با ایران توشه
یک قدم جلوتر
باش!

ایران توشه
توشه ای برای موفقیت



ایران توشه، مرجع دانلود فایل های کمک آموزشی دانش آموزان و کنکوری ها

همه چیز
برای موفقیت
تو!

- دانلود رایگان آزمون های آزمایشی قلمچی، خیلی سبز، گاج و ...
- دانلود رایگان گام به گام تمامی دروس
- دانلود رایگان نمونه سوالات امتحانی (نوبت اول و دوم) و امتحانات نهایی
- دانلود رایگان جزوه های کمک آموزشی و شب امتحانی

کافیست هر فایلی که نیاز دارید را در گوگل به این صورت جستجو کنید:

فایل مورد نیاز + ایران توشه

مثال:

دانلود گام به گام دوازدهم تجربی ایران توشه



irantooshe.ir

t.me/irantooshe

گام به گام توسعه برنامه‌سازی و پایگاه داده پایه یازدهم
پودمان سوم

عنوان فصل: طراحی واسط گرافیکی

از صفحه‌ی ۹۹ تا صفحه‌ی ۱۵۲ کتاب

در این فایل پاسخ فعالیت‌ها، کارهای کلاسی، تمرین‌ها و پرسش‌های فصل طراحی واسط گرافیکی آمده است.
مباحث فصل: ایجاد واسط گرافیکی کاربر، ویژگی‌ها و رویدادها، کادرهای محاوره‌ای و کادر پیام، زمان‌سنج، دکمه‌ی رادیویی و کادر علامت، گروه‌بندی و سربرگ، پروژه‌ی چندفرمی، کادر فهرست و کادر ترکیبی

تهیه‌شده در ایران‌توشه، توشه‌ای برای موفقیت

irantooshe.ir

فهرست پرسش‌ها

۱ صفحه ۱۰۲ · فعالیت کارگاهی — اجزای IDE پروژه‌های ویندوزی ۹

۲ صفحه ۱۰۴ · کارگاه ۱ — با RightToLeft=Yes چه تغییری می‌کند؟ ۱۰

۳ صفحه ۱۰۴ · کارگاه ۱ — جدول ۲: BackgroundImageLayout ۱۰

۴ صفحه ۱۰۵ · فعالیت گروهی — جدول ۳: FormBorderStyle ۱۰

۵ صفحه ۱۰۵ · فعالیت گروهی — جدول ۴: ویژگی‌های فرم ۱۱

۶ صفحه ۱۰۵ · فعالیت گروهی — RightToLeftLayout ۱۲

۷ صفحه ۱۰۶ · کارگاه ۲ — عرض برچسب و RightToLeft برچسب ۱۲

۸ صفحه ۱۰۷ · کارگاه ۲ — مقادیر TextAlign ۱۲

۹ صفحه ۱۰۷ · کارگاه ۳ — کادر متن: Name، Text، Ctrl+کشیدن، PasswordChar ۱۳

ایران توشه
توشه ای برای موفقیت

۱۰ صفحه ۱۰۸ · کارگاه ۳ — MaxLength، Cursor، Enabled، Visible ۱۳
۱۱ صفحه ۱۰۸ · فعالیت گروهی — جدول ویژگی‌های کنترل‌ها ۱۴
۱۲ صفحه ۱۰۹ · فعالیت منزل — فرم مشخصات کاربر ۱۵
۱۳ صفحه ۱۰۹ · فعالیت منزل — ماشین حساب شبیه ویندوز ۱۰ ..... ۱۵
۱۴ صفحه ۱۱۰ · فعالیت کارگاهی — خروج از پروژه Login ۱۶
۱۵ صفحه ۱۱۲ · کارگاه ۴ — خروج با this.Close ۱۶
۱۶ صفحه ۱۱۳ · کارگاه ۴ — پاک‌کردن کادرها و Parse ۱۶
۱۷ صفحه ۱۱۳ · کارگاه ۴ — رویداد کلیک ضرب، تفریق و تقسیم ۱۷
۱۸ صفحه ۱۱۴ · کارگاه ۵ — افزودن تصویر به منابع ۱۷
۱۹ صفحه ۱۱۵ · فعالیت کارگاهی — گزینه Add New Image ۱۸
۲۰ صفحه ۱۱۶ · کارگاه ۶ — Import تصویر: Local resource و Project resource file ۱۸

ایران توشه
توشه ای برای موفقیت

۲۱ صفحه ۱۱۶ · کارگاه ۶ — SizeMode و ویژگی Image ۱۸

۲۲ صفحه ۱۱۶ · کارگاه ۶ — دکمه‌های نمایش تصویر ۱ و ۲ ..... ۱۹

۲۳ صفحه ۱۱۷ · فعالیت منزل — کادر تصویر در فرم‌ها ۱۹

۲۴ صفحه ۱۱۷ · کارگاه ۷ — بزرگ‌نمایی و محل تصویر ۲۰

۲۵ صفحه ۱۱۸ · کارگاه ۷ — Left و Top و بزرگ/کوچک‌نمایی ۲۰

۲۶ صفحه ۱۱۸ · کارگاه ۷ — ImageLocation ۲۰

۲۷ صفحه ۱۱۹ · کارگاه ۷ — پیمایش تصاویر: if و پوشه Debug ۲۰

۲۸ صفحه ۱۱۹ · کارگاه ۷ — دکمه‌های قبلی و بعدی ۲۱

۲۹ صفحه ۱۱۹ · فعالیت کارگاهی — Show و Hide ۲۱

۳۰ صفحه ۱۱۹ · فعالیت منزل — بازی ساده: حرکت آدمک ۲۲

۳۱ صفحه ۱۲۱ · کارگاه ۸ — OpenFileDialog: پرونده غیرتصویری ۲۲

ایران توشه
توشه ای برای موفقیت

۳۲ صفحه ۱۲۲ · کارگاه ۸ — انتخاب‌نکردن پرونده و DialogResult ۲۲
۳۳ صفحه ۱۲۲ · کارگاه ۹ — پروژه PictureCity: فرم و انتخاب عکس ۲۳
۳۴ صفحه ۱۲۳ · کارگاه ۹ — عنوان کادر محاوره‌ای (Title) ۲۴
۳۵ صفحه ۱۲۴ · فعالیت گروهی — جدول ۵: پارامترهای MessageBox.Show و شمارشی‌ها ۲۴
۳۶ صفحه ۱۲۴ · کارگاه ۱۰ — اجرای پیام خطای نام ۲۵
۳۷ صفحه ۱۲۵ · کارگاه ۱۰ — پیام خطا برای نام خانوادگی؛ حذف عنوان ۲۵
۳۸ صفحه ۱۲۵ · کارگاه ۱۰ — پاک‌کردن نام با Retry ۲۵
۳۹ صفحه ۱۲۵ · فعالیت کارگاهی — OK/Cancel با دکمه پیش‌فرض Cancel ۲۶
۴۰ صفحه ۱۲۵ · فعالیت منزل — فرم ورود کاربر با پیام ۲۷
۴۱ صفحه ۱۲۹ · کارگاه ۱ — زمان‌سنج و بازی پک‌من ۲۷
۴۲ صفحه ۱۳۰ · کارگاه ۱ — حرکت خودرو با زمان‌سنج (MoveCar) ۲۷

ایران توشه
توشه ای برای موفقیت

۴۳ صفحه ۱۳۰ · فعالیت منزل — توقف در لبه‌ی فرم و حرکت به عقب ۲۸
۴۴ صفحه ۱۳۱ · کارگاه ۲ — زمان‌سنج ساده (Chronometer) ۲۹
۴۵ صفحه ۱۳۱ · فعالیت منزل — نمایش ساعت سیستم ۳۰
۴۶ صفحه ۱۳۳ · فعالیت گروهی — جدول ۶: ویژگی‌های RadioButton و CheckBox ۳۰
۴۷ صفحه ۱۳۴ · کارگاه ۳ — نمایش گذرواژه و نوع کاربر (Login) ۳۱
۴۸ صفحه ۱۳۵ · فعالیت گروهی — انتخاب هم‌زمان دو دکمه‌ی رادیویی ۳۲
۴۹ صفحه ۱۳۶ · کارگاه ۴ — کنترل کادر گروهی (Tour): ویژگی‌ها ۳۲
۵۰ صفحه ۱۳۶ · کارگاه ۴ — رویداد CheckedChanged کادرهای علامت ۳۳
۵۱ صفحه ۱۳۷ · کارگاه ۴ — جدول ۷: شکل‌های متد Next و تغییر تصویر ۳۳
۵۲ صفحه ۱۳۸ · کارگاه ۴ — دکمه‌ی پاسخ صحیح (switch) ۳۴
۵۳ صفحه ۱۳۸ · فعالیت گروهی — تغییر برنامه‌ی Tour با کلیک برچسب ۳۵

ایران توشه
توشه ای برای موفقیت

۵۴ صفحه ۱۳۸ · فعالیت گروهی — جدول کنترل‌ها، ویژگی‌ها و رویدادها ۳۵
۵۵ صفحه ۱۴۰ · کارگاه ۵ — کنترل سربرگ (PersonalInfo) ۳۶
۵۶ صفحه ۱۴۲ · کارگاه ۶ — پروژه‌ی چندفرمی: فرم شروع و فیلدها ۳۶
۵۷ صفحه ۱۴۴ · کارگاه ۶ — رویداد کلیک ورود و رویداد Load فرم صفحه‌ی شخصی ۳۷
۵۸ صفحه ۱۴۵ · فعالیت کارگاهی — نمایش فرم: Show یا ShowDialog ۳۸
۵۹ صفحه ۱۴۵ · فعالیت منزل — قفل ۳۰ ثانیه‌ای پس از سه بار ورود اشتباه ۳۹
۶۰ صفحه ۱۴۵ · پژوهش — معایب سطح دسترسی public و انتقال داده بین فرم‌ها ۴۰
۶۱ صفحه ۱۴۶ · فعالیت گروهی — کادر فهرست یا کادر ترکیبی ۴۱
۶۲ صفحه ۱۴۸ · کارگاه ۷ — انتخابات شورای دانش‌آموزی (StudentElection) ۴۱
۶۳ صفحه ۱۴۸ · فعالیت کارگاهی — مرتب‌سازی کادر ترکیبی و نمایش اندیس انتخاب‌شده ۴۲
۶۴ صفحه ۱۴۹ · پروژه فاز ۳ — واسط گرافیکی برنامه‌ی کتابخانه‌ی هنرستان ۴۲

ایران توشه
توشه ای برای موفقیت

۱ صفحه ۱۰۲ فعالیت کارگاهی

جدول ۱ را کامل کنید.

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.

جدول ۱ـ اجزای IDE پروژه‌های ویندوزی

ردیف	نام پنجره	کاربرد
۱	Toolbox	فهرست کنترل‌ها (Label، Button، TextBox و ...) را در دسته‌های مختلف نمایش می‌دهد تا روی فرم بکشیم.
۲	Solution Explorer	فهرستی از پرونده‌های تشکیل‌دهنده پروژه را نمایش می‌دهد.
۳	Properties	ویژگی‌ها و رویدادهای کنترل (یا فرم) انتخاب‌شده را نمایش می‌دهد و مقداردهی آن‌ها را ممکن می‌کند.
۴	Form	صفحه‌ی طراحی؛ کنترل‌ها روی آن چیده می‌شوند و ظاهر برنامه دیده می‌شود.
۵	Error List	خطاها و هشدارهای کامپایل را نمایش می‌دهد.
۶	Output	پیام‌های ساخت (Build) و اجرای پروژه را نشان می‌دهد.

1. منوی File ← New ← Project ← Windows Forms App، نام پروژه را FirstAPP و مسیر را انتخاب و Create را بزنید.
2. با آیکن سنجاق (Auto Hide) در نوار عنوان Toolbox آن را ثابت کنید؛ با کلیک دوباره روی سنجاق یا دکمه‌ی Close آن را ببندید (از منوی View ← Toolbox دوباره باز می‌شود).
3. با کلیک روی سنجاق Properties آن را به حالت مخفی (Auto Hide) ببرید و با نگه‌داشتن ماوس روی زبانه‌اش یا کلیک دوباره روی سنجاق از حالت مخفی خارج کنید.
4. پنجره‌ها را با کشیدن نوار عنوان به جای مناسب (Toolbox سمت چپ، Solution Explorer و Properties سمت راست) ببرید؛ در صورت به‌هم‌ریختگی از Window ← Reset Window Layout استفاده کنید.

این فایل توسط سایت ایران‌توشه تهیه شده است.
irantooshe.ir

۲ صفحه ۱۰۴ کارگاه ۱

با تغییر مقدار ویژگی RightToLeft به Yes چه تغییری در عنوان فرم مشاهده می‌کنید؟

عنوان فرم (Text) به‌جای سمت چپ نوار عنوان، در سمت راست آن قرار می‌گیرد و جهت نوشتن متن راست‌به‌چپ می‌شود. دکمه‌های کنترل (بستن، بزرگ‌کردن، کوچک‌کردن) در جای خود (سمت راست) می‌مانند، مگر اینکه RightToLeftLayout نیز true شود.

۳ صفحه ۱۰۴ کارگاه ۱

مقادیر دیگر این ویژگی را بررسی کرده، نتیجه را در جدول ۲ بنویسید.

جدول ۲ـ تأثیر مقادیر ویژگی BackgroundImageLayout روی فرم

جدول ۲ـ تأثیر مقادیر ویژگی BackgroundImageLayout

مقدار	نتیجه
None	تصویر با اندازه‌ی اصلی در گوشه‌ی بالا-چپ فرم قرار می‌گیرد؛ اگر کوچک‌تر باشد بقیه‌ی فرم خالی می‌ماند و اگر بزرگ‌تر باشد بریده می‌شود.
Tile	تصویر با اندازه‌ی اصلی پشت‌سرهم (کاشی‌وار) تکرار می‌شود تا فرم پر شود (مقدار پیش‌فرض).
Center	تصویر با اندازه‌ی اصلی در مرکز فرم قرار می‌گیرد.
Stretch	تصویر کشیده می‌شود تا تمام فرم را پر کند؛ نسبت ابعاد حفظ نمی‌شود.
Zoom	تصویر با حفظ نسبت ابعاد بزرگ/کوچک می‌شود تا در فرم جا شود؛ ممکن است بخشی از فرم خالی بماند.

ایران توشه
توشه ای برای موفقیت

۴ صفحه ۱۰۵ فعالیت گروهی

ویژگی FormBorderStyle فرم را با مقادیر مختلف آزمایش کنید.

مقادیر این ویژگی را بررسی کرده، نتیجه را در جدول ۳ بنویسید.

جدول ۳ـ تأثیر مقادیر ویژگی FormBorderStyle روی فرم

مقدار	نتیجه
None	فرم بدون کادر و نوار عنوان است؛ اندازه‌اش با ماوس تغییر نمی‌کند.
FixedSingle	کادر تک‌خطی و اندازه‌ی ثابت؛ دکمه‌های کوچک/بزرگ‌کردن و بستن دارد.
Fixed3D	مانند FixedSingle با کادر سه‌بعدی (برجسته).
FixedDialog	کادر ثابت مخصوص پنجره‌های محاوره‌ای؛ بدون آیکن و دکمه‌های کوچک/بزرگ‌کردن.
Sizable	مقدار پیش‌فرض؛ کادر معمولی که اندازه‌ی آن با ماوس تغییر می‌کند.
FixedToolWindow	اندازه‌ی ثابت، نوار عنوان باریک و فقط دکمه‌ی بستن؛ در نوار وظیفه نمی‌آید.
SizableToolWindow	مانند FixedToolWindow ولی اندازه‌ی آن قابل تغییر است.

۵ صفحه ۱۰۵ **فعالیت گروهی**

با کمک هم‌گروهی خود جدول ۴ را کامل کنید.

جدول ۴ـ عملکرد برخی ویژگی‌های فرم

جدول ۴ـ عملکرد برخی ویژگی‌های فرم

ویژگی	عملکرد
Font	نوع، اندازه و سبک قلم متن فرم را تعیین می‌کند (کنترل‌های روی فرم آن را به ارث می‌برند).
Size (Width, Height)	عرض و ارتفاع فرم را بر حسب پیکسل مشخص می‌کند.
Locked	اگر true باشد فرم در محیط طراحی با ماوس جابه‌جا و تغییر اندازه نمی‌شود.
Text	متن نوار عنوان فرم را تعیین می‌کند.
Icon	آیکن گوشه‌ی نوار عنوان و نوار وظیفه را تعیین می‌کند.
ControlBox	نمایش یا عدم نمایش دکمه‌های کنترل (بستن، کوچک و بزرگ‌کردن) و آیکن نوار عنوان.

ایران توشه
توشه‌ای برای موفقیت

۶ صفحه ۱۰۵ فعالیت گروهی

ویژگی RightToLeftLayout فرم را true کنید. در ظاهر فرم چه تغییری ایجاد می‌شود؟

وقتی RightToLeft برابر Yes باشد و RightToLeftLayout برابر true شود، کل فرم قرینه می‌شود: دکمه‌های کنترل و آیکن نوار عنوان به سمت چپ می‌روند، عنوان در سمت راست می‌ماند و چیدمان اجزا (مثلاً نوار پیمایش) به‌صورت راست‌به‌چپ نمایش داده می‌شود. اگر RightToLeft برابر Yes نباشد، این ویژگی اثری ندارد.

۷ صفحه ۱۰۶ کارگاه ۲

با ماوس عرض (Width) برچسب را به اندازه‌ی عرض فرم تغییر دهید. عرض برچسب تغییر نمی‌کند. چرا؟
ویژگی RightToLeft برچسب را به Yes تغییر داده، نتیجه را بررسی کنید.

چون ویژگی AutoSize برچسب به‌طور پیش‌فرض true است و اندازه‌ی برچسب همیشه خودکار با اندازه‌ی متن آن تنظیم می‌شود؛ با false کردن آن می‌توان اندازه را دستی تغییر داد.

با RightToLeft = Yes جهت متن برچسب راست‌به‌چپ می‌شود و (وقتی برچسب عریض است) متن در سمت راست برچسب قرار می‌گیرد؛ علائم نگارشی مانند نقطه و گیومه نیز در جای درست فارسی نمایش داده می‌شوند.

۸ صفحه ۱۰۷ کارگاه ۲

ویژگی TextAlign را با مقادیر دیگر مقداردهی کرده، نتیجه را روی برچسب بررسی کنید.

این ویژگی ۹ مقدار دارد که محل قرارگرفتن متن را درون برچسب (یا دکمه) مشخص می‌کند:

مقادیر TextAlign

بالا	وسط	پایین
TopLeft	MiddleLeft	BottomLeft
TopCenter	MiddleCenter	BottomCenter
TopRight	MiddleRight	BottomRight

نکته: Top و Bottom و Middle جهت عمودی و Left و Center و Right جهت افقی را تعیین می‌کنند.

۹ صفحه ۱۰۷ کارگاه ۳

ویژگی Name کادر متن چه مقداری دارد؟

ویژگی Text کادر متن را بررسی کنید. مقدار پیش‌فرض این ویژگی چیست؟ ... آیا کادر متن فقط برای دریافت اطلاعات به کار می‌رود؟

آیا مقدار ویژگی‌های کادر متن جدید با کادر متن قبلی تفاوت دارد؟ کدام ویژگی کادر متن جدید با قبلی متفاوت است؟

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

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

- مقدار پیش‌فرض Name: textBox1 (نام کنترل + عدد).
- مقدار پیش‌فرض Text: رشته‌ی خالی (""). خیر؛ کادر متن برای نمایش نتیجه هم به کار می‌رود (با ReadOnly = true کاربر نمی‌تواند آن را ویرایش کند).
- بله، فقط Name متفاوت است (textBox2) و نیز Location (محل روی فرم)؛ بقیه‌ی ویژگی‌ها (اندازه، قلم، رنگ و ...) یکسان کپی می‌شوند.
- خیر، فقط یک نویسه؛ نوع داده‌ی آن char است.
- مقدار Inherit (از فرم ارث می‌برند)؛ یعنی در عمل جهت راست‌به‌چپ فرم را می‌گیرند.

۱۰ صفحه ۱۰۸ کارگاه ۳

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؟

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؟

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.

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

- بعد از نویسه‌ی هشتم، نویسه‌های بیشتر پذیرفته نمی‌شوند (چیزی نوشته نمی‌شود و معمولاً صدای هشدار شنیده می‌شود).
- شکل اشاره‌گر هنگام قرارگرفتن روی دکمه از فلش به دست (انگشت اشاره) تغییر می‌کند.
- دکمه کم‌رنگ (خاکستری) می‌شود و به کلیک پاسخ نمی‌دهد؛ کادر متن نیز خاکستری شده و نمی‌توان در آن نوشت؛ متن برچسب هم کم‌رنگ دیده می‌شود.

ایران توشه
توشه‌ای برای موفقیت

- کنترل در زمان اجرا دیده نمی‌شود (در محیط طراحی همچنان دیده می‌شود) و کاربر نمی‌تواند با آن کار کند.

۱۱ صفحه ۱۰۸ فعالیت گروهی

با کمک هم‌گروهی خود، جدول ویژگی‌های کنترل‌ها را تکمیل کنید.

آیا می‌توان یک ویژگی چند کنترل را هم‌زمان تغییر داد؟

جدول ویژگی‌های کنترل‌ها

نام ویژگی	عملکرد	کنترل
Name	نام کنترل (نام متغیر دسترسی به آن در کد)	همه کنترل‌ها
BackColor	رنگ زمینه	Label, TextBox, Button, Form
ForeColor	رنگ نوشته (قلم)	Label, TextBox, Button, Form
Text	متنی که روی کنترل نمایش داده یا در آن نگهداری می‌شود	Label, TextBox, Button, Form
Enabled	فعال یا غیرفعال بودن کنترل (پاسخ‌دادن به کاربر)	همه کنترل‌ها
Visible	نمایش یا پنهان‌بودن کنترل در زمان اجرا	همه کنترل‌ها
AutoSize	تنظیم خودکار اندازه‌ی کنترل با اندازه‌ی متن	Label, Form, Button
TextAlign	محل قرارگرفتن متن درون کنترل	Label, Button, TextBox (افقی)
Size	عرض و ارتفاع کنترل	همه کنترل‌ها
Location	محل گوشه‌ی بالا-چپ کنترل روی فرم (X, Y)	همه کنترل‌ها
Cursor	شکل اشاره‌گر ماوس روی کنترل	همه کنترل‌ها
Dock	چسباندن کنترل به یکی از لبه‌های فرم (Top, Bottom, Left, Right, Fill)	همه کنترل‌ها
RightToLeft	جهت نمایش متن و چیدمان راست‌به‌چپ	همه کنترل‌ها

بله؛ با نگه‌داشتن Ctrl یا Shift و کلیک روی چند کنترل (یا انتخاب با کادر ماوس)، پنجره‌ی Properties ویژگی‌های مشترک را نشان می‌دهد و تغییر هر مقدار روی همه‌ی کنترل‌های انتخاب‌شده اعمال می‌شود.

ایران توشه
توشه‌ای برای موفقیت

۱۲ صفحه ۱۰۹ فعالیت منزل

فرم مشخصات کاربر را ایجاد کنید (شکل ۸).

ـ ویژگی Name کنترل‌های کادر متن و دکمه را به صورت زیر قرار دهید: txtFirstName، txtLastName، txtUserName، txtPassword، txtRePassword، txtAddress، btnSave، btnDelete، btnSearch.

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

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

طراحی فرم (RightToLeft = Yes)

کنترل	Name	ویژگی‌های مهم
Label ×5	—	Text: نام، نام خانوادگی، نام کاربری، گذرواژه، تکرار گذرواژه، نشانی
TextBox	txtFirstName, txtLastName	—
TextBox	txtUserName	MaxLength = 10
TextBox	txtPassword, txtRePassword	* = PasswordChar
TextBox	txtAddress	Multiline = true (سپس ارتفاع را زیاد کنید)
Button	btnSave, btnDelete, btnSearch	Text: ثبت، حذف، جستجو

۱۳ صفحه ۱۰۹ فعالیت منزل

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.

1. فرم: BackColor = 242, 242, 242، FormBorderStyle = FixedSingle، MaximizeBox = false.
2. یک TextBox یا Label بزرگ به نام txtDisplay در بالا (TextAlign = Right، Font بزرگ، ReadOnly = true).
3. ۱۶ دکمه در جدولی ۴×۴ (ردیف‌ها: ÷ 9 8 7، × 6 5 4، − 3 2 1، + = . 0) به‌اندازه‌ی یکسان بچینید (Make Same Size ← Format و Align).
4. برای همه‌ی دکمه‌ها (با انتخاب هم‌زمان): FlatStyle = Flat، FlatAppearance.BorderSize = 0، رنگ زمینه‌ی سفید برای ارقام و خاکستری برای عملگرها، Cursor = Hand.

5. نام‌ها: btn0 ... btn9، btnAdd، btnSub، btnMul، btnDiv، btnEqual، btnDot.

۱۴ صفحه ۱۱۰ 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؟

هیچ اتفاقی نمی‌افتد؛ زیرا برای رویداد Click دکمه‌ی خروج هنوز متد رویداد (واکنش) نوشته نشده است و برنامه فقط به رویدادهایی پاسخ می‌دهد که برایشان کد نوشته شده باشد.

برای خروج باید در متد رویداد کلیک دکمه، فرم بسته شود: this.Close(); (یا Application.Exit();)؛ با بسته‌شدن فرم اصلی، برنامه پایان می‌یابد.

۱۵ صفحه ۱۱۲ کارگاه ۴

برنامه را با کلید F5 اجرا کنید. روی دکمه خروج کلیک کنید. چه اتفاقی می‌افتد؟

با کلیک روی btnExit متد btnExit_Click اجرا و دستور this.Close() فرم را می‌بندد؛ چون این فرم، فرم اصلی برنامه است، اجرای برنامه نیز پایان می‌یابد و IDE به حالت طراحی برمی‌گردد.

۱۶ صفحه ۱۱۳ کارگاه ۴

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

چرا برای انتساب ویژگی Text کادرهای متن به متغیر num1 و num2 از متد Parse استفاده شده است؟

این دستور محتوای ویژگی Text کادر متن txtNumber1 را رشته‌ی خالی می‌کند، یعنی متن کادر پاک می‌شود.

چون ویژگی Text از نوع رشته (string) است و متغیرهای num1 و num2 از نوع double هستند؛ مقدار رشته‌ای مستقیماً به عدد انتساب نمی‌شود و باید با double.Parse به عدد تبدیل شود تا بتوان محاسبه انجام داد.

۱۷ صفحه ۱۱۳ کارگاه ۴

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

```
private void btnSub_Click(object sender, EventArgs e)
{
    double num1 = double.Parse(txtNumber1.Text);
    double num2 = double.Parse(txtNumber2.Text);
    txtResult.Text = (num1 - num2).ToString();
}
private void btnMul_Click(object sender, EventArgs e)
{
    double num1 = double.Parse(txtNumber1.Text);
    double num2 = double.Parse(txtNumber2.Text);
    txtResult.Text = (num1 * num2).ToString();
}
private void btnDiv_Click(object sender, EventArgs e)
{
    double num1 = double.Parse(txtNumber1.Text);
    double num2 = double.Parse(txtNumber2.Text);
    if (num2 == 0)
        txtResult.Text = "Error";
    else
        txtResult.Text = (num1 / num2).ToString();
}
```

نام دکمه‌ها btnSub، btnMul و btnDiv است. بررسی صفر نبودن مقسوم‌علیه از نمایش بی‌نهایت جلوگیری می‌کند.

۱۸ صفحه ۱۱۴ کارگاه ۵

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

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

تصویر به فهرست منابع پروژه در پنجره‌ی Resources.resx اضافه می‌شود و به‌صورت یک تصویر کوچک با نام پرونده نمایش داده می‌شود؛ یک نسخه از آن در پوشه‌ی Resources مسیر پروژه کپی می‌شود.

در Solution Explorer زیر شاخه‌ی Resources پروژه، پرونده‌ی تصویر جدید (با همان نام) اضافه می‌شود و با افزودن هر تصویر، یک پرونده‌ی دیگر به این پوشه افزوده می‌شود.

ایران توشه
توشه ای برای موفقیت

۱۹ صفحه ۱۱۵ فعالیت کارگاهی

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؟

یک تصویر خالی جدید با قالب انتخابی (JPEG، GIF، BMP، PNG یا آیکن) ساخته و در ویرایشگر تصویر داخلی ویژوال استودیو باز می‌شود تا با ابزارهای نقاشی آن را بکشید و به منابع پروژه افزوده شود.

هنگامی به کار می‌رود که تصویر آماده نداریم و می‌خواهیم آیکن یا تصویر ساده‌ای را مستقیماً در IDE طراحی کنیم.

۲۰ صفحه ۱۱۶ کارگاه ۶

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.

تفاوت دو گزینه

Local resource	Project resource file
تصویر در پرونده‌ی resx. همان فرم ذخیره می‌شود (جاسازی در فرم).	تصویر در منابع پروژه (Resources.resx و پوشه‌ی Resources) نگهداری می‌شود.
فقط همان کنترل/فرم به آن دسترسی دارد.	در همه‌ی فرم‌ها و در کد با نام Properties.Resources. قابل استفاده است.
برای تصویر مخصوص یک فرم مناسب است.	برای تصویری که چند بار یا در چند فرم استفاده می‌شود مناسب است.

ایران توشه
توشه ای برای موفقیت

۲۱ صفحه ۱۱۶ کارگاه ۶

ویژگی SizeMode را انتخاب کرده، مقدار آن را روی StretchImage قرار دهید. عملکرد این ویژگی چیست؟ مقدار ویژگی Image کادر تصویر چه چیزی را نشان می‌دهد؟ مقدار این ویژگی را پاک کنید. چه اتفاقی می‌افتد؟

SizeMode نحوه‌ی نمایش تصویر در کادر را تعیین می‌کند. StretchImage تصویر را بزرگ یا کوچک می‌کند تا دقیقاً اندازه‌ی کادر شود و کامل دیده شود (نسبت ابعاد ممکن است به‌هم بخورد). مقادیر دیگر: Normal، CenterImage، AutoSize، Zoom.

Image تصویر نمایش‌داده‌شده در کادر (منبع تصویر) را نگه می‌دارد. با پاک‌کردن آن، تصویر از کادر حذف می‌شود و کادر خالی (بدون تصویر) می‌ماند.

۲۲ صفحه ۱۱۶ **کارگاه ۶**

دو دکمه با متن «تصویر ۱» و «تصویر ۲» به فرم اضافه کنید. کدی بنویسید که با کلیک روی هر دکمه تصویر متناظر آن در کادر تصویر نمایش داده شود.

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

```
// image names in Resources: pic1, pic2, pic3 ...
private void btnPic1_Click(object sender, EventArgs e)
{
    picSample.Image = Properties.Resources.pic1;
}
private void btnPic2_Click(object sender, EventArgs e)
{
    picSample.Image = Properties.Resources.pic2;
}
private void btnPic3_Click(object sender, EventArgs e)
{
    picSample.Image = Properties.Resources.pic3;   // same pattern for other buttons
}
```

picN نام پرونده‌ی تصویر در پوشه‌ی منابع است. برای هر تصویر دیگر یک دکمه و متد مشابه می‌نویسیم.

۲۳ صفحه ۱۱۷ **فعالیت منزل**

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.

- فرم مشخصات کاربر: PictureBox با Name = picUser، SizeMode = StretchImage، BorderStyle = FixedSingle و Button با Name = btnLoadPicture و بارگذاری تصویر = Text.
- فرم Login: کنترل جدید (کادر متن، یا طبق عبارت کتاب، کادر تصویر برای آرم برنامه) را با نام مناسب (txtXxx یا picLogo) اضافه و تنظیم کنید.

کد دکمه‌ی btnLoadPicture در کارگاه ۸ (صفحه‌ی ۱۲۱) نوشته می‌شود.

متن کتاب در مورد فرم Login کامل خوانا نیست؛ این جواب ممکن است دقیق نباشد.

ایران توشه
توشه ای برای موفقیت

۲۴ صفحه ۱۱۷ کارگاه ۷

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

خیر. با افزایش Width و Height، گوشه‌ی بالا-چپ کادر ثابت می‌ماند و تصویر فقط به سمت راست و پایین بزرگ می‌شود، بنابراین از وسط فرم خارج می‌گردد.

۲۵ صفحه ۱۱۸ کارگاه ۷

اگر Top = 0 و Left = 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 ... آیا در پنجره ویژگی‌ها به Left و Top دسترسی دارید؟

متد رویداد کلیک دکمه کوچک‌نمایی را بنویسید.

در گوشه‌ی بالا-چپ ناحیه‌ی کاربر فرم (مبدأ مختصات). با افزایش Left کنترل به راست و با افزایش Top به پایین می‌رود. در پنجره‌ی Properties این دو ویژگی دیده نمی‌شوند و به‌جای آن‌ها Location (شامل X و Y) را داریم؛ Left و Top فقط در کد قابل استفاده‌اند.

```
private void btnZoomOut_Click(object sender, EventArgs e)
{
    picSample.Width -= 4;
    picSample.Height -= 3;
    picSample.Left = (this.Width - picSample.Width) / 2;
    picSample.Top = (this.Height - picSample.Height) / 2;
}
```

ایران توشه
توشه ای برای موفقیت

۲۶ صفحه ۱۱۸ کارگاه ۷

تصویر دلخواه 1.jpg را در درایو C قرار دهید ... برنامه را اجرا کنید. با کلیک روی دکمه btnNext چه مشاهده می‌کنید؟

تصویر c:\\1.jpg در کادر تصویر بارگذاری و نمایش داده می‌شود (اگر مسیر یا نام پرونده غلط باشد، تصویر بارگذاری نمی‌شود و علامت خطا (ضربدر قرمز) دیده می‌شود).

۲۷ صفحه ۱۱۹ کارگاه ۷

عملکرد دستور if در متد btnNext_Click چیست؟

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

وقتی i به ۶ می‌رسد (بعد از تصویر پنجم) آن را دوباره ۱ می‌کند تا نمایش به تصویر اول برگردد و چرخه‌ی تصاویر ۱ تا ۵ ادامه یابد؛ در غیر این صورت نام 6.JPG وجود ندارد.

نشانی "1.JPG" نسبی است و نسبت به پوشه‌ی پرونده‌ی اجرایی (همان bin\\Debug) جست‌وجو می‌شود؛ اگر تصاویر آنجا نباشند، پیدا و نمایش داده نمی‌شوند.

۲۸ صفحه ۱۱۹ کارگاه ۷

برای نمایش تصویر قبلی متد رویداد کلیک btnPrev را بنویسید.

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

```
int i = 1;   // number of the shown picture (class field)

private void ShowPicture()
{
    picSample.ImageLocation = i.ToString() + ".JPG";
    btnPrev.Enabled = (i > 1);
    btnNext.Enabled = (i < 5);
}
private void btnNext_Click(object sender, EventArgs e)
{
    if (i < 5) i++;
    ShowPicture();
}
private void btnPrev_Click(object sender, EventArgs e)
{
    if (i > 1) i--;
    ShowPicture();
}
// in Form1_Load: ShowPicture();
```

Form1_Load را برای نمایش اولیه‌ی تصویر ۱ و غیرفعال‌شدن btnPrev از رویداد Load فرم بسازید.

۲۹ صفحه ۱۱۹ فعالیت کارگاهی

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(); و picSample.Hide();

- picSample.Hide();: کادر تصویر را پنهان می‌کند (معادل Visible = false).

ایران توشه
توشه ای برای موفقیت

- picSample.Show();: کادر تصویر پنهان‌شده را دوباره نمایش می‌دهد (معادل Visible = true).

۳۰ صفحه ۱۱۹ فعالیت منزل

شکل ۲۰ فرم طراحی یک بازی ساده را نشان می‌دهد. ... این فرم را طراحی کرده، متد رویداد کلیک دکمه‌ها را برای حرکت آدمک بنویسید.

طراحی: یک PictureBox به نام picMan (آدمک، SizeMode = StretchImage)، چند PictureBox برای کاشی‌ها و چهار دکمه‌ی btnUp، btnDown، btnLeft، btnRight. اندازه‌ی یک کاشی را ۴۰ پیکسل فرض می‌کنیم.

```
const int step = 40;   // one tile

private void btnUp_Click(object sender, EventArgs e)
{
    picMan.Top -= step;
}
private void btnDown_Click(object sender, EventArgs e)
{
    picMan.Top += step;
}
private void btnLeft_Click(object sender, EventArgs e)
{
    picMan.Left -= step;
}
private void btnRight_Click(object sender, EventArgs e)
{
    picMan.Left += step;
}
```

ایران توشه
توشه ای برای موفقیت

۳۱ صفحه ۱۲۱ کارگاه ۸

برنامه را اجرا کنید و یک پرونده تصویری انتخاب کنید. آیا می‌توان پرونده‌های غیرتصویری را انتخاب کرد؟

بله؛ چون هنوز Filter تنظیم نشده و همه‌ی پرونده‌ها نمایش داده می‌شوند. اگر پرونده‌ی غیرتصویری انتخاب شود، کادر تصویر نمی‌تواند آن را بارگذاری کند و به‌جای تصویر، علامت خطا (ضربدر قرمز) نشان می‌دهد.

۳۲ صفحه ۱۲۲ کارگاه ۸

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

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

در کد مرحله‌ی ۳ و ۴، FileName رشته‌ی خالی است و ImageLocation خالی می‌شود؛ در نتیجه کادر تصویر خالی می‌ماند (یا تصویر قبلی پاک می‌شود).

در کد مرحله‌ی ۵ با Cancel یا بستن کادر، ()ShowDialog مقدار DialogResult.Cancel برمی‌گرداند، شرط if برقرار نیست و هیچ تغییری در کادر تصویر ایجاد نمی‌شود (تصویر قبلی می‌ماند).

۳۳ صفحه ۱۲۲ کارگاه ۹

یک پروژه ویندوزی به نام PictureCity ایجاد کنید. فرم مناسب را طراحی کنید (شکل ۲۵).
ویژگی کادر متن را برای نوشتن متن به زبان فارسی تغییر دهید.
کد متد رویداد کلیک دکمه «انتخاب عکس» را برای تغییر عکس به انتخاب کاربر بنویسید.

طراحی فرم

کنترل	Name	ویژگی
TextBox	textBox1	Multiline = true، RightToLeft = Yes (نوشتن فارسی)
PictureBox	pictureBox1	SizeMode = StretchImage
Button	btnPicture, btnColor, btnFont, btnExit	Text: انتخاب عکس، انتخاب رنگ، انتخاب قلم، خروج
Component	openFileDialog1, colorDialog1, fontDialog1	OpenFileDialog، ColorDialog، FontDialog

```
private void btnPicture_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "Image Files(*.BMP;*.JPG;*.GIF)|*.BMP;*.JPG;*.GIF";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
        pictureBox1.ImageLocation = openFileDialog1.FileName;
}
private void btnExit_Click(object sender, EventArgs e)
{
    this.Close();
}
```

ایران توشه
توشه‌ای برای موفقیت

۳۴ صفحه ۱۲۳ کارگاه ۹

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؟

```
openFileDialog1.Title = "انتخاب عکس";
```

عنوان نوار عنوان کادر محاوره‌ای به‌جای عنوان پیش‌فرض Open به «انتخاب عکس» تغییر می‌کند.

این فایل توسط سایت ایران‌توشه تهیه شده است.

irantooshe.ir

۳۵ صفحه ۱۲۴ فعالیت گروهی

جدول ۵ ـ پارامترهای متد Show

با استفاده از راهنمای مایکروسافت و به کمک هم‌گروهی خود اعضای نوع داده‌های شمارشی MessageBoxIcon، MessageBoxButtons، MessageBoxDefaultButton و DialogResult را تعیین کنید.

جدول ۵ـ پارامترهای متد Show

پارامتر	نوع	شرح
پیام (text)	string	متنی که در کادر نمایش داده می‌شود؛ این پارامتر اجباری است.
عنوان (caption)	string	متنی که در نوار عنوان کادر نمایش داده می‌شود.
دکمه‌ها (buttons)	MessageBoxButtons (شمارشی)	دکمه‌هایی را مشخص می‌کند که باید در کادر پیام نمایش داده شوند.
نماد (icon)	MessageBoxIcon (شمارشی)	نماد (آیکن) کادر پیام را تعیین می‌کند.
دکمه‌ی پیش‌فرض (defaultButton)	MessageBoxDefaultButton (شمارشی)	دکمه‌ی پیش‌فرض کادر پیام را تعیین می‌کند.

اعضای انواع شمارشی

نوع	اعضا
MessageBoxButtons	OK, OKCancel, AbortRetryIgnore, YesNoCancel, YesNo, RetryCancel
MessageBoxIcon	None، Error (= Hand = Stop)، Question, Warning (= Exclamation) Information (= Asterisk)
MessageBoxDefaultButton	Button1, Button2, Button3
DialogResult	None, OK, Cancel, Abort, Retry, Ignore, Yes, No

۳۶ صفحه ۱۲۴ کارگاه ۱۰

برنامه را اجرا کنید. نامی بیش از ۱۰ نویسه وارد کنید و روی دکمه ثبت کلیک کنید. چه اتفاقی می‌افتد؟

کادر پیامی با متن «تعداد نویسه‌های نام بیشتر از حد مجاز است»، عنوان «پیام خطا» و دو دکمه‌ی Retry و Cancel (بدون نماد) نمایش داده می‌شود و تا انتخاب کاربر، برنامه منتظر می‌ماند.

۳۷ صفحه ۱۲۵ کارگاه ۱۰

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 می‌افتد؟

```
private void btnSave_Click(object sender, EventArgs e)
{
    if (txtFirstName.TextLength > 10)
        MessageBox.Show("پیام خطا", "تعداد نویسه‌های نام بیشتر از حد مجاز است",
                        MessageBoxButtons.RetryCancel);
    if (txtLastName.TextLength > 10)
        MessageBox.Show("پیام خطا", "تعداد نویسه‌های نام خانوادگی بیشتر از حد مجاز است",
                        MessageBoxButtons.RetryCancel);
}
```

با حذف عنوان، آرگومان‌های متد با هیچ‌یک از شکل‌های Show سازگار نیست (MessageBoxButtons به‌جای string در جای دوم قرار می‌گیرد) و برنامه خطای کامپایل می‌دهد. اگر دکمه‌ها را هم حذف کنیم و فقط پیام بماند، کادر با عنوان خالی و دکمه‌ی OK نمایش داده می‌شود.

۳۸ صفحه ۱۲۵ کارگاه ۱۰

کد را ویرایش کنید تا در صورت انتخاب دکمه Retry کادر متن نام و نام‌خانوادگی خالی شود.

دستورات مرحله ۴ را به صورت زیر تغییر دهید ... برنامه را اجرا کنید. نتیجه اجرای برنامه چه تفاوتی با اجرای قبل دارد؟

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

```
private void btnSave_Click(object sender, EventArgs e)
{
    if (txtFirstName.TextLength > 10)
        if (MessageBox.Show("پیام خطا" ,"تعداد نویسه‌های نام بیشتر از حد مجاز است",
            MessageBoxButtons.RetryCancel, MessageBoxIcon.Warning,
            MessageBoxDefaultButton.Button1) == DialogResult.Retry)
            txtFirstName.Clear();

    if (txtLastName.TextLength > 10)
        if (MessageBox.Show("پیام" ,"تعداد نویسه‌های نام خانوادگی بیشتر از حد مجاز است",
خطا",
            MessageBoxButtons.RetryCancel, MessageBoxIcon.Warning,
            MessageBoxDefaultButton.Button1) == DialogResult.Retry)
            txtLastName.Clear();
}
```

تفاوت: در کادر پیام، نماد هشدار (مثلث زرد با علامت تعجب) نمایش داده می‌شود و دکمه‌ی Retry (Button1) به‌عنوان دکمه‌ی پیش‌فرض انتخاب شده است (با Enter فعال می‌شود).

۳۹ صفحه ۱۲۵ **فعالیت کارگاهی**

با کمک هنرآموز خود تغییراتی در دستورات کارگاه ۱۰ ایجاد کنید که دو دکمه Ok، Cancel در کادر پیام نمایش داده شود. دکمه پیش‌فرض را دکمه Cancel تعیین کنید و تصویر نماد کادر محاوره‌ای را تغییر دهید.

```
MessageBox.Show("پیام خطا" ,"تعداد نویسه‌های نام بیشتر از حد مجاز است",
    MessageBoxButtons.OKCancel, MessageBoxIcon.Error,
    MessageBoxDefaultButton.Button2);
```

Button2 دومین دکمه یعنی Cancel است؛ نماد خطا (Error، دایره‌ی قرمز با ضربدر) نمایش داده می‌شود. نمادهای Question، Information و Warning هم قابل استفاده‌اند.

ایران توشه
توشه ای برای موفقیت

۴۰ صفحه ۱۲۵ فعالیت منزل

برنامه‌ای بنویسید که در «فرم ورود کاربر» اگر نام کاربری و گذرواژه معتبر بود پیام خوش‌آمدگویی و در صورت معتبر نبودن پیام خطا را نمایش دهد.

طراحی: txtUserName، txtPassword (PasswordChar = *)، btnLogin، btnExit و برچسب‌های راهنما.
نام کاربری و گذرواژه‌ی معتبر را ثابت (مثلاً admin و 1234) در نظر می‌گیریم.

```
private void btnLogin_Click(object sender, EventArgs e)
{
    if (txtUserName.Text == "admin" && txtPassword.Text == "1234")
        MessageBox.Show("ورود" ,"خوش آمدید", MessageBoxButtons.OK,
MessageBoxIcon.Information);
    else
        MessageBox.Show("خطا" ,"نام کاربری یا گذرواژه نادرست است", MessageBoxButtons.OK,
MessageBoxIcon.Error);
}
private void btnExit_Click(object sender, EventArgs e)
{
    this.Close();
}
```

۴۱ صفحه ۱۲۹ کارگاه ۱

به نظر شما چگونه می‌توان حرکت شخصیت‌های این بازی (شکل ۲۸ و ۲۹) را طراحی کرد؟ اگر قرار باشد شما این بازی را در VS بنویسید، برای حرکت‌های سریع و مداوم پک‌من از چه چیزی استفاده می‌کنید؟

شخصیت‌ها را با کنترل PictureBox نمایش می‌دهیم و برای حرکت مداوم، مکان آنها (Left و Top) را در فواصل زمانی بسیار کوتاه (مثلاً هر ۵۰ میلی‌ثانیه) اندکی تغییر می‌دهیم. این تکرار زمان‌بندی‌شده با کنترل **زمان‌سنج (Timer)** انجام می‌شود: ویژگی Interval فاصله‌ی زمانی را تعیین می‌کند و دستور حرکت را در رویداد Tick می‌نویسیم.

ایران توشه
توشه ای برای موفقیت

۴۲ صفحه ۱۳۰ کارگاه ۱

آیا زمان‌سنج روی فرم دیده می‌شود؟

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

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

۵ ـ برای توقف خودرو متد رویداد کلیک دکمه‌ی btnStop را بنویسید.

۶ ـ امکان کم و زیاد کردن سرعت خودرو را به برنامه اضافه کنید. چگونه می‌توان سرعت خودرو را کم یا زیاد کرد؟

خیر؛ زمان‌سنج یک component است و روی سطح فرم دیده نمی‌شود؛ در پایین پنجره‌ی طراحی (سینی اجزا یا Component Tray) قرار می‌گیرد.

متد timer1_Tick هر بار که فاصله‌ی زمانی Interval می‌گذرد اجرا می‌شود و با دستور picCar.Left += 5 کادر تصویر خودرو را ۵ پیکسل به راست می‌برد. تا وقتی زمان‌سنج فعال نشود (Enabled = false که مقدار پیش‌فرض است) رویداد Tick رخ نمی‌دهد و خودرو حرکت نمی‌کند. پس از نوشتن متد کلیک btnStart و فعال شدن زمان‌سنج، خودرو به‌طور پیوسته حرکت می‌کند.

```
private void timer1_Tick(object sender, EventArgs e)
{
    picCar.Left += 5;
}
private void btnStart_Click(object sender, EventArgs e)
{
    timer1.Enabled = true;
}
private void btnStop_Click(object sender, EventArgs e)
{
    timer1.Enabled = false;
}
```

تغییر سرعت

سرعت را می‌توان با کم کردن Interval (تکرار سریع‌تر) یا زیاد کردن مقدار جابه‌جایی در هر Tick افزایش داد. برای این کار دو دکمه‌ی btnFast و btnSlow اضافه می‌کنیم:

```
int step = 5;                    // در کلاس Form
private void timer1_Tick(object sender, EventArgs e)
{
    picCar.Left += step;
}
private void btnFast_Click(object sender, EventArgs e)
{
    step += 2;
}
private void btnSlow_Click(object sender, EventArgs e)
{
    if (step > 1) step -= 2;
}
```

ایران توشه
توشه ای برای موفقیت

۴۳ صفحه ۱۳۰ فعالیت منزل

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

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

```
int step = 5;
private void timer1_Tick(object sender, EventArgs e)
{
    picCar.Left += step;
    // رسیدن به لبه‌ی راست یا چپ فرم
    if (picCar.Left + picCar.Width >= this.ClientSize.Width || picCar.Left <= 0)
        timer1.Enabled = false;
}
private void btnForward_Click(object sender, EventArgs e)
{
    step = 5;                // حرکت به جلو (راست)
    timer1.Enabled = true;
}
private void btnBackward_Click(object sender, EventArgs e)
{
    step = -5;               // حرکت به عقب (چپ)
    timer1.Enabled = true;
}
```

با منفی کردن مقدار step خودرو به سمت چپ (عقب) حرکت می‌کند.

۴۴ صفحه ۱۳۱ کارگاه ۲

۳ ـ ویژگی Interval زمان‌سنج را باید با چه عددی مقداردهی کنید؟

۴ ـ کد ثانیه‌شمار را در رویداد Tick زمان‌سنج بنویسید. متغیرهای second و minute را برای ذخیره ثانیه و دقیقه تعریف کنید.

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

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

ثانیه‌شمار باید هر یک ثانیه یک بار عمل کند؛ چون Interval بر حسب میلی‌ثانیه است، مقدار آن 1000 خواهد بود.

```
public partial class Form1 : Form
{
    int second = 0, minute = 0;     // تعریف متغیرها در کلاس 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();
    }
    private void btnStart_Click(object sender, EventArgs e)
    {
        minute = 0; second = 0;
        lblMinute.Text = "0"; lblSecond.Text = "0";
        timer1.Enabled = true;
    }
    private void btnStop_Click(object sender, EventArgs e)
    {
        timer1.Enabled = false;
    }
}
```

۴۵ صفحه ۱۳۱ **فعالیت منزل**

ساعت سیستم را در برنامه‌ی خود نمایش دهید.

یک برچسب به نام lblTime و یک زمان‌سنج با Interval = 1000 و Enabled = true روی فرم قرار می‌دهیم:

```
private void timer1_Tick(object sender, EventArgs e)
{
    lblTime.Text = DateTime.Now.ToLongTimeString();
}
```

۴۶ صفحه ۱۳۳ **فعالیت گروهی**

جدول ۶ ـ برخی ویژگی‌های مشترک کنترل‌های دکمه‌ی رادیویی و کادر علامت (ویژگی‌های Checked، Enabled، CheckAlign و Text).

این جدول در خود کتاب کامل است و پاسخ‌دهی لازم ندارد؛ برای یادآوری:

ایران توشه
توشه‌ای برای موفقیت

جدول ۶

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

۴۷ صفحه ۱۳۴ کارگاه ۳

برنامه را اجرا کنید. عملکرد متد رویداد chkPassword_CheckedChanged چیست؟

۵ ـ متد رویداد کلیک دکمه‌ی «ورود» را بنویسید. برنامه را اجرا کنید. با انتخاب انواع کاربر دکمه‌ی ورود را کلیک کنید.

این متد هر بار که وضعیت انتخاب کادر علامت chkPassword تغییر کند اجرا می‌شود: اگر علامت زده شود (Checked == true) با PasswordChar = '\0' نویسه‌های گذرواژه نمایش داده می‌شوند و اگر علامت برداشته شود با PasswordChar = '*' دوباره با ستاره پنهان می‌شوند.

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

با انتخاب هر دکمه‌ی رادیویی و کلیک روی «ورود»، پیام خوش‌آمدگویی مربوط به همان نوع کاربر نمایش داده می‌شود.

ایران توشه
توشه ای برای موفقیت

۴۸ صفحه ۱۳۵ فعالیت گروهی

به نظر شما چگونه می‌توانید هم‌زمان دو دکمه‌ی رادیویی را در این پنجره انتخاب کنید؟

دکمه‌های رادیویی که در یک ظرف (فرم، GroupBox یا Panel) قرار دارند یک گروه را تشکیل می‌دهند و در هر گروه فقط یک گزینه انتخاب می‌شود. برای انتخاب هم‌زمان دو دکمه باید آنها را در دو گروه جداگانه (مثلاً یک گروه برای نحوه‌ی نمایش پوشه‌ها و گروه دیگر برای روش انتخاب پوشه) قرار داد؛ آنگاه از هر گروه یک گزینه قابل انتخاب است.

۴۹ صفحه ۱۳۶ کارگاه ۴

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

۵ ـ پنل را روی فرم جابه‌جا کنید. دکمه‌های رادیویی روی پنل چه تغییری می‌کنند؟

با RightToLeft = Yes ویژگی RightToLeft همه‌ی کادرهای علامت داخل کادر گروهی نیز Yes می‌شود؛ متن آنها راست‌چین و علامت‌ها در سمت راست قرار می‌گیرند.

با تغییر قلم (Font) کادر گروهی، قلم دکمه‌های رادیویی داخل آن هم همان‌طور تغییر می‌کند؛ نتیجه: کنترل‌های داخل ظرف «فرزند» آن هستند و برخی ویژگی‌های عمومی (قلم، رنگ، RightToLeft، Enabled) از ظرف (والد) به فرزندان به ارث می‌رسد.

قلم پنل نیز مانند کادر گروهی به دکمه‌های رادیویی داخل آن منتقل می‌شود. کنترل Panel ویژگی Text ندارد؛ برای نوشتن عنوان باید یک برچسب (Label) روی آن قرار داد (یا از GroupBox که دارای Text است استفاده کرد).

با جابه‌جا کردن پنل، همه‌ی دکمه‌های رادیویی داخل آن همراه با پنل جابه‌جا می‌شوند؛ زیرا موقعیت فرزندان نسبت به والد ثابت می‌ماند.

ایران توشه
توشه‌ای برای موفقیت

۵۰ صفحه ۱۳۶ کارگاه ۴

۶ ـ 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.

هر کادر علامت با برچسب هم‌شماره‌ی خود مرتبط است. با علامت زدن کادر، برچسب پنهان می‌شود و قسمت زیر آن دیده می‌شود:

```
private void checkBox1_CheckedChanged(object sender, EventArgs e)
{
    label1.Visible = false;
}
// برای کادرهای ۲ تا ۹ نیز مشابه:
private void checkBox2_CheckedChanged(object sender, EventArgs e)
{
    label2.Visible = false;
}
```

برای اینکه با برداشتن علامت، برچسب دوباره ظاهر شود، می‌توان نوشت: label1.Visible = !checkBox1.Checked;.

۵۱ صفحه ۱۳۷ کارگاه ۴

جدول ۷ را به کمک راهنمای مایکروسافت و با توجه به شکل‌های مختلف این متد کامل کنید.

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

جدول ۷ـ شکل‌های مختلف متد Next

شکل	شرح
Next()	عدد صحیح تصادفی نامنفی (از ۰ تا حداکثر مقدار int) تولید می‌کند.
Next(int max)	عدد صحیح تصادفی از ۰ تا max−1 (کوچک‌تر از max) تولید می‌کند.
Next(int min, int max)	عدد صحیح تصادفی از min تا max−1 (min شامل، max شامل نیست) تولید می‌کند.

برای تولید عدد ۱ تا ۸ باید rnd.Next(1, 9) نوشته شود. متغیرهای n و rnd در کلاس Form تعریف می‌شوند تا در متد کلیک دکمه‌ی «پاسخ صحیح» هم قابل استفاده باشند:

ایران توشه
توشه‌ای برای موفقیت

```
int n;
Random rnd = new Random();
private void btnChangePicture_Click(object sender, EventArgs e)
{
    n = rnd.Next(1, 9);              // عدد تصادفی ۱ تا ۸
    string s = n.ToString() + ".jpg";
    pictureBox1.ImageLocation = s;   // تصویر ۱.jpg تا ۸.jpg در پوشه‌ی Debug
}
```

۵۲ صفحه ۱۳۸ کارگاه ۴

۸ ـ متد رویداد کلیک دکمه‌ی «پاسخ صحیح» را بنویسید. با استفاده از عدد تصادفی تولیدشده در متد کلیک دکمه‌ی «تغییر تصویر»، متد رویداد کلیک دکمه‌ی «پاسخ صحیح» را کامل کنید.

برای هر تصویر پیام مربوط به مکان همان تصویر نمایش داده می‌شود (حالت‌های ۱ و ۲ در کتاب آمده است؛ سایر مکان‌ها را مطابق ۸ تصویر انتخابی خودتان بنویسید):

```
private void btnCorrect_Click(object sender, EventArgs e)
{
    switch (n)
    {
        case 1:
            MessageBox.Show("پاسخ صحیح: سی و سه پل در استان اصفهان", "پاسخ صحیح",
MessageBoxButtons.OKCancel);
            break;
        case 2:
            MessageBox.Show("پاسخ صحیح: دروازه قرآن در شیراز", "پاسخ صحیح",
MessageBoxButtons.OKCancel);
            break;
        case 3:
            MessageBox.Show("پاسخ صحیح: برج میلاد در تهران", "پاسخ صحیح",
MessageBoxButtons.OKCancel);
            break;
        case 4:
            MessageBox.Show("پاسخ صحیح: تخت جمشید در استان فارس", "پاسخ صحیح",
MessageBoxButtons.OKCancel);
            break;
        // ... حالت‌های ۵ تا ۸ برای بقیه‌ی تصویرها
    }
}
```

این جواب ممکن است دقیق نباشد.

ایران توشه
توشه‌ای برای موفقیت

۵۳ صفحه ۱۳۸ فعالیت گروهی

ـ برنامه‌ی Tour را تغییر دهید تا به‌جای انتخاب کادر علامت با کلیک روی هر برچسب بخشی از تصویر ظاهر شود.

کادرهای علامت را حذف کرده، برای رویداد Click هر برچسب متدی می‌نویسیم که همان برچسب را پنهان کند:

```
private void label1_Click(object sender, EventArgs e)
{
    label1.Visible = false;
}
// ... مشابه برای label2 تا label9

// یا یک متد مشترک برای همه‌ی برچسب‌ها:
private void label_Click(object sender, EventArgs e)
{
    ((Label)sender).Visible = false;
}
```

۵۴ صفحه ۱۳۸ فعالیت گروهی

با کمک هم‌گروهی خود و هنرآموز جدول زیر را کامل کنید. (نام کنترل، کاربرد، ویژگی‌های مهم، رویدادهای مهم)

کنترل‌های انتخاب و گروه‌بندی

نام کنترل	کاربرد	ویژگی‌های مهم	رویدادهای مهم
دکمه‌ی رادیویی (RadioButton)	انتخاب فقط یک گزینه از میان چند گزینه	CheckAlign، Checked، Text، Enabled	CheckedChanged، Click
کادر علامت (CheckBox)	انتخاب یک یا چند گزینه (علامت‌گذاری)	CheckAlign، Checked، Text، Enabled	CheckedChanged، Click
کادر گروهی (GroupBox)	گروه‌بندی و دسته‌بندی کنترل‌ها با عنوان	Text، Enabled، Font، RightToLeft	Enter، Leave، Click
پنل (Panel)	گروه‌بندی کنترل‌ها بدون عنوان (با امکان نوار پیمایش)	AutoScroll، BorderStyle، BackColor، Enabled	Paint، Click، Enter

ایران توشه
توشه ای برای موفقیت

۵۵ صفحه ۱۴۰ کارگاه ۵

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

۵ ـ یک سربرگ دیگر به کنترل Tab اضافه کنید ... گزینه‌ی Add را انتخاب کنید. چه اتفاقی می‌افتد؟ نتیجه را در فرم مشاهده کنید.

۸ ـ کد زیر را در متد رویداد کلیک دکمه‌ی «ادامه» برای رفتن به سربرگ اطلاعات تحصیلی بنویسید:

tabControl1.SelectedTab = tabPage2; کاربرد ویژگی SelectedTab چیست؟

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

با RightToLeftLayout = true (به همراه RightToLeft = Yes) چیدمان سربرگ‌ها از راست به چپ می‌شود: سربرگ اول در سمت راست و نوار سربرگ‌ها راست‌چین قرار می‌گیرد.

با کلیک روی Add یک سربرگ جدید (مثلاً tabPage3) در فهرست سربرگ‌ها ساخته می‌شود و در فرم، در انتهای نوار سربرگ‌ها دیده می‌شود.

ویژگی SelectedTab سربرگی را که در حال حاضر انتخاب و نمایش داده شده مشخص می‌کند؛ با مقداردهی آن در برنامه می‌توان سربرگ نمایش‌داده‌شده را تغییر داد.

```
private void btnContinue_Click(object sender, EventArgs e)
{
    tabControl1.SelectedTab = tabPage2;   // رفتن به سربرگ اطلاعات تحصیلی
}
private void btnExit1_Click(object sender, EventArgs e)   // دکمه‌ی خروج سربرگ اول
{
    this.Close();
}
private void btnExit2_Click(object sender, EventArgs e)   // دکمه‌ی خروج سربرگ دوم
{
    this.Close();
}
```

ایران توشه
توشه ای برای موفقیت

۵۶ صفحه ۱۴۲ کارگاه ۶

برنامه را با فشردن کلید میانبر F5 اجرا کنید. پس از اجرای برنامه کدام فرم نمایش داده می‌شود؟

نام فرم دوم (MainForm) را به‌جای LoginForm در ورودی متد Run بنویسید. برنامه را اجرا کنید. کدام فرم به‌عنوان فرم شروع برنامه نمایش داده می‌شود؟

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

پس از اضافه شدن MainForm هنوز LoginForm نمایش داده می‌شود؛ زیرا فرم شروع برنامه همان فرمی است که در دستور ;(Application.Run(new LoginForm در پرونده‌ی Program.cs آمده است، نه هر فرمی که در پروژه وجود دارد.

```
// Program.cs
Application.Run(new MainForm());
```

پس از این تغییر MainForm به‌عنوان فرم شروع نمایش داده می‌شود.

مقدار اولیه‌ی فیلد succeeded (از نوع bool) برابر false و مقدار اولیه‌ی فیلد userName (از نوع string) برابر null است.

۵۷ صفحه ۱۴۴ کارگاه ۶

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

۷ ـ متد رویداد Load فرم صفحه‌ی شخصی کاربر را بنویسید.

ایران توشه
توشه ای برای موفقیت

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

```
// در LoginForm
public bool succeeded;
public string userName;
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);
}

// در MainForm
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();
}
```

دستور LoginForm frm = new LoginForm(); یک شیء (نمونه) از کلاس LoginForm می‌سازد و آدرس آن را در متغیر frm نگه می‌دارد تا بتوان فرم ورود را نمایش داد و به فیلدهای عمومی آن (succeeded و userName) دسترسی داشت.

ایران توشه
توشه ای برای موفقیت

۵۸ صفحه ۱۴۵ **فعالیت کارگاهی**

ـ متن lblWelcome را به‌گونه‌ای تغییر دهید که ظاهر UserForm مشابه شکل ۴۰ شود.

ـ برای نمایش فرم می‌توان از متدهای Show یا ShowDialog استفاده کرد. به کمک هنرآموز خود تفاوت این دو متد را بررسی کنید.

ـ آیا می‌توان در این برنامه به‌جای متد ShowDialog از متد Show استفاده کرد؟ چرا؟

با تغییر ویژگی‌های Font، ForeColor و TextAlign برچسب و قرار دادن نام کاربر در متن آن، ظاهر فرم مانند شکل ۴۰ می‌شود (مثلاً: «کاربر گرامی علیرضا خوش آمدید» با قلم درشت و وسط‌چین).

تفاوت Show و ShowDialog

متد	تفاوت
Show()	فرم را به‌صورت غیرمحاوره‌ای (modeless) نمایش می‌دهد؛ اجرای برنامه بلافاصله به دستور بعدی می‌رود و کاربر می‌تواند هم‌زمان با فرم‌های دیگر کار کند.
ShowDialog()	فرم را به‌صورت محاوره‌ای (modal) نمایش می‌دهد؛ تا بستن آن فرم، اجرای دستورات بعدی در فرم فراخواننده متوقف می‌شود و کاربر نمی‌تواند به فرم‌های دیگر برگردد.

خیر؛ با Show() برنامه پس از نمایش فرم ورود بلافاصله دستور if (frm.succeeded) را اجرا می‌کند، در حالی که کاربر هنوز نام کاربری و گذرواژه را وارد نکرده است و succeeded برابر false می‌ماند؛ در نتیجه MainForm بسته می‌شود. برای صبر کردن تا پایان ورود اطلاعات باید ShowDialog() به کار رود.

۵۹ صفحه ۱۴۵ **فعالیت منزل**

برنامه‌ی 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 امکان‌پذیر نباشد.

در LoginForm یک شمارنده‌ی خطا و یک زمان‌سنج (timerLock با Interval = 30000 و Enabled = false) اضافه می‌کنیم:

ایران توشه
توشه ای برای موفقیت

```
int wrong = 0;     // تعداد ورودهای اشتباه (فیلد کلاس)
private void btnLogin_Click(object sender, EventArgs e)
{
    if (txtUserName.Text == "علیرضا" && txtPassword.Text == "123")
    {
        succeeded = true;
        userName = txtUserName.Text;
        this.Close();
    }
    else
    {
        wrong++;
        MessageBox.Show("خطا" ,"نام کاربری یا گذرواژه معتبر نیست", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        if (wrong == 3)
        {
            MessageBox.Show("ورود به مدت ۳۰ ثانیه غیرفعال است");
            txtUserName.Enabled = txtPassword.Enabled = btnLogin.Enabled = false;
            timerLock.Enabled = true;
        }
    }
}
private void timerLock_Tick(object sender, EventArgs e)
{
    timerLock.Enabled = false;                          // فقط یک بار
    txtUserName.Enabled = txtPassword.Enabled = btnLogin.Enabled = true;
    wrong = 0;
}
```

۶۰ صفحه ۱۴۵ پژوهش

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

با public بودن فیلد، هر کلاس یا فرم دیگری می‌تواند بدون هیچ کنترلی مقدار آن را بخواند یا تغییر دهد؛ این کار اصل پنهان‌سازی اطلاعات را نقض می‌کند و ممکن است مقدار نامعتبر وارد شود و خطایابی دشوار شود.

- استفاده از ویژگی (Property) با get و set برای کنترل دسترسی و اعتبارسنجی مقدار،
- ارسال داده از طریق سازنده‌ی (Constructor) فرم دوم، مثلاً new MainForm(userName)،
- استفاده از متغیرهای ایستا (static) در یک کلاس مشترک،
- استفاده از خاصیت Tag کنترل‌ها یا خواندن از کنترل‌های internal فرم دیگر.

ایران توشه
توشه‌ای برای موفقیت

۶۱ صفحه ۱۴۶ فعالیت گروهی

فرض کنید در یک برنامه‌ی ویندوزی از بین نام ۳۰ هنرجو، می‌خواهیم یکی را انتخاب کنیم. به نظر شما کدام‌یک از کنترل‌ها برای طراحی این برنامه مناسب‌تر هستند؟ آیا لازم است نام ۳۰ هنرجو به‌طور کامل روی فرم نمایش داده شود؟

کادرهای انتخابی در دو شکل ۴۳ و ۴۴ را با هم مقایسه کنید. چه تفاوتی بین این دو کادر می‌بینید؟

کنترل‌های کادر فهرست (ListBox) و به‌ویژه کادر ترکیبی (ComboBox) مناسب‌اند. لازم نیست نام هر ۳۰ هنرجو همیشه روی فرم دیده شود؛ با این کنترل‌ها تعداد زیادی گزینه در فضای کم جا می‌گیرد و کاربر فقط در صورت نیاز فهرست را می‌بیند (یا در ListBox با نوار پیمایش آن را مرور می‌کند).

در کادر ترکیبی (شکل ۴۳) فقط عنصر انتخاب‌شده در یک کادر متن دیده می‌شود و با کلیک روی پیکان، فهرست باز (کشویی) می‌شود و کاