومی	غیر قابل قبول	توجه به ایمنی و بهداشت محیط کارگاه	۱
توجهات زیست محیطی	حفاظت از تجهیزات کارگاه			
نگرش	دقت در چینش خلافتی کنترل ها در فرم برای طراحی واسط گرافیکی کاربر پسند			

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



ارزشیابی مرحله ۱



مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
ایجاد واسط گرافیکی کاربری	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است. زمان: ۱۵ دقیقه	بالاتر از حد انتظار	ایجاد پروژه ویندوزی - طراحی واسط گرافیکی کاربرپسند	۳
		در حد انتظار	ایجاد پروژه ویندوزی - طراحی واسط گرافیکی کاربری	۲
		پایین‌تر از حد انتظار	ایجاد پروژه ویندوزی	۱

واکنش برنامه به رویدادها

هنگام کار با ویندوز برای انجام عملیات مختلف بارها از راست‌کلیک استفاده کرده‌اید. آیا راست‌کلیک در بخش‌های مختلف ویندوز و برنامه‌های متفاوت نتیجه یکسان دارد؟ راست‌کلیک یک رویداد (Event) و واکنش برنامه به آن، متد رویداد است. رویداد، یک کلاس یا یک شیء را قادر می‌سازد تا دیگر کلاس‌ها و اشیاء را از رخ دادن اتفاقی باخبر کند. واکنشی که برنامه در مقابل رویدادها نشان خواهد داد، باید پیش‌بینی شده و متدهای مربوط به آنها نوشته شود. به این متدها Event Handler می‌گویند. متد قطعه کدی شامل تعدادی دستور است که این مجموعه دستورات با فراخوانی متد اجرا می‌شوند.

فعالیت کارگاهی



پروژه Login را اجرا کنید. نام کاربری و گذرواژه را در کادرهای متن وارد کنید. برای خروج از برنامه، روی دکمه خروج کلیک کنید. چه اتفاقی می‌افتد؟ به نظر شما چگونه باید از اجرای پروژه خارج شویم؟

در پروژه‌های کنسولی، برنامه معمولاً به وسیله دستورات متنی اجرا می‌شود. در صورتی که مشخصه اصلی پروژه‌های ویندوزی رویدادگرا بودن آن است یعنی برنامه در زمان اجرا به رویدادهای مختلف اشیای برنامه مانند Componentها، کنترل‌ها و دیگر کلاس‌ها واکنش نشان می‌دهد.

کارگاه ۴ ایجاد واکنش به رویداد کلیک

می‌خواهیم یک ماشین حساب ساده ویندوزی طراحی و برنامه‌نویسی کنیم (شکل ۱۰).



شکل ۱۰- فرم ماشین حساب ساده

در این برنامه با کلیک روی دکمه عملگرها، نتیجه عملیات در کادر متن سوم نوشته می‌شود. با کلیک روی دکمه «خروج» برنامه بسته می‌شود و با کلیک روی دکمه «پاک کردن» محتوای کادرهای متن پاک می‌شود.

۱ پروژه ویندوزی به نام **MyCalculator** ایجاد کنید.

۲ واسط کاربری مناسب را طراحی کنید.

نام کنترل‌های کادر متن را به ترتیب `txtNumber1` و `txtNumber2` و `txtResult` قرار دهید. نام دکمه «خروج» را `btnExit` و نام دکمه «پاک کردن» را `btnClear` قرار دهید. نام دکمه عمل جمع را `btnAdd` قرار دهید (شکل ۱۰).

۳ برای دکمه «خروج» متد رویداد کلیک ایجاد کنید.

در پنجره طراحی فرم با دابل کلیک روی هر کنترل، وارد پنجره کدنویسی می‌شوید و متد رویداد پیش‌فرض کنترل، در کلاس فرم ایجاد می‌شود. رویداد پیش‌فرض کنترل دکمه، رویداد کلیک (`Click`) است.

```
namespace MyClaculator
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnExit_Click(object sender, EventArgs e)
        {
            // محل نوشتن کد
        }
    }
}
```

شکل ۱۱- متد رویداد کلیک

شکل عمومی متد رویدادها

```
private void (آرگومان‌های رویداد، فرستنده پیام) نام متد
{
    دستورات واکنش نسبت به رویداد
}
```

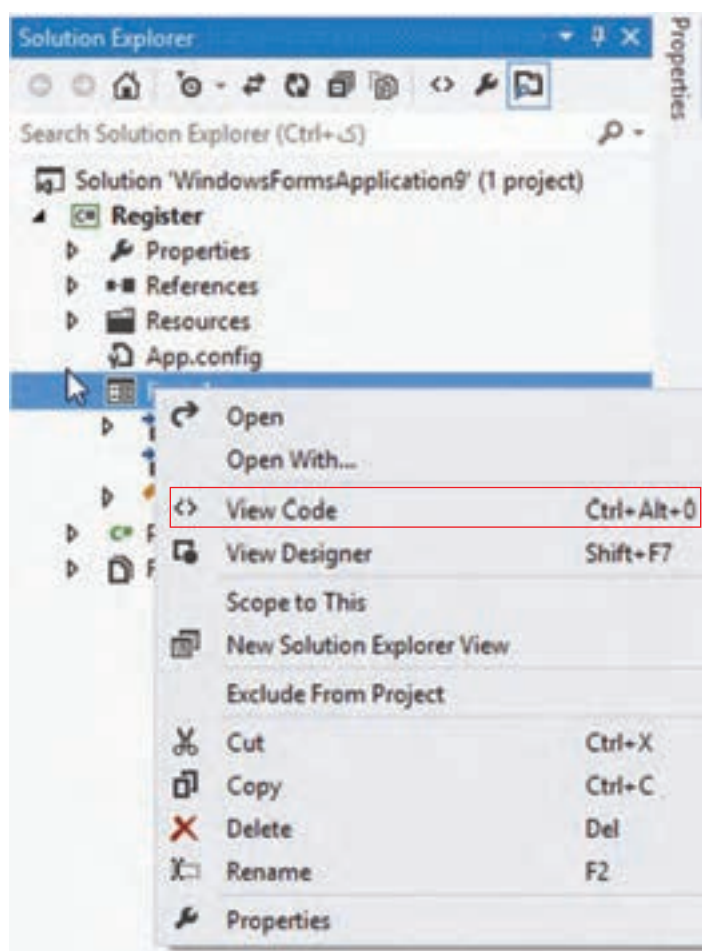
نام پیش‌فرض متد رویداد به صورت «نام رویداد - نام کنترل» است. تا زمانی که شما دستوری در بدنه متد `btnExit.Click` ننویسید، این متد کاری انجام نمی‌دهد.

۴ در متد btnExit_Click دستور زیر را بنویسید.

```
private void btnExit_Click (object sender, EventArgs e)
{
    this.Close();
}
```

به جای متد this.Close() از Close() نیز می‌توانید استفاده کنید. منظور از کلیدواژه this فرمی است که متد در کلاس آن نوشته شده است. برنامه را با کلید F5 اجرا کنید. روی دکمه خروج کلیک کنید. چه اتفاقی می‌افتد؟
۵ وارد صفحه کدنویسی شوید.

برای رفتن به صفحه کدنویسی چند روش وجود دارد. می‌توانید روی پرونده فرم (Form1.cs) در پنجره Solution Explorer راست کلیک کرده، گزینه View Code را انتخاب کنید (شکل ۱۲).



شکل ۱۲- منوی زمینه‌ای پرونده فرم

۶ برای پاک کردن محتوای کادرهای متن، متد رویداد Click دکمه «پاک کردن» را بنویسید. محتوای کادر متن در ویژگی Text آن نگهداری می‌شود. برای تغییر این محتوا باید ویژگی Text را مقداردهی کنید.

شکل کلی تغییر مقدار ویژگی یک کنترل هنگام کدنویسی

مقدار جدید ویژگی = نام ویژگی. نام کنترل

برای مثال برای نوشتن Ali در کادر متن txtFirstName کد زیر را بنویسید:

```
txtFirstName.Text="Ali";
private void btnClear_Click(object sender, EventArgs e)
{
    txtNumber1.Text = "";
    txtNumber2.Text = "";
    txtResult.Text = "";
}
```

کد txtNumber1.Text = "" چه کاری انجام می‌دهد؟

نوع داده ویژگی‌های کنترل با هم متفاوت است. برای مثال ویژگی Text از نوع رشته است. برای اطلاع از نوع داده ویژگی کافیست در هنگام کدنویسی اشاره‌گر ماوس را روی ویژگی نگه دارید.

```
txtNumber1.Text = "";
```

string TextBox.Text
Gets or sets the current text in the System.Windows.Forms.TextBox.

۷ متد رویداد کلیک دکمه «پاک کردن» را به روش دیگری بنویسید.

از متد Clear کادر متن هم می‌توان برای پاک کردن متن استفاده کرد.

```
txtNumber1.Clear();
```

۸ متد رویداد کلیک دکمه عمل جمع را بنویسید.

```
private void btnAdd_Click(object sender, EventArgs e)
{
    double num1 = double.Parse(txtNumber1.Text);
    double num2 = double.Parse(txtNumber2.Text);
    double res = num1 + num2;
    txtResult.Text = res.ToString();
}
```

برای تبدیل مقدار یک متغیر عددی به نوع رشته‌ای از متد ToString استفاده می‌شود. چرا برای انتساب ویژگی Text کادرهای متن به متغیر num1 و num2 از متد Parse استفاده شده است؟

برنامه را اجرا کنید. با ورود دو عدد در کادر متن‌ها و کلیک دکمه جمع، خروجی را مشاهده کنید.

۹ متد رویداد کلیک دکمه‌های عمل ضرب، تفریق و تقسیم را بنویسید.

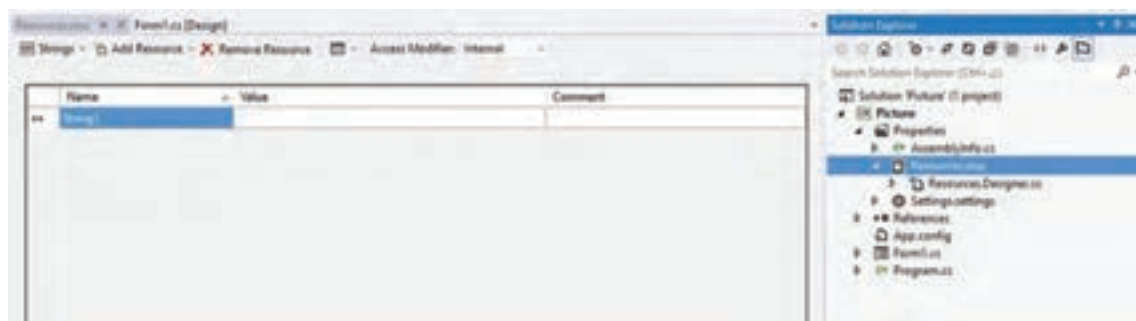
کارگاه ۵ | اضافه کردن تصویر به پوشه منابع Resource

می‌خواهیم یک آلبوم تصویر ایجاد کنیم. برای استفاده راحت‌تر از تصاویر موردنظر در برنامه، آنها را به عنوان منابع پروژه در پوشه Resource قرار می‌دهیم.

۱ یک پروژه ویندوزی به نام **PictureViewer** ایجاد کنید.

۲ کادر مدیریت پرونده‌های منبع را باز کنید.

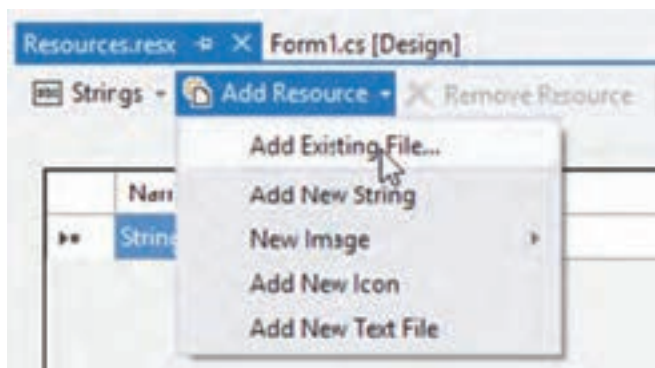
در پنجره Solution Explorer روی مثلث کنار پوشه Properties کلیک کنید تا ساختار درختی آن باز شود، سپس روی پرونده Resources.resx دابل کلیک کنید تا زبانه آن به پروژه اضافه شود (شکل ۱۳).



شکل ۱۳- پنجره مدیریت پرونده‌های منبع

۳ پرونده تصویر موجود را به پروژه اضافه کنید.

کادر Resources.resx که در صفحه پروژه ظاهر شده، شامل گزینه Add Resource است که به شما امکان اضافه کردن پرونده‌هایی به عنوان پرونده منبع در پروژه را می‌دهد. روی مثلث کنار این گزینه کلیک کنید تا فهرست کشویی آن باز شود، سپس گزینه Add Existing File را انتخاب کنید (شکل ۱۴).



شکل ۱۴- اضافه کردن پرونده‌های موجود به عنوان منبع

در کادر باز شده، یکی از تصاویر موجود در سیستم خود را انتخاب کنید. چه اتفاقی می‌افتد؟

۴ پنج پرونده تصویر به منبع پروژه اضافه کنید.

با اضافه کردن هر پرونده تصویر به پنجره Solution Explorer دقت کنید چه تغییری در این پنجره مشاهده می‌کنید؟

۵ سه تصویر دیگر با روش کشیدن و رها کردن (Drag & Drop) به پروژه منبع پروژه اضافه کنید.

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

فعالیت کارگاهی



یادداشت



در مسیر ذخیره‌سازی پروژه، پوشه‌ای به نام Resources وجود دارد که یک نسخه از پرونده تصاویر انتخاب شده، در آن قرار دارد. به همین دلیل به این پوشه، پوشه منابع (Resource File) می‌گویند.

کارگاه ۶ استفاده از کنترل کادر تصویر در برنامه

می‌خواهیم در برنامه آلبوم تصویر، عکس منظره‌ای از شهر را در وسط فرم قرار دهیم.

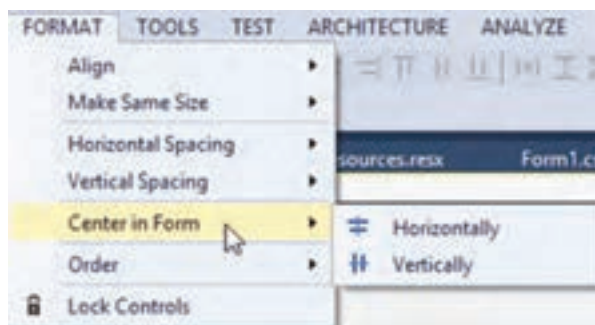
۱ پروژه PictureBox را باز کنید.

۲ یک کنترل کادر تصویر (PictureBox) روی فرم قرار دهید.

یک کنترل کادر تصویر با نام picSample با اندازه ۱۵۰ در ۱۰۰ به فرم اضافه کنید.

۳ کادر تصویر را وسط فرم قرار دهید.

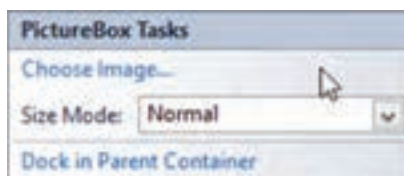
برای تنظیم فاصله کنترل‌ها از یکدیگر و فرم از منوی Format استفاده کنید (شکل ۱۵).



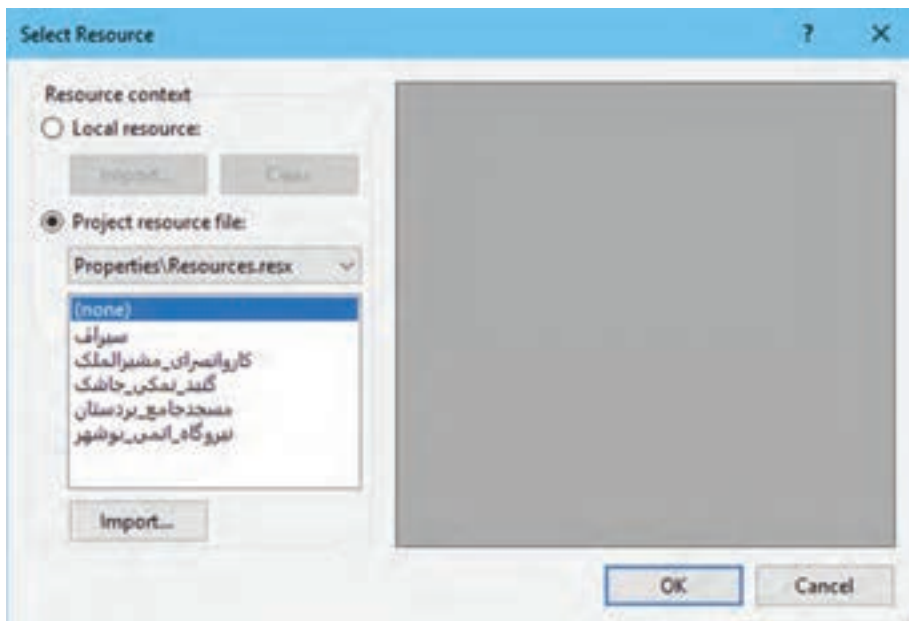
شکل ۱۵- وسط چین کردن کنترل

۴ تصویر دلخواه را در کادر تصویر وارد کنید (Import).

روی مثلث کنار کادر تصویر کلیک کنید و گزینه Choose Image را انتخاب کنید (شکل ۱۶). کادر محاوره‌ای Select Resource باز می‌شود (شکل ۱۷).



شکل ۱۶- منوی مقداردهی ویژگی‌های مهم کادر تصویر



شکل ۱۷- کادر محاوره‌ای Select Resource

فعالیت گروهی



با استفاده از گزینه‌های Local resource و Project resource file تصویر دلخواه را در کادر تصویر قرار دهید و تفاوت این دو گزینه را بنویسید.

۵ ویژگی کادر تصویر را تنظیم کنید تا تصویر به اندازه کادر تصویر شده، کامل دیده شود.

ممکن است تصویر انتخابی شما به طور کامل در کادر تصویر دیده نشود. این مشکل زمانی دیده می‌شود که اندازه کنترل کادر تصویر کوچک‌تر از اندازه تصویر مورد نظر باشد. ویژگی SizeMode را انتخاب کرده، مقدار آن را روی StretchImage قرار دهید. عملکرد این ویژگی چیست؟

۶ تصویر کادر تصویر را حذف کنید.

مقدار ویژگی Image کادر تصویر چه چیزی را نشان می‌دهد؟ مقدار این ویژگی را پاک کنید. چه اتفاقی می‌افتد؟
۷ دو دکمه با متن تصویر ۱ و تصویر ۲ به فرم اضافه کنید.

۸ کدی بنویسید که با کلیک روی هر دکمه تصویر متناظر آن در کادر تصویر نمایش داده شود.

در متد رویداد کلیک دکمه، ویژگی Image را با تصویر مورد نظر مقداردهی کنید. این ویژگی از نوع تصویر است. از تصاویری استفاده کنید که در پوشه منابع قرار دادیم. در کدنویسی برای دسترسی به منابع پروژه، باید از کلاس Resources که در فضای نام Properties است به صورت زیر استفاده کنیم.

نام منبع: Properties.Resources

برای نمایش تصویر مورد نظر در کادر تصویر، در دستور زیر به جای نام منبع، نام پرونده تصویر موجود در پوشه منابع را بنویسید.

نام منبع: Properties.Resources.picSample.Image

۹ برای نمایش سایر تصاویر، دکمه اضافه کنید و کد رویداد کلیک هر یک را بنویسید.

به فرم پروژه Login یک کادر متن و به فرم مشخصات کاربری یک کنترل کادر تصویر با نام picUser و یک دکمه برای بارگذاری تصویر کاربر با نام btnLoadPicture اضافه کنید.



فعالیت کارگاهی



کارگاه ۷ تغییر اندازه و محل کنترل روی فرم

می‌خواهیم برنامه آلبوم تصویر را توسعه دهیم تا قابلیت نمایش تصویر، بزرگ‌تر و کوچک‌تر شدن تصویر را داشته باشد.



شکل ۱۸- فرم نمایشگر تصویر

۱ پروژه PictureViewer را باز کنید.

۲ طراحی فرم را انجام دهید.

اندازه عرض فرم ۴۰۰ و ارتفاع فرم را ۳۰۰ قرار دهید. دکمه‌ای برای بزرگ‌نمایی به نام btnZoomIn و دکمه‌ای برای کوچک‌نمایی به نام btnZoomOut و دو دکمه btnNext و btnPrev برای نمایش تصویر قبلی و بعدی روی کادر تصویر به فرم اضافه کنید (شکل ۱۸).

۳ متد رویداد کلیک دکمه btnZoomIn را بنویسید.

با دابل کلیک روی دکمه btnZoomIn متد رویداد کلیک را ایجاد کرده، کدهای زیر را بنویسید.

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

```
{
    picSample.Width += 4;
    picSample.Height += 3;
}
```

برنامه را اجرا کنید و نتیجه کلیک روی دکمه + را بررسی کنید. آیا تصویر پس از تغییر اندازه، در وسط فرم باقی می‌ماند؟

۴ برای قرار گرفتن تصویر وسط فرم، کد متد `btnZoomIn_Click` را توسعه دهید.



شکل ۱۹- ویژگی `Left` و `Top` کنترل

دو ویژگی `Left` و `Top` کنترل‌ها، موقعیت کنترل را روی فرم نشان می‌دهند (شکل ۱۹). اگر `Left=0` و `Top=0` باشد، کنترل در کدام نقطه از فرم قرار می‌گیرد؟ دستوراتی برای افزایش مقدار ویژگی `Left` و `Top` کادر تصویر به متد `btnZoomIn_Click` اضافه کنید و با اجرای برنامه، نتیجه آنها را بررسی کنید. سپس این دستورات را از برنامه حذف کنید. آیا در پنجره ویژگی‌ها به `Left` و `Top` دسترسی دارید؟

برای اینکه تصویر وسط فرم قرار گیرد، دستورات زیر را به متد `btnZoomIn_Click` اضافه کنید.

```
picSample.Left = (this.Width - picSample.Width) / 2;
```

```
picSample.Top = (this.Height - picSample.Height) / 2;
```

۵ متد رویداد کلیک دکمه کوچک‌نمایی را بنویسید.

۶ متد رویداد `btnNext_Click` را بنویسید.

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

```
{
```

```
    picSample.ImageLocation = "c:\\1.jpg";
```

```
}
```

تصویر دلخواه `1.jpg` را در درایو `C` قرار دهید. با قرار دادن نشانی تصویر موردنظر در ویژگی `ImageLocation` می‌توانید تصویر را داخل کادر تصویر قرار دهید. دقت کنید که نشانی و نام تصویر را به شکل صحیح و کامل بنویسید و به جای علامت `\` در مسیر از `\\` استفاده کنید. برنامه را اجرا کنید. با کلیک روی دکمه `btnNext` چه مشاهده می‌کنید؟

۷ متد رویداد کلیک دکمه `btnNext` را برای نمایش ۵ تصویر متفاوت تغییر دهید.

برای نمایش تصویرهای مختلف روی کادر تصویر، نسخه مشابه پنج پرونده تصویری دلخواه با نام `1.JPG` و `2.JPG` و `3.JPG` و `4.JPG` و `5.JPG` را در پوشه `Debug` برنامه قرار دهید. پوشه `Debug` برنامه که حاوی پرونده اجرایی برنامه است در مسیر «`bin\پوشه پروژه`» قرار دارد.

برای نگهداری شماره تصویری که در کادر تصویر دیده می‌شود، یک متغیر در بخش کلاس فرم تعریف کنید.

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

```
{
```

```
    int i = 1; // شماره پرونده تصویری که در کادر تصویر بارگذاری می‌شود
```

متد btnNext_Click را به صورت زیر تغییر دهید.

```
private void btnNext_Click(object sender, EventArgs e)
{
    string path = i.ToString() + ".JPG";
    picSample.ImageLocation = path;
    i++;
    if (i == 6)
        i = 1;
}
```

برنامه را اجرا کنید و چندین بار روی دکمه btnNext کلیک کنید. عملکرد دستور if در متد btnNext_Click چیست؟

چرا یک نسخه از پرونده‌های تصویری را در پوشه Debug قرار دادیم؟

کنجکاوی



۸ برای نمایش تصویر قبلی متد رویداد کلیک btnPrev را بنویسید.

۹ کد را طوری تغییر دهید تا با نمایش تصویر آخر دکمه btnNext و با نمایش تصویر اول دکمه btnPrev غیرفعال شوند.

یک دکمه روی فرم قرار دهید و در متد رویداد کلیک آن هر بار یکی از دستورات زیر را نوشته، برنامه را اجرا کنید و عملکرد هر دستور را بنویسید.

```
picSample.Show();
picSample.Hide();
```

فعالیت کارگاهی



فعالیت منزل



شکل ۲۰- طراحی بازی ساده

شکل ۲۰ فرم طراحی یک بازی ساده را نشان می‌دهد. در این بازی آدمک با کلیک دکمه‌های جهت‌دار به اندازه یک کاشی در جهت تعیین‌شده، حرکت می‌کند و نمی‌تواند از کاشی‌ها عبور کند. با خرید کالای ایرانی یعنی برخورد با کاشی کالای ایرانی امتیاز کسب می‌کند و در صورت خرید کالای خارجی و یا برخورد با خرچنگ‌ها امتیاز از دست می‌دهد. این فرم را طراحی کرده، متد رویداد کلیک دکمه‌ها را برای حرکت آدمک بنویسید. متد حرکت‌های دیگر، تشخیص برخورد و ثبت امتیاز را می‌توانید با مهارت‌هایی که در بخش‌های بعدی به‌دست می‌آورید، بنویسید.



آنچه آموختیم:

۱.
۲.
۳.

ارزشیابی مرحله ۲



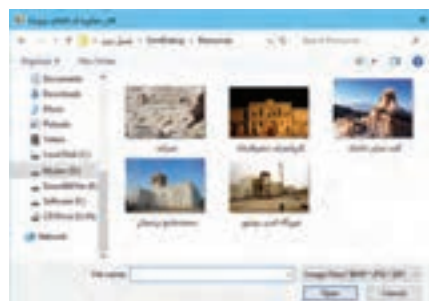
مراحل کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره‌دهی)	نمره
واکنش به رویدادها	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است پرونده تصویر زمان: ۲۰ دقیقه	بالاتر از حد انتظار	تعیین رویداد مورد نیاز - نوشتن متد رویداد - رفع خطای برنامه	۳
		در حد انتظار	تعیین رویداد مورد نیاز - نوشتن متد رویداد	۲
		پایین‌تر از حد انتظار	تعیین رویداد مورد نیاز	۱

کادرهای محاوره‌ای



شکل ۲۱- کادر محاوره‌ای پیام

در برنامه‌های مختلف کادرهایی شبیه شکل ۲۱ بسیار دیده‌اید. به این کادرها، کادر محاوره‌ای (DialogBox) می‌گوییم.

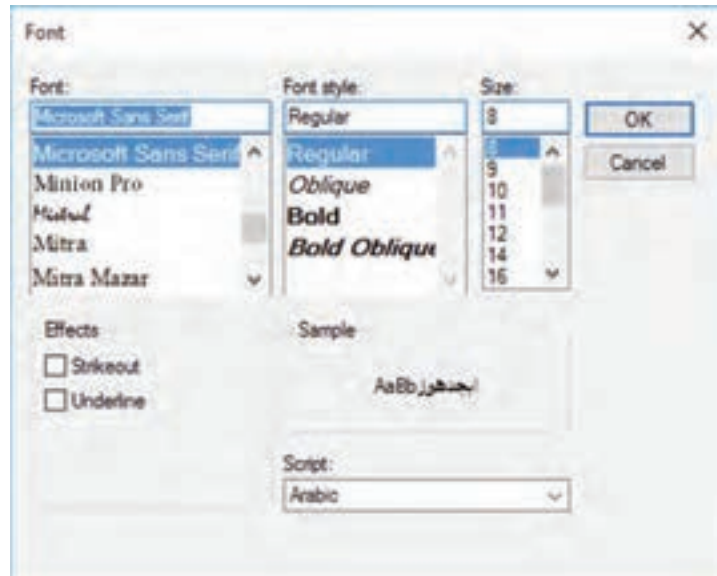


شکل ۲۲- کادر محاوره‌ای انتخاب پرونده

از کادر محاوره‌ای برای نمایش پیام و دریافت اطلاعات از کاربر استفاده می‌شود. کادر محاوره‌ای می‌تواند براساس موقعیت برنامه دارای ظاهری متفاوت باشد. برخی از کادرهای محاوره‌ای دارای عملکرد از پیش تعیین شده هستند. مانند کادر محاوره‌ای OpenFileDialog برای انتخاب پرونده (شکل ۲۲)، و ColorDialog برای انتخاب رنگ (شکل ۲۳) و FontDialog برای انتخاب قلم (شکل ۲۴) که در بیشتر برنامه‌ها استفاده می‌شوند.



شکل ۲۳- کادر محاوره‌ای انتخاب رنگ



شکل ۲۴- کادر محاوره‌ای انتخاب قلم

کارگاه ۸ کادر محاوره‌ای انتخاب پرونده

می‌خواهیم در فرم مشخصات کاربر علاوه بر دریافت مشخصات کاربر، امکان قرار دادن تصویر کاربر را هم اضافه کنیم.

۱ پروژه ثبت اطلاعات کاربر را باز کنید.

۲ یک کادر محاوره‌ای **OpenFileDialog** به فرم اضافه کنید.

ابزارهای موجود در جعبه ابزار نظیر **OpenFileDialog** که هنگام استفاده روی فرم دیده نمی‌شوند، **Component** هستند.

۳ روی دکمه «بارگذاری تصویر» دابل کلیک کنید و کدهای زیر را بنویسید.

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

```
{
    openFileDialog1.ShowDialog();
    picUser.ImageLocation = openFileDialog1.FileName;
}
```

برای نمایش کادر محاوره‌ای از متد (**ShowDialog**) استفاده می‌کنیم. نشانی پرونده انتخاب شده در ویژگی **FileName** کادر محاوره‌ای قرار دارد. برنامه را اجرا کنید و یک پرونده تصویری انتخاب کنید. آیا می‌توان پرونده‌های غیرتصویری را انتخاب کرد؟

۴ نمایش پرونده‌ها را در کادر محاوره‌ای انتخاب پرونده محدود کنید.

برای محدود کردن نمایش پرونده‌ها در کادر محاوره‌ای کد زیر را قبل از متد **ShowDialog** اضافه کنید.

```
openFileDialog1.Filter = "Image Files(*.BMP;*.JPG;*.GIF)|*.BMP;*.JPG;*.GIF";
```

کادر محاوره‌ای باز کردن پرونده دارای ویژگی **Filter** است که با آن می‌توانیم نوع پرونده‌هایی را تعیین کنیم که در کادر محاوره‌ای نشان داده می‌شود.

شکل کلی مقداردهی ویژگی Filter

....|فیلتر نوع دوم |توضیحات نوع اول|توضیحات نوع اول

اگر در کادر محاوره‌ای پرونده‌ای انتخاب نشود، چه چیزی در کادر تصویر نمایش داده خواهد شد؟
۵ بر اساس پاسخ کاربر به کادر محاوره‌ای، برنامه را مدیریت کنید.

برای مدیریت برنامه با توجه به عملکرد کاربر کد مرحله ۴ را به صورت زیر ویرایش کنید.

```
openFileDialog1.Filter = "Image Files(*.BMP;*.JPG;*.GIF)|*.BMP;*.JPG;*.GIF";
```

```
if(openFileDialog1.ShowDialog() == DialogResult.OK)
```

```
picUser.ImageLocation = openFileDialog1.FileName;
```

برنامه را اجرا کنید. در صورت عدم انتخاب پرونده در کادر محاوره‌ای چه اتفاقی می‌افتد؟

کارگاه ۹ کادرهای محاوره‌ای رنگ و قلم

می‌خواهیم یک برنامه با عنوان «شهر من خانه من» ایجاد کنیم که کاربر بتواند برای هر تصویر انتخابی خود، نظر خود را با رنگ و قلم دلخواه در کادر متن بنویسد.



شکل ۲۵- استفاده از کادرهای محاوره‌ای

۱ یک پروژه ویندوزی به نام PictureCity ایجاد کنید.

۲ فرم مناسب را طراحی کنید (شکل ۲۵).

یک کنترل کادر متن و کادر تصویر، سه دکمه برای انتخاب تصویر، رنگ و قلم و یک دکمه برای خروج از برنامه و کادر محاوره‌ای OpenFileDialog و ColorDialog و FontDialog روی فرم قرار دهید.

۳ ویژگی کادر متن را برای نوشتن متن به زبان فارسی تغییر دهید.

۴ کد متد رویداد کلیک دکمه «انتخاب عکس» را برای تغییر عکس به انتخاب کاربر بنویسید.

۵ کد متد رویداد کلیک دکمه «انتخاب رنگ» را بنویسید.

برای این که رنگ قلم کادر متن به رنگ انتخابی کاربر تغییر کند، دستور زیر را بنویسید.


```
ColorDialog1.ShowDialog();
```

```
textBox1.ForeColor = colorDialog1.Color;
```

رنگی که کاربر به وسیله کادر محاوره‌ای ColorDialog انتخاب می‌کند، در ویژگی Color این کنترل ذخیره می‌شود. **۶** کد متد رویداد کلیک دکمه «انتخاب قلم» را برای تعیین قلم به انتخاب کاربر بنویسید.

```
FontDialog1.ShowDialog();
```

```
textBox1.Font = fontDialog1.Font;
```

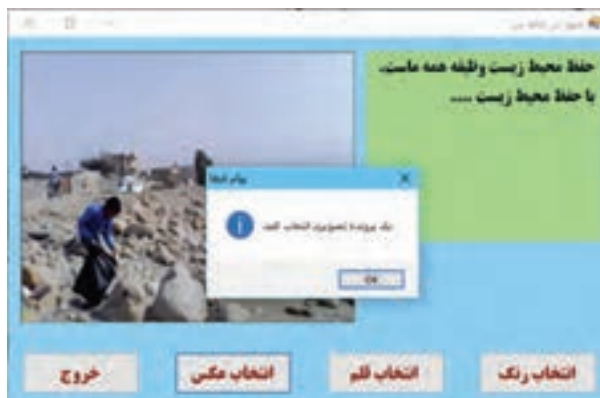
ویژگی Font در کادر محاوره‌ای FontDialog، تنظیمات قلم انتخاب شده به وسیله کاربر شامل نوع، اندازه و حالت قلم را نشان می‌دهد. با اجرای این دستور همه این تنظیمات برای قلم کادر متن مقداری می‌شود. برنامه را اجرا کنید و نتیجه کلیک روی دکمه‌ها را ببینید.

۷ عنوان کادرهای محاوره‌ای را تغییر دهید.

برای تعیین عنوان کادر محاوره‌ای از ویژگی Title استفاده کنید. ویژگی Title کادر محاوره‌ای OpenFileDialog را «انتخاب عکس» قرار دهید. برنامه را اجرا کنید و روی دکمه «انتخاب عکس» کلیک کنید. چه تفاوتی با اجرای قبل مشاهده می‌کنید؟

کادر پیام

گاهی در یک برنامه نیاز دارید که پیامی را به کاربر اطلاع دهید یا به کاربر هشدار دهید که یک پیشامد غیرمنتظره رخ داده است. در چنین مواقعی از کادرهای پیام استفاده می‌شود. برای مثال فرض کنید هنگام اجرای کارگاه ۸ کاربر پرونده‌ای را انتخاب کرده که قالب پرونده‌های تصویر را ندارد، در این صورت باید به وسیله یک کادر پیام به او هشدار داده شود (شکل ۲۶).



شکل ۲۶- کادر محاوره‌ای پیام

برای نمایش کادر پیام از کلاس **MessageBox** استفاده می‌کنیم. این کلاس دارای متد **Show** است که کادر پیام را با پیام دلخواه شما نمایش داده، منتظر پاسخ کاربر می‌ماند. متن پیام، عنوان پنجره، تعداد و نوع دکمه‌های کادر پیام به وسیله پارامترهایی مشخص می‌شود که به متد **Show** فرستاده می‌شوند.

شکل کلی فراخوانی متد Show

(دکمه پیش فرض, Icon, دکمه‌ها, عنوان, پیام); **MessageBox.Show**

جدول ۵- پارامترهای متد Show

پارامتر	نوع	شرح
پیام	string	متنی است که در کادر نمایش داده می‌شود. این پارامتر اجباری است.
عنوان	string	متنی است که در نوار عنوان کادر نمایش داده می‌شود.
دکمه‌ها	نوع داده شمارشی MessageBoxButtons	دکمه‌هایی را مشخص می‌کند که باید در کادر پیام نمایش داده شوند.
نماد	نوع داده شمارشی MessageBoxIcon	نماد کادر پیام
دکمه پیش‌فرض	نوع داده شمارشی MessageBoxDefaultButton	دکمه پیش‌فرض کادر پیام را تعیین می‌کند.

نوع داده شمارشی **MessageBoxButtons** تعیین‌کننده نوع دکمه است. هنگامی که کاربر یکی از دکمه‌های کادر پیام را انتخاب می‌کند، دکمه انتخابی در متغیری از نوع داده شمارشی **DialogResult** قرار می‌گیرد.

فعالیت گروهی



با استفاده از راهنمای مایکروسافت و به کمک هم گروهی خود اعضای نوع داده‌های شمارشی، **MessageBoxIcon**، **MessageBox DefaultButton**، **MessageBoxButtons** و **DialogResult** را تعیین کنید.

کارگاه ۱۰ توسعه برنامه با استفاده از کادر پیام

می‌خواهیم در پروژه فرم اطلاعات کاربر، از ورود نام بیشتر از ۱۰ نویسه با نمایش پیام مناسب جلوگیری کنیم.

۱ پروژه ثبت اطلاعات کاربر را باز کنید.

۲ متد رویداد کلیک دکمه «ثبت» را به صورت زیر بنویسید.

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

```
{
    int firstNameLength = txtFirstName.TextLength;
    if(firstNameLength>10)
        MessageBox.Show("تعداد نویسه‌های نام بیشتر از حد مجاز است",
        MessageBoxButtons.RetryCancel);
}
```

ویژگی **TextLength** تعداد نویسه‌های کادر متن را نگهداری می‌کند. برنامه را اجرا کنید. نامی بیش از ۱۰ نویسه وارد کنید و روی دکمه ثبت کلیک کنید. چه اتفاقی می‌افتد؟



شکل ۲۷- پیام خطا در فرم اطلاعات کاربر

۳ دستورات را کامل کنید تا همین پیام خطا برای نام خانوادگی نیز نمایش داده شود (شکل ۲۷). عبارت "پیام خطا" را حذف کنید. چه اتفاقی می افتد؟

۴ کد را ویرایش کنید تا در صورت انتخاب دکمه **Retry** کادر متن نام و نام خانوادگی خالی شود. متد **Show**، مقداری از نوع داده شمارشی **DialogResult** برمی گرداند. این مقدار مشخص می کند، کدام یک از دکمه های کادر پیام به وسیله کاربر انتخاب شده است. با توجه به این نکته، می خواهیم برنامه را به شکلی کامل کنیم که اگر کاربر دکمه **Retry** را انتخاب کرد، متن وارد شده در نام و نام خانوادگی حذف شده، آماده دریافت اطلاعات جدید شود. دستورات دکمه ثبت را به شکل زیر تغییر دهید.

```
if(MessageBox.Show("پیام خطا", "تعداد نویسه های نام بیشتر از حد مجاز است",  
    MessageBoxButtons.RetryCancel) == DialogResult.Retry)  
    txtFirstName.Clear();
```

۵ دستورات مرحله ۴ را به صورت زیر تغییر دهید.

```
if(MessageBox.Show("پیام خطا", "تعداد نویسه های نام بیشتر از حد مجاز است",  
    MessageBoxButtons.RetryCancel, MessageBoxIcon.Warning,  
    MessageBoxDefaultButton.Button1) == DialogResult.Retry)  
    txtFirstName.Clear();
```

برنامه را اجرا کنید. نتیجه اجرای برنامه چه تفاوتی با اجرای قبل دارد؟

۶ دستورات رویداد کلیک دکمه «ثبت» را کامل کنید تا در صورت انتخاب دکمه **Retry** در پیام خطای نام خانوادگی، کادر متن «نام خانوادگی» خالی شود.

با کمک هنرآموز خود تغییراتی در دستورات کارگاه ۱۰ ایجاد کنید که دو دکمه **Ok**, **Cancel** در کادر پیام نمایش داده شود. دکمه پیش فرض را دکمه **Cancel** تعیین کنید و تصویر نماد کادر محاوره ای را تغییر دهید.

فعالیت کارگاهی



برنامه ای بنویسید که در «فرم ورود کاربر» اگر نام کاربری و گذرواژه معتبر بود پیام خوش آمدگویی و در صورت معتبر نبودن پیام خطا را نمایش دهد.

فعالیت منزل





آنچه آموختم:

۱.
۲.
۳.

ارزشیابی مرحله ۳



مراحل کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره‌دهی)	نمره
استفاده از کادرهای محاوره‌ای	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است. زمان: ۲۰ دقیقه	بالاتر از حد انتظار	افزودن کادر محاوره‌ای به واسط گرافیکی کاربری و تنظیم ویژگی‌های آن - نوشتن برنامه استفاده از کادر محاوره‌ای - نمایش کادر پیام با مشخصات تعیین شده - مدیریت برنامه بر اساس پاسخ کاربر-رفع خطای برنامه	۳
		در حد انتظار	افزودن کادر محاوره‌ای به واسط گرافیکی کاربری و تنظیم ویژگی‌های آن - نوشتن برنامه استفاده از کادر محاوره‌ای - نمایش کادر پیام با مشخصات تعیین شده	۲
		پایین‌تر از حد انتظار	افزودن کادر محاوره‌ای به واسط گرافیکی کاربری و تنظیم ویژگی‌های آن	۱
معیار شایستگی انجام کار: کسب حداقل نمره ۲ از مرحله ایجاد واسط گرافیکی کاربری کسب حداقل نمره ۲ از بخش شایستگی‌های غیرفنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش کسب حداقل میانگین ۲ از مراحل کار				

جدول ارزشیابی پایانی

شرح کار:

۱ ایجاد واسط گرافیکی کاربری

۲ استفاده از کادرهای محاوره‌ای

۲ واکنش به رویدادها

استاندارد عملکرد:

با استفاده از IDE برنامه نویسی، یک واسط گرافیکی کاربر در پروژه ایجاد کرده و کد رویدادهای آن را بنویسد.

شاخص‌ها:

شماره مرحله کار	شاخص‌های مرحله کار
۱	ایجاد پروژه ویندوزی - طراحی واسط گرافیکی کاربرپسند
۲	نوشتن متد رویدادهای مورد نیاز برنامه - رفع خطای برنامه
۳	به کارگیری کادرهای محاوره‌ای مورد نیاز در برنامه - استفاده از کادر پیام بر اساس نیاز - مدیریت برنامه بر اساس پاسخ کاربر - 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: ۶۰ دقیقه (ایجاد واسط گرافیکی کاربری ۱۵ دقیقه - واکنش به رویدادها ۲۰ دقیقه - استفاده از کادرهای محاوره‌ای ۲۰ دقیقه)

معیار شایستگی:

ردیف	مرحله کار	حداقل نمره قبولی از ۳	نمره هنرجو
۱	ایجاد واسط گرافیکی کاربری	۲	
۲	واکنش به رویدادها	۱	
۳	استفاده از کادرهای محاوره‌ای	۱	
شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست‌محیطی و نگرش: مدیریت کیفیت، پایش شاخص‌های کیفیت - زبان فنی رعایت ارگونومی حفاظت از تجهیزات کارگاه دقت در چیش خلاقانه کنترل‌ها در فرم برای طراحی واسط گرافیکی کاربرپسند			
میانگین نمرات			*

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

واحد یادگیری ۶

■ شایستگی کار با کنترل‌های پیشرفته

آیا تا به حال پی برده اید

- در بازی‌های رایانه‌ای کنترل زمانی چگونه طراحی می‌شود؟
 - چگونه می‌توان ساعت رایانه را در بالای پروژه‌ها اضافه کرد؟
 - آزمون‌های چند گزینه‌ای یا صحیح - غلط چگونه در سی شارپ کدنویسی می‌شوند؟
 - چگونه می‌توان تعداد زیادی کنترل را به شکلی زیبا و با دسترسی راحت در یک فرم جا داد؟
 - برای نمایش داده‌های زیاد و انتخاب کاربر، چه پیشنهادی دارید؟
- هدف از این واحد شایستگی به کارگیری کنترل‌های پیشرفته در پروژه‌های ویندوزی است.

استاندارد عملکرد

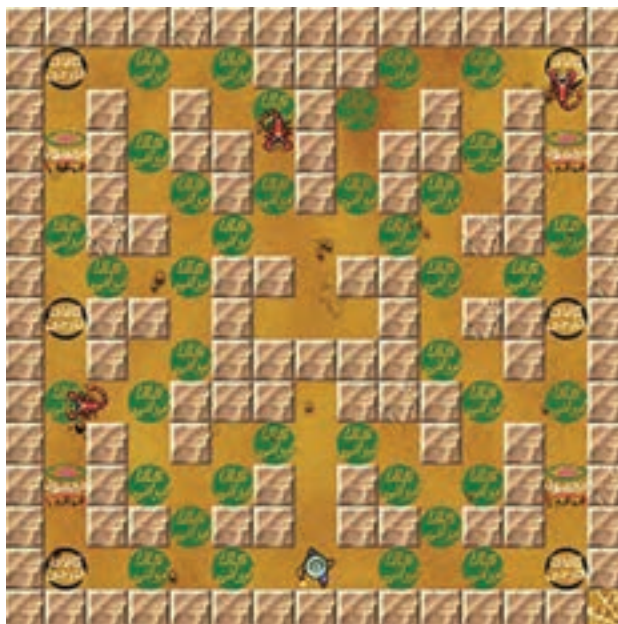
با استفاده از دانش انواع کنترل‌های پیشرفته و گروه‌بندی آنها، پروژه چند فرمی ایجاد کرده، برای رویدادهای آن برنامه بنویسد.

زمان سنج

شکل ۲۸ و ۲۹ تصویر بازی پک من (PAC-MAN) و کالای ایرانی است. حتماً تاکنون با این بازی یا بازی‌های شبیه به آن برخورد کرده‌اید. به نظر شما چگونه می‌توان حرکت شخصیت‌های این بازی را طراحی کرد؟ اگر قرار باشد شما این بازی را در VS بنویسید، برای حرکت‌های سریع و مداوم پک من از چه چیزی استفاده می‌کنید؟



شکل ۲۸- تصویر بازی PAC-MAN

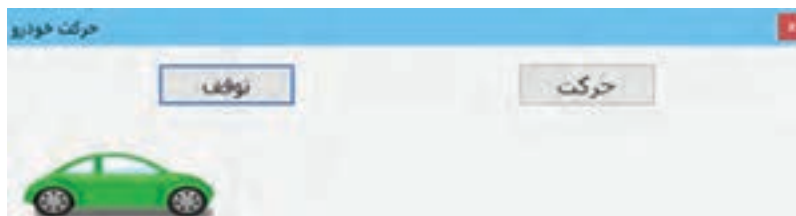


شکل ۲۹- تصویر بازی کالای ایرانی

برنامه‌های زیادی، مانند این بازی وجود دارند که دستور یا دستوراتی باید در یک فاصله زمانی مشخص تکرار شوند. مانند زمانی که می‌خواهیم یک عبارت هرچند ثانیه یک‌بار روی صفحه نمایش داده شود. در چنین مواقعی می‌توانیم از زمان‌سنج (Timer) استفاده کنیم. مهم‌ترین ویژگی زمان‌سنج Interval است که فاصله زمانی تکرار دستور برحسب میلی ثانیه را تنظیم می‌کند. رویداد Tick تنها رویداد زمان‌سنج است. این رویداد در فاصله زمانی مشخص شده در ویژگی Interval رخ می‌دهد و دستورات متد این رویداد اجرا می‌شوند.

کارگاه ۱ رویداد Tick

می‌خواهیم برنامه‌ای بنویسیم که با کلیک دکمه «حرکت» خودرو در راستای افقی حرکت کند.



شکل ۳۰- فرم برنامه حرکت خودرو

۱ پروژه‌ای با نام MoveCar ایجاد کنید.

۲ فرم مناسب را طراحی کنید.

دو کنترل دکمه با نام btnStart و btnStop و یک کنترل کادرتصویر با نام picCar و یک زمان‌سنج به فرم اضافه کنید (شکل ۳۰). آیا زمان‌سنج روی فرم دیده می‌شود؟
زمان‌سنج نیز یک component است.

۳ متد رویداد Tick زمان‌سنج را بنویسید.

روی زمان‌سنج دابل کلیک کنید تا متد رویداد Tick ایجاد شود. کد زیر را بنویسید.

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

```
{  
    picCar.Left += 5;  
}
```

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

۴ متد رویداد کلیک دکمه btnStart را بنویسید.

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

```
{  
    timer1.Enabled = true;  
}
```

برنامه را اجرا کنید. آیا خودرو حرکت می‌کند؟ چرا؟

۵ برای توقف خودرو متد رویداد کلیک دکمه btnStop را بنویسید.

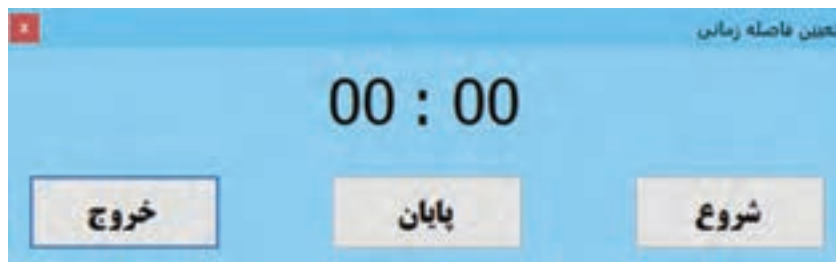
۶ امکان کم و زیاد کردن سرعت خودرو را به برنامه اضافه کنید.

چگونه می‌توان سرعت خودرو را کم یا زیاد کرد؟ کنترل‌های مورد نیاز را به فرم اضافه کرده، کدنویسی کنید.

- برنامه را طوری تغییر دهید که در صورت برخورد خودرو به فرم هم، خودرو متوقف شود.
- برای حرکت کردن خودرو به سمت عقب نیز برنامه بنویسید.

کارگاه ۲ تعیین زمان سپری شده

در بسیاری از برنامه‌ها نیاز داریم که مدت زمان سپری شده را تعیین کنیم. می‌خواهیم برنامه‌ای بنویسیم که مدت زمان سپری شده بعد از کلیک دکمه را نمایش دهد (شکل ۳۱).



شکل ۳۱- فرم برنامه Chronometer

کنجکاوی



فعالیت منزل



۱ پروژه‌ای با نام **Chronometer** ایجاد کنید.

۲ فرم مناسب را طراحی کنید (شکل ۳۱).

با استفاده از سه کنترل برچسب و سه دکمه، یک زمان‌سنج ساده با برچسب دقیقه‌شمار (lblMinute) و ثانیه‌شمار (lblSecond) طراحی کنید.

۳ برای ایجاد ثانیه‌شمار یک زمان‌سنج (timer1) روی فرم قرار دهید.

ویژگی Interval زمان‌سنج را باید با چه عددی مقداردهی کنید؟

۴ کد ثانیه‌شمار را در رویداد Tick زمان‌سنج بنویسید.

متغیرهای second و minute را برای ذخیره ثانیه و دقیقه تعریف کنید.

```
public partial class Form1 : Form
{
    .....
    public Form1()
    {
        InitializeComponent();
    }
    private void timer1_Tick (object sender, EventArgs e)
    {
        second += 1;
        if (second == 60)
        {
            second = 0;
            minute += 1;
        }
        lblSecond.Text = second.ToString();
        lblMinute.Text = minute.ToString();
    }
}
```

۵ دستور فعال‌سازی زمان‌سنج را در متد کلیک دکمه «شروع» (btnStart) بنویسید.

علاوه بر فعال‌سازی زمان‌سنج در متد رویداد کلیک دکمه شروع باید زمان محاسبه شده را با مقداردهی متغیرهای minute و second صفر کرده، مقدار آنها را در برچسب‌ها نمایش دهید.

۶ زمان‌سنج را در متد رویداد کلیک دکمه «توقف» (btnStop) غیرفعال کنید.

برنامه را اجرا کنید.

ساعت سیستم را در برنامه خود نمایش دهید.

راهنمایی:

برای نمایش ساعت سیستم در برنامه از یک زمان‌سنج و برچسب استفاده کنید و برای نمایش زمان در برچسب از دستور زیر استفاده کنید که ساعت فعلی سیستم را به شکل long time روی برچسب نشان می‌دهد.

```
lblTime.Text = DateTime.Now.ToLongTimeString();
```

فعالیت کارگاهی





آنچه آموختم:

۱.
۲.
۳.

جدول ارزشیابی شایستگی‌های غیرفنی، ایمنی و بهداشت و توجهات زیست محیطی



شایستگی‌ها	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
شایستگی‌های غیرفنی	مدیریت کیفیت، پایش شاخص‌های کیفیت - زبان فنی	قابل قبول	۲ انتخاب کنترل‌های کاربرپسند - توجه به نحوه چینش کنترل‌ها در فرم و تنظیم ویژگی‌های آنها از لحاظ دسترسی راحت به آنها و زیبایی ظاهر واسط کاربری - خلاقیت در طراحی واسط گرافیکی کاربری	
ایمنی و بهداشت	رعایت ارگونومی			
توجهات زیست محیطی	حفاظت از تجهیزات کارگاه	غیر قابل قبول	۱ توجه به ایمنی و بهداشت محیط کارگاه	
نگرش	دقت در دسته بندی و چینش خلاقانه کنترل‌ها در فرم برای طراحی واسط گرافیکی کاربرپسند			

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

ارزشیابی مرحله ۱



مراحل کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
به کارگیری زمان سنج	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم افزار IDE برنامه نویسی روی آن نصب است زمان: ۲۰ دقیقه	بالاتر از حد انتظار	۳ اضافه کردن زمان سنج به واسط کاربری و تنظیم ویژگی‌های آن - نوشتن کد برنامه - رفع خطای برنامه	
		حد انتظار	۲ اضافه کردن زمان سنج به واسط کاربری و تنظیم ویژگی‌های آن - نوشتن کد برنامه	
		پایین تر از حد انتظار	۱ اضافه کردن زمان سنج به واسط کاربری و تنظیم ویژگی‌های آن	

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

شکل ۳۲ فرمی را نشان می‌دهد که در آن کاربر باید مکانی که در تصویر نمایش داده شده را تشخیص دهد و از بین چندین گزینه، گزینه درست را انتخاب کند. همچنین مکان‌های تاریخی شهر اصفهان را علامت بزند. در قسمت اول شکل ۳۲ که با عنوان مکان‌های دیدنی ایران مشخص شده است، از بین ۴ انتخاب، شما تنها می‌توانید یک گزینه را انتخاب کنید. نام کنترلی که در این قسمت استفاده شده، دکمه رادیویی (RadioButton) است. کنترل دکمه رادیویی برای ایجاد مجموعه‌ای از گزینه‌ها به کار می‌رود که در این مجموعه فقط یک گزینه را می‌توان انتخاب کرد.



شکل ۳۲- کنترل‌های دکمه رادیویی و کادر علامت

در قسمت دوم شکل ۳۲ شما اجازه انتخاب یک یا چند گزینه را دارید. کنترلی که در این قسمت استفاده شده، کنترل کادر علامت (CheckBox) است. با کلیک روی این کنترل، علامت \surd در آن ظاهر و با کلیک دوباره این علامت حذف می‌شود.

جدول ۶- برخی ویژگی‌های مشترک کنترل‌های دکمه رادیویی و کادر علامت

نام ویژگی	عملکرد
Checked	انتخاب یا عدم انتخاب کنترل
CheckAlign	محل قرار گرفتن کادر انتخاب کنترل
Enabled	فعال بودن یا فعال نبودن کنترل
Text	متن کنار کنترل

کارگاه ۳ پاسخ برنامه به انتخاب کادر علامت و دکمه رادیویی

می‌خواهیم برنامه‌ای بنویسیم که در فرم ورود کاربر نمایش گذرواژه با خواست کاربر ممکن شود و با توجه به نوع کاربر که می‌تواند مدیر، معلم یا دانش‌آموز باشد هنگام ورود پیام خوش‌آمدگویی مناسب نمایش داده شود.

۱ پروژۀ Login را باز کنید.

۲ فرم مناسب را طراحی کنید (شکل ۳۳).

یک کنترل کادر علامت به نام chkPassword به فرم اضافه کنید. سه کنترل دکمه رادیویی با نام rdoManager، rdoTeacher و rdoStudent به فرم اضافه کنید.



شکل ۳۳- سفارشی کردن فرم ورود کاربر

۳ دکمه رادیویی rdoManager را انتخاب کنید.

برای انتخاب دکمه‌های رادیویی باید ویژگی Checked آن را true قرار دهید.

۴ متد رویداد CheckedChanged کادر علامت را بنویسید.

با دابل کلیک روی کادر علامت، متد رویداد CheckedChanged کادر علامت را ایجاد کنید و کدهای زیر را بنویسید.

```
private void chkPassword_CheckedChanged (object sender, EventArgs e)
{
    if (chkPassword.Checked==true)
        txtPassword.PasswordChar='\0';
    else
        txtPassword.PasswordChar='*';
}
```

برنامه را اجرا کنید. عملکرد متد رویداد chkPassword_CheckedChanged چیست؟

۵ متد رویداد کلیک دکمه «ورود» را بنویسید.

```
private void btnLogin_Click (object sender, EventArgs e)
{
    if (rdoManager.Checked)
        MessageBox.Show ("مدیر خوش آمدید");
}
```

```
else if (rdoTeacher.Checked)
    MessageBox.Show ("معلم خوش آمدید");
else if (rdoStudent.Checked)
    MessageBox.Show ("هنرجو خوش آمدید");
}
```

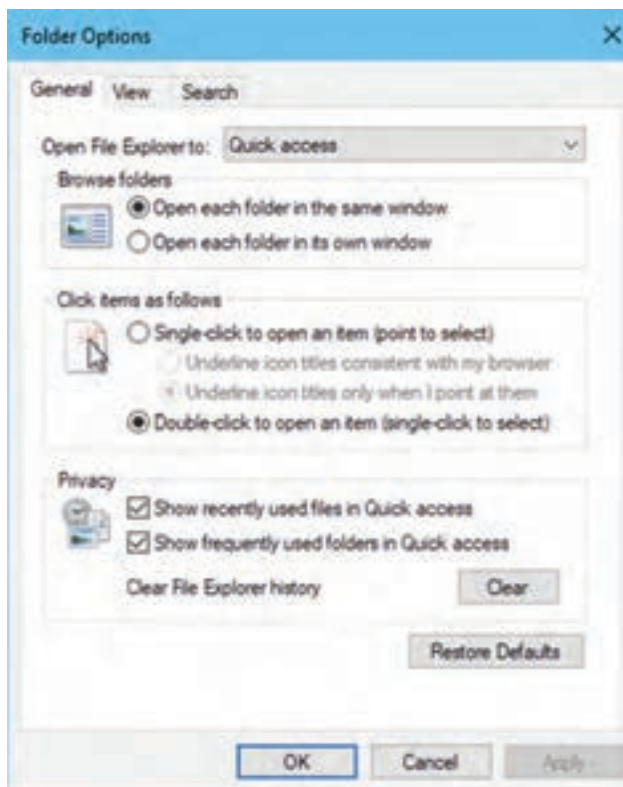
برنامه را اجرا کنید. با انتخاب انواع کاربر دکمه ورود را کلیک کنید.

کاراکتر '\0' یک کاراکتر null است. با قرار دادن کاراکتر '\0' در ویژگی PasswordChar کاراکترهای نوشته شده در کادر دیده می شوند.

یادداشت



دسته بندی کنترل ها



شکل ۳۴ پنجره تنظیم مشخصات پوشه در ویندوز را نشان می دهد. به نظر شما چگونه می توانید هم زمان دو دکمه رادیویی را در این پنجره انتخاب کنید؟ در این پنجره گزینه های مربوط به تغییر نحوه نمایش پوشه ها در یک گروه و روش انتخاب کردن پوشه در گروه دیگری قرار گرفته است تا کاربر بتواند از هر گروه یک گزینه را انتخاب کند. برای دسته بندی کردن کنترل ها در برنامه، از کنترل های گروه بندی استفاده می شود که به آنها Container می گویند.

شکل ۳۴- گروه بندی دکمه های رادیویی

کارگاه ۴ کنترل کادر گروهی

در این پروژه می‌خواهیم یک سرگرمی با تصاویر مکان‌های دیدنی و تاریخی طراحی کنیم. یک تصویر از بین ۸ تصویر به صورت تصادفی در کادر تصویر قرار می‌گیرد. تصویر با قرار گرفتن ۹ برچسب روی آن از دید کاربر پنهان می‌شود. کاربر برای مشاهده هر یک از قسمت‌های تصویر، کادر علامت مربوط به آن را انتخاب می‌کند و به محض حدس تصویر نام مکان و شهر یا استان تصویر را در دو گروه دکمه‌های رادیویی انتخاب می‌کند.

۱ یک پروژه با نام **Tour** ایجاد کنید.

۲ **فرم برنامه را طراحی کنید (شکل ۳۵).**

یک کادر تصویر روی فرم قرار داده، تصویری داخل آن وارد کنید. با ۹ برچسب کادر تصویر را پر کنید تا تصویر به طور کامل مخفی شود.

دو کنترل کادرگروهی (GroupBox) برای دسته‌بندی کادرهای علامت و دکمه‌های رادیویی روی فرم قرار داده، ویژگی **Text** این کنترل‌ها را به «نمایش قسمتی از تصویر» و «نام تصویر» تغییر دهید.

۹ کنترل کادر علامت را داخل کادر گروهی «نمایش قسمتی از تصویر» قرار داده، عبارت «شماره ۱» تا «شماره ۹» را در ویژگی **Text** کادرهای علامت بنویسید. ۸ دکمه رادیویی در کادر گروهی «نام تصویر» قرار دهید. کنترل پنل (Panel) را از جعبه ابزار به فرم اضافه کرده، دکمه‌های رادیویی نام شهر یا استان مکان تاریخی را روی آن قرار دهید.

۳ **ویژگی کادرهای گروهی را تغییر دهید.**

ویژگی **RightToLeft** کادر گروهی «نمایش

قسمتی از تصویر» را **Yes** قرار دهید. به کادرهای علامت دقت کنید. چه تغییری در ویژگی‌های آنها اتفاق می‌افتد؟ قلم کادر گروهی «نام تصویر» را تغییر دهید، سپس ویژگی قلم دکمه‌های رادیویی را بررسی کنید. چه تغییری مشاهده می‌کنید؟ از این تغییرات چه نتیجه‌ای می‌گیرید؟

۴ **ویژگی‌های کنترل پنل را تنظیم کنید.**

ویژگی قلم پنل را تغییر دهید. چه تغییری در دکمه‌های رادیویی داخل پنل مشاهده می‌کنید؟ چگونه می‌توانید یک عنوان برای کنترل پنل بنویسید؟

۵ **پنل را روی فرم جابه‌جا کنید.**

دکمه‌های رادیویی روی پنل چه تغییری می‌کنند؟

۶ **کد متد رویداد CheckedChanged کادرهای علامت را بنویسید.**

متد رویداد **CheckedChanged** کادرهای علامت را برای نمایش بخشی از تصویر بنویسید که با برچسب پوشیده شده است.

باید قسمتی از تصویر که با برچسب پنهان شده است، با انتخاب کادر علامت هم شماره با برچسب نمایش داده شود. با مخفی کردن برچسب، قسمتی از تصویر که زیر برچسب قرار دارد، نمایان می‌شود.



شکل ۳۵- دسته‌بندی کنترل‌ها

```
private void checkBox1_CheckedChanged (object sender, EventArgs e)
{
    label1.Visible = false;
}
```

متد رویداد کلیک دکمه «تغییر تصویر» را کامل کنید تا با هر بار کلیک روی دکمه، تصویر جدیدی نمایش داده شود.

برای تولید اعداد تصادفی از کلاس **Random** استفاده می‌شود. این کلاس دارای متد **Next** برای تولید اعداد تصادفی صحیح است. جدول ۷ را به کمک راهنمای مایکروسافت و با توجه به شکل‌های مختلف این متد کامل کنید.

جدول ۷- شکل‌های مختلف متد **Next**

شرح	شکل
	Next ()
	Next (int max)
	Next (int min, int max)

برای استفاده از این متد ابتدا باید متغیری از نوع کلاس **Random** ایجاد کنید.

```
Random rnd = new Random ();
```

سپس با استفاده از متد **Next** عدد تصادفی صحیح تولید کنید.

```
n = rnd.Next (1,9); // ۱ تا ۸ تصادفی صحیح از
```

۸ تصویر از مکان‌های تاریخی در پوشه **Debug** پروژه با نام شماره‌های ۱ تا ۸ (1.jpg, 2.jpg, ...) قرار دهید. با تولید یک عدد تصادفی بین ۱ تا ۸، متدی بنویسید که یک تصویر را از بین ۸ تصویر انتخاب کند. از آنجا که در این برنامه عدد تصادفی تولید شده در متد رویداد کلیک دکمه تغییر تصویر و دکمه پاسخ صحیح استفاده می‌شود، متغیر **n** و **rnd** را در کلاس **Form** تعریف می‌کنیم. قطعه کد زیر را در کلاس **Form** بنویسید.

```
int n ;
```

```
Random rnd= new Random();
```

```
private void btnChangePicture_Click (object sender, EventArgs e)
{
    n = rnd.Next (1,9);
    string s=n.ToString () + ".jpg";
    pictureBox1.ImageLocation = s;
}
```

۸ متد رویداد کلیک دکمه «پاسخ صحیح» را بنویسید.

با استفاده از عدد تصادفی تولید شده در متد کلیک دکمه «تغییر تصویر»، متد رویداد کلیک دکمه «پاسخ صحیح» را کامل کنید.

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

```
{
    switch (n)
    {
        case 1:
            MessageBox.Show ("پاسخ صحیح", "پاسخ صحیح: سی و سه پل در استان اصفهان",
            MessageBoxButtons.OKCancel);
            break;
        case 2:
            MessageBox.Show ("پاسخ صحیح", "پاسخ صحیح: دروازه قرآن در شیراز",
            MessageBoxButtons.OKCancel);
            break;
        case 3:
            .....
            .....
            .....
    }
}
```

- برنامه Tour را تغییر دهید تا به جای انتخاب کادر علامت با کلیک روی هر برجسب بخشی از تصویر ظاهر شود.

- با کمک هم گروهی خود و هنرآموز جدول زیر را کامل کنید.

فعالیت گروهی



نام کنترل	کاربرد	ویژگی‌های مهم	رویدادهای مهم
دکمه رادیویی (RadioButton)		Checked CheckAlign	CheckedChanged ()
کادر علامت (CheckBox)		Checked
کادر گروهی (GroupBox)		
پنل (Panel)		

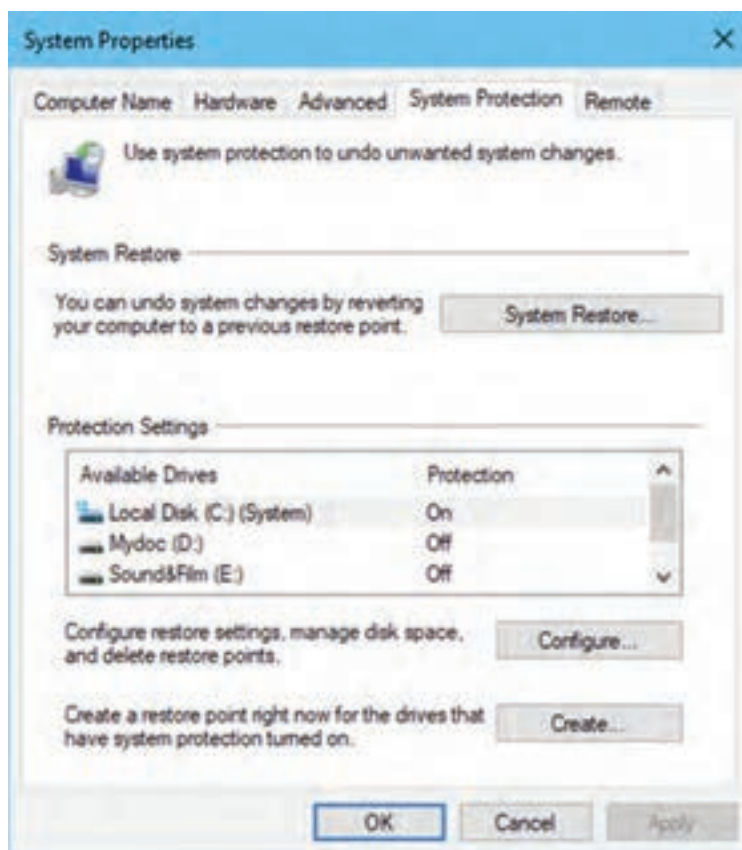


اگر بخواهیم از متغیری در بخش‌های مختلف کلاس Form استفاده کنیم، آن را در کلاس Form تعریف می‌کنیم.

زمانی که یک کنترل در داخل پنل یا کادر گروه‌بندی قرار می‌گیرد، آن کنترل به فرزند پنل یا کادر گروهی تبدیل می‌شود و این ارتباط بین والد و فرزند اجازه می‌دهد که مقادیر ویژگی‌های عمومی بین آنها به اشتراک گذاشته شود. برای مثال اگر مقدار ویژگی Enabled پنل را برابر با false قرار دهیم، نه تنها خود پنل بلکه تمامی کنترل‌های داخل آن نیز غیرفعال می‌شوند.

کنترل سربرگ (TabControl)

گاهی اوقات مجبوریم اطلاعات مختلفی از یک موضوع مشخص را دسته‌بندی کرده، در یک فرم نمایش دهیم و همه این اطلاعات نمی‌توانند در یک صفحه قرار گیرند. شکل ۳۶ پنجره مشخصات سیستم را نشان می‌دهد که شامل قسمت‌های مختلف است. هنگام کار کردن با ویندوز و برنامه‌های ویندوزی با پنجره‌های شبیه شکل ۳۶ زیاد برخورد کرده‌ایم. این نوع پنجره‌ها شامل تعدادی سربرگ (tab) هستند که با کلیک روی هر سربرگ، محتویات آن نمایش داده می‌شود. در VS برای ایجاد سربرگ در برنامه از کنترل TabControl استفاده می‌کنیم.



شکل ۳۶- کنترل سربرگ در پنجره‌های ویندوزی

کارگاه ۵ دسته‌بندی کنترل‌ها با کنترل Tab

می‌خواهیم با استفاده از کنترل Tab، دو صفحه برای ورود اطلاعات شخصی و اطلاعات تحصیلی ایجاد کنیم.

شکل ۳۷- سربرگ دریافت مشخصات فردی

شکل ۳۸- سربرگ دریافت اطلاعات تحصیلی

۱ پروژه‌ای با نام PersonalInfo ایجاد کنید.

۲ از جعبه ابزار یک TabControl روی فرم قرار دهید.

ویژگی Dock کنترل Tab را از پنجره ویژگی‌ها طوری تنظیم کنید که این کنترل کل فرم را پر کند. ویژگی RightToLeftLayout را true قرار دهید. چه تغییری اتفاق می‌افتد؟


۳ تنظیمات سربرگ اول کنترل Tab را انجام دهید.

روی سربرگ اول که با tabPage1 مشخص شده است کلیک کرده، سپس روی صفحه فرم آن کلیک کنید تا پنجره ویژگی‌های سربرگ اول فعال شود. عنوان سربرگ اول را در ویژگی Text به «مشخصات فردی» تغییر دهید. دو کنترل پنل در سربرگ اول قرار داده، کنترل‌های لازم را برای دریافت مشخصات فردی روی آن قرار دهید (شکل ۳۷).

۴ تنظیمات سربرگ دوم کنترل Tab را انجام دهید.

عنوان سربرگ دوم را به «اطلاعات تحصیلی» تغییر دهید و کنترل‌های لازم را برای دریافت اطلاعات تحصیلی در سربرگ دوم قرار دهید.

۵ یک سربرگ دیگر به کنترل Tab اضافه کنید.

از پنجره ویژگی‌های کنترل Tab، در ستون مقدار ویژگی TabPages روی علامت  کلیک کنید تا کادر محاوره‌ای ویرایش سربرگ‌های کنترل Tab (TabPage Collection Editor) باز شود. در این کادر امکان حذف و اضافه کردن سربرگ به کنترل و همچنین تنظیم ویژگی‌های سربرگ‌ها وجود دارد. در کادر باز شده گزینه Add را انتخاب کنید. چه اتفاقی می‌افتد؟ نتیجه را در فرم مشاهده کنید.

۶ عنوان، قلم و رنگ زمینه سربرگ سوم را از طریق ویژگی **TabPage** کنترل **Tab** تغییر دهید. از طریق ویژگی **TabPage** کادر محاوره‌ای ویرایش سربرگ‌های کنترل **Tab** را باز کنید و عنوان، قلم و رنگ زمینه سربرگ سوم را تغییر دهید. تغییرات را در برنامه مشاهده کنید.

۷ سربرگ سوم را حذف کنید.

۸ کد زیر را در متد رویداد کلیک دکمه «ادامه» برای رفتن به سر برگ اطلاعات تحصیلی بنویسید.
`tabControl1.SelectedTab = tabPage2;`

کاربرد ویژگی **SelectedTab** چیست؟

۹ متد رویداد کلیک دکمه «خروج» هر دو سربرگ را بنویسید.

برداشت



آنچه آموختم:

۱.
۲.
۳.

ارزشیابی مرحله ۲



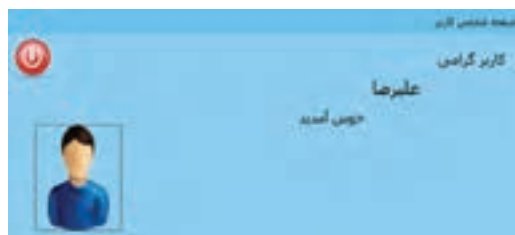
مراحل کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
به کارگیری کنترل‌های انتخاب و گروه‌بندی	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۲۰ دقیقه	بالاتر از حد انتظار	افزودن کنترل‌های انتخاب و گروه‌بندی و تنظیم ویژگی‌های آنها - نوشتن کد برنامه - رفع خطای برنامه	۳
		در حد انتظار	افزودن کنترل‌های انتخاب و گروه‌بندی و تنظیم ویژگی‌های آنها - نوشتن کد برنامه	۲
		پایین‌تر از حد انتظار	افزودن کنترل‌های انتخاب و گروه‌بندی	۱

کارگاه ۶ ایجاد پروژه‌های چند فرمی

همیشه با یک فرم نمی‌توانیم برنامه را کامل کنیم. در بسیاری از مواقع باید از یک فرم، فرم دیگری را فراخوانی کنیم و در واقع بین فرم‌ها حرکت کنیم. فرم‌های متنوع برای دسته‌بندی موضوعات برنامه به کاربر کمک می‌کنند که مسیر حرکت برنامه را سریع‌تر پیدا کند. قصد داریم برنامه‌ای بنویسیم که کاربر با ورود نام کاربری و گذرواژه به فرم صفحه شخصی کاربر منتقل شود و در آن فرم، پیام خوش‌آمدگویی به همراه نام کاربر دیده شود.



شکل ۳۹- فرم ورود کاربر (LoginForm)



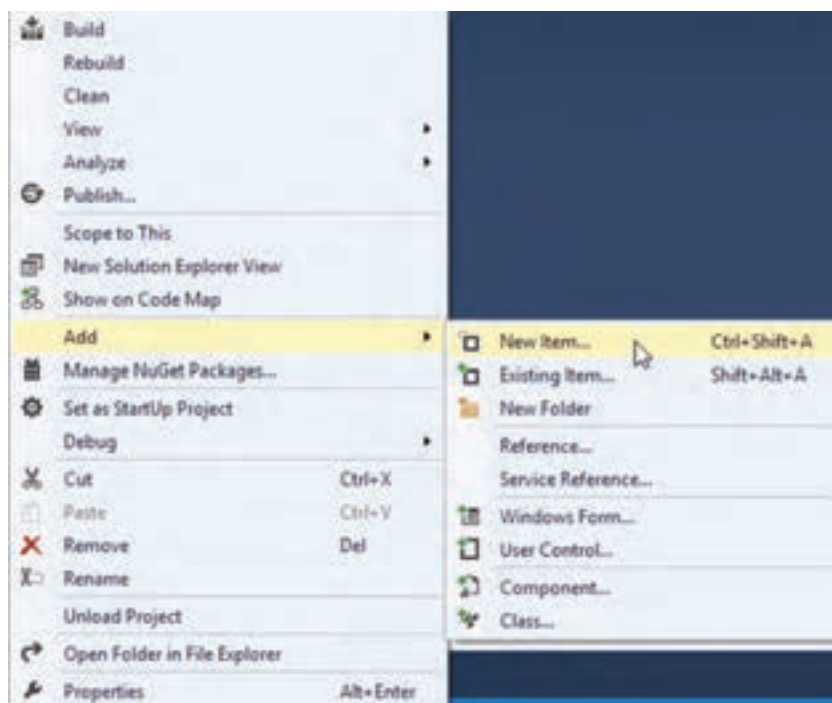
شکل ۴۰- فرم صفحه شخصی کاربر (MainForm)

۱ پروژه ویندوزی Login را باز کنید.

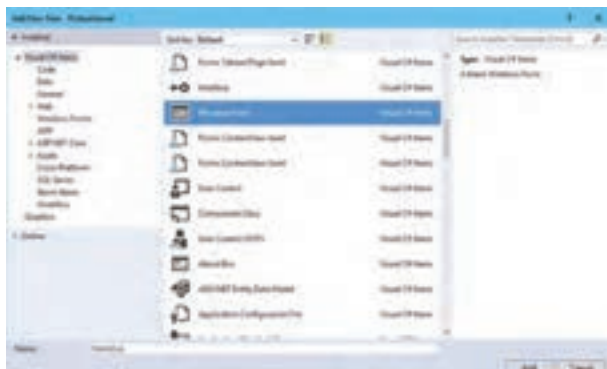
نام Form1 را به LoginForm تغییر دهید.

۲ فرم جدیدی با نام MainForm به برنامه اضافه کنید.

روی نام پروژه در پنجره Solution Explorer راست کلیک کنید و گزینه Add/New Item را انتخاب کنید (شکل ۴۱) تا کادرمحاوره‌ای Add New Item باز شود (شکل ۴۲).



شکل ۴۱- منوی اضافه کردن فرم به پروژه



شکل ۴۲- اضافه کردن فرم جدید به برنامه

با انتخاب Windows Form در قسمت Name این کادر محاوره‌ای نام فرم نمایش داده می‌شود. نام فرم را به MainForm تغییر دهید. دقت کنید که نام فرم، نام کلاس فرم هم خواهد بود. به پنجره Solution Explorer دقت کنید. فرم MainForm به پنجره اضافه شده است. برنامه را با فشردن کلید میان‌بر F5 اجرا کنید. پس از اجرای برنامه کدام فرم نمایش داده می‌شود؟

۲ MainForm را فرم شروع برنامه قرار دهید.

در پنجره Solution Explorer روی پرونده Program.cs دابل کلیک کنید، تا باز شود.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;

namespace Login
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new LoginForm());
        }
    }
}
```

در متد Main در این پرونده، متد Run از کلاس Application فراخوانی می‌شود. آرگومان ورودی متد Run در اینجا، دستور ایجاد فرمی است که در شروع برنامه نمایش داده می‌شود. اولین فرم که همراه با ایجاد پروژه ویندوزی به صورت خودکار ایجاد می‌شود، به صورت پیش‌فرض فرم شروع برنامه است (LoginForm).

`Application.Run (new LoginForm ());`

نام فرم دوم (MainForm) را به جای LoginForm در ورودی متد Run بنویسید.

`Application.Run (new MainForm ());`

برنامه را اجرا کنید. کدام فرم به عنوان فرم شروع برنامه نمایش داده می‌شود؟

۴ فرم MainForm را مطابق شکل ۴۰ طراحی کنید.

یک کنترل برچسب با نام lblWelcome برای نمایش پیام خوش آمدگویی به کاربر به فرم MainForm اضافه کنید.

۵ امکان دسترسی به اطلاعات یک فرم را برای فرم‌های دیگر فراهم کنید.

در LoginForm دو متغیر succeeded و userName را مطابق کد زیر اعلان کنید.

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

```
{
```

```
    public bool succeeded;
```

```
    public string userName;
```

برای ورود به صفحه شخصی کاربر (MainForm) باید نام کاربری و گذرواژه معتبر باشد. متغیر succeeded برای این منظور تعریف شده است. برای پیام خوش آمدگویی به نام کاربر نیاز داریم، متغیر userName برای ذخیره نام کاربر تعریف شده است. این اطلاعات در LoginForm قرار دارد برای اینکه فرم‌های دیگر برنامه هم به این اطلاعات دسترسی داشته باشند، سطح دسترسی این متغیرها را public قرار می‌دهیم. به صورت پیش فرض سطح دسترسی متغیرهای یک فرم private است و فقط خود فرم به آنها دسترسی دارد.

به متغیرهایی که در کلاس تعریف می‌شوند فیلد (field) می‌گویند. فیلدها در کلاس به صورت خودکار مقداردهی اولیه می‌شوند. مقدار اولیه فیلد succeeded و userName چیست؟

۶ در LoginForm متد رویداد کلیک دکمه «ورود» را بنویسید.

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

```
{
```

```
    if (txtUserName.Text == "علیرضا" && txtPassword.Text == "123")
```

```
    {
```

```
        succeeded = true;
```

```
        userName = txtUserName.Text;
```

```
        this.Close();
```

```
    }
```

```
    else
```

```
        MessageBox.Show("خطا", "نام کاربری یا گذرواژه معتبر نیست", MessageBoxButtons.OK, MessageBoxIcon.Error);
```

```
}
```

۷ متد رویداد Load فرم صفحه شخصی کاربر را بنویسید.

قبل از نمایش فرم صفحه شخصی کاربر باید نام کاربری و گذرواژه از کاربر دریافت شده، صحت آنها بررسی شود. روی فرم دابل کلیک کرده، متد رویداد Load را بنویسید.

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

```
{
```

```

LoginForm frm = new LoginForm();
frm.ShowDialog();
if (frm.succeeded)
{
    lblWelcome.Text = "خوش آمدید " + frm.userName + " کاربر گرامی ";
}
else
    this.Close();
}
    
```

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

کنجکاوی



- متن `lblWelcome` را به گونه‌ای تغییر دهید که ظاهر `UserForm` مشابه شکل ۴۰ شود.
 - برای نمایش فرم می‌توان از متدهای `Show` یا `ShowDialog` استفاده کرد. به کمک هنرآموز خود تفاوت این دو متد را بررسی کنید.
 - آیا می‌توان در این برنامه به جای متد `ShowDialog` از متد `Show` استفاده کرد؟ چرا؟

فعالیت کارگاهی



برنامه کارگاه ۶ را توسعه دهید تا اگر کاربر سه بار نام کاربری و گذرواژه را اشتباه وارد کرد پس از نمایش پیام مناسب، ورود اطلاعات به مدت ۳۰ ثانیه امکان پذیر نباشد.

فعالیت منزل



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

پژوهش



آنچه آموختم:

برداشت



۱.
۲.
۳.



مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
به کارگیری چند فرم در پروژه	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۱۵ دقیقه	بالاتر از حد انتظار	افزودن فرم جدید به پروژه - نوشتن کد برنامه - تعیین فرم اصلی پروژه - انتقال اطلاعات از یک فرم به فرم دیگر - رفع خطای برنامه	۳
		در حد انتظار	افزودن فرم جدید به پروژه - تعیین فرم اصلی پروژه - نوشتن کد برنامه	۲
		پایین تر از حد انتظار	افزودن فرم جدید به پروژه	۱

کنترل‌های کادر فهرست (ListBox) و کادر ترکیبی (ComboBox)



شکل ۴۳- کنترل کادر ترکیبی در برنامه‌های ویندوزی



شکل ۴۴- کنترل کادر فهرست در برنامه‌های ویندوزی

فرض کنید در یک برنامه ویندوزی از بین نام ۳۰ هنرجو، می‌خواهیم یکی را انتخاب کنیم. به نظر شما کدام یک از کنترل‌ها برای طراحی این برنامه مناسب‌تر هستند؟ آیا لازم است نام ۳۰ هنرجو به‌طور کامل روی فرم نمایش داده شود؟ شکل ۴۳ یک نمونه از پنجره‌هایی است که در ویندوز دیده‌اید. اگر قرار بود نام همه درایوها همیشه در این پنجره نمایش داده شوند، علاوه بر اینکه فضای زیادی اشغال می‌شد، برنامه زیبایی خود را نیز از دست می‌داد. در چنین مواردی از کنترل کادر فهرست (ListBox) و یا کادر ترکیبی (Combobox) استفاده می‌شود.

کنترل کادر فهرست به کاربر امکان می‌دهد که از بین عناصری که در فهرست وجود دارد، یک عنصر را انتخاب کند (شکل ۴۴). کادرهای انتخابی در دو شکل ۴۳ و ۴۴ را با هم مقایسه کنید. چه تفاوتی بین این دو کادر می‌بینید؟

کادری که در شکل ۴۳ نشان داده شده است، کادر ترکیبی است. این کادر از یک کادر فهرست و یک کادر متن تشکیل می‌شود که عنصر انتخاب شده از کادر فهرست در کادر متن قابل نمایش است و ویژگی Text، متن موجود در کادر متن را نشان می‌دهد مهم‌ترین ویژگی‌ها و متدهای کادر فهرست و کادر ترکیبی در جدول ۸ و ۹ نمایش داده شده است.

جدول ۸- ویژگی‌های مهم کادر فهرست و کادر ترکیبی

ویژگی	کاربرد
Items	تعیین عناصر فهرست
Sorted	مرتب‌سازی فهرست
Items.Count	تعداد عناصر فهرست
SelectedIndex	اندیس عنصر انتخاب شده
SelectedItem	عنصر انتخاب شده

جدول ۹- متدهای مهم کادر فهرست و کادر ترکیبی

متد	شکل کلی متد
Add (اضافه کردن به فهرست)	(عنصر متنی) Items.Add. نام کنترل
Remove (حذف عناصر)	(عنصر انتخابی) Items.Remove. نام کنترل
RemoveAt (حذف عناصر)	(اندیس عنصر) Items.RemoveAt. نام کنترل
Clear (حذف همه عناصر)	() Items.Clear. نام کنترل

کارگاه ۷ استفاده از کادر فهرست و کادر ترکیبی

در این کارگاه قصد داریم برای اخذ رأی انتخابات شورای دانش‌آموزی فرمی طراحی کنیم تا دانش‌آموز از بین نامزدهای موجود در کادر ترکیبی تعدادی از آنها را انتخاب کرده، به فهرست منتخبان در کادر فهرست اضافه کند و قابلیت حذف و جست‌وجو نیز داشته باشد.



شکل ۴۵- فرم استفاده از کادر فهرست و کادر ترکیبی

۱ پروژه ویندوزی به نام **StudentElection** ایجاد کنید.

۲ فرم مناسب را طراحی کنید

در کنترل کادر ترکیبی نام ۱۵ نامزد را به دلخواه وارد کنید. از ویژگی **Items** استفاده کنید (شکل ۴۵).

۳ متد رویداد کلیک دکمه **btnAdd** را برای اضافه کردن نامزدها به فهرست منتخبان بنویسید. برای اضافه کردن نامزد به فهرست کدهای زیر را بنویسید.

```
private void btnAdd_Click (object sender, EventArgs e)
{
    if (lstResults.Items.Count<5)
        if (cboCandidates.Text.Length > 0)
            lstResults.Items.Add (cboCandidates.Text);
}
```

با اجرای برنامه چند نامزد را می‌توان به فهرست منتخبان اضافه کرد؟

۴ کد مرحله ۳ را تغییر دهید تا در صورت درخواست اضافه کردن نامزد ششم، پیام مناسب نمایش دهد.

۵ متد رویداد کلیک دکمه «حذف» را بنویسید.

برای حذف از فهرست منتخبان کدهای زیر را بنویسید.

```
private void btnDelete_Click (object sender, EventArgs e)
{
    int index=lstResults.SelectedIndex;
    if (index >= 0)
        lstResults.Items.RemoveAt (index);
}
```

۶ متد رویداد کلیک دکمه «پاک کردن فهرست» را بنویسید.

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

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

فعالیت کارگاهی





فاز ۳:

برای برنامه کتابخانه هنرستان واسط گرافیکی مانند فرم زیر طراحی کنید.
- سربرگ جست‌وجو را برای فراهم کردن امکان جست‌وجو بر اساس نام کتاب، نام مؤلف، موضوع و سال چاپ طراحی کنید.

- در سربرگ اعضا، مشخصات عضو شامل نام و نام خانوادگی، کد عضویت، تلفن، نشانی، کد ملی و عکس مطابق شکل زیر طراحی کنید.

- دکمه ثبت: مشخصات عضو را در آرایه‌های متناظر ذخیره می‌کند.
- دکمه جدید: فرم را برای ثبت مشخصات عضو جدید خالی می‌کند.
- دکمه درج تصویر: عکس مناسب را انتخاب و نمایش می‌دهد و در ضمن آدرس آن را در آرایه مناسب جهت بازیابی ذخیره می‌کند.
- سربرگ کتاب را مشابه سربرگ اعضا برای درج اطلاعات زیر طراحی کنید.
نام کتاب ، مؤلف ، مترجم ، انتشارات ، سال چاپ ، موضوع ، شابک ، عکس کتاب
- در نوار عنوان فرم تاریخ و ساعت دیده شود. از دستور: `Date.Now.ToString()` استفاده کنید.
- فرم Login را برای دریافت نام کاربری و گذرواژه کاربر مدیر به پروژه اضافه کنید.
- در صورتی که در فرم Login بیش از ۳ بار نام کاربری و گذرواژه اشتباه وارد شود، امکان دریافت نام کاربری و گذرواژه به مدت یک دقیقه غیر فعال شود.
- در شروع برنامه سربرگ اعضا و کتاب غیرفعال باشد و با کلیک روی دکمه «ورود مدیر» فرم Login نمایش داده شود و در صورت معتبر بودن نام کاربری و گذرواژه وارد شده در این فرم، سربرگ اعضا و کتاب برای کاربر مدیر فعال شود.



آنچه آموختم:

۱.
۲.
۳.

ارزشیابی مرحله ۴



مراحل کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
استفاده از کنترل‌های فهرست	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۲۰ دقیقه	بالاتر از حد انتظار	افزودن کنترل فهرست به واسط کاربری و تنظیم ویژگی‌های آن - نوشتن کد برای حذف و درج به فهرست - رفع خطای برنامه	۳
		در حد انتظار	افزودن کنترل فهرست به واسط کاربری و تنظیم ویژگی‌های آن - نوشتن کد برای حذف و درج به فهرست	۲
		پایین‌تر از حد انتظار	افزودن کنترل فهرست به واسط کاربری و تنظیم ویژگی‌های آن	۱

معیار شایستگی انجام کار:

کسب حداقل نمره ۲ از مرحله به کارگیری زمان‌سنج

کسب حداقل نمره ۲ از بخش شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش

کسب حداقل میانگین ۲ از مراحل کار

جدول ارزشیابی پایانی

شرح کار:
به کارگیری کنترل زمان سنج
به کارگیری چند فرم در پروژه

به کارگیری کنترل های انتخاب و گروه بندی
استفاده از کنترل های فهرست

استاندارد عملکرد:
با استفاده از دانش انواع کنترل های پیشرفته و گروه بندی آنها، پروژه چند فرمی ایجاد کرده، برای رویدادهای آن برنامه بنویسد.

شاخص ها:

شماره مرحله کار	شاخص های مرحله کار
۱	به کارگیری زمان سنج در واسط کاربری بر حسب نیاز - نوشتن برنامه و رفع خطای آن
۲	به کارگیری کنترل های انتخاب و گروه بندی در واسط کاربری بر حسب نیاز - نوشتن برنامه و رفع خطای آن
۳	به کارگیری چند فرم در برنامه - نوشتن برنامه و رفع خطای آن
۴	به کارگیری کنترل های فهرست در واسط کاربری - نوشتن برنامه و رفع خطای آن

شرایط انجام کار و ابزار و تجهیزات:
مکان: کارگاه رایانه مطابق استاندارد تجهیزات هنرستان ها
تجهیزات: رایانه ای که نرم افزار IDE برنامه نویسی روی آن نصب است.
زمان: ۷۵ دقیقه (به کارگیری زمان سنج ۲۰ دقیقه - به کارگیری کنترل های انتخاب و گروه بندی ۲۰ دقیقه - به کارگیری چند فرم در پروژه ۱۵ دقیقه - استفاده از کنترل های فهرست ۲۰ دقیقه)

معیار شایستگی:

ردیف	مرحله کار	حداقل نمره قبولی از ۳	نمره هنرجو
۱	به کارگیری زمان سنج	۲	
۲	به کارگیری کنترل های انتخاب و گروه بندی	۱	
۳	به کارگیری چند فرم در پروژه	۱	
۴	استفاده از کنترل های فهرست	۱	
شایستگی های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش: مدیریت کیفیت، پایش شاخص های کیفیت - زبان فنی رعایت ارگونومی حفاظت از تجهیزات کارگاه دقت در دسته بندی و چینش خلاقانه کنترل ها در فرم برای طراحی واسط گرافیکی کاربر پسند		۲	
میانگین نمرات			※

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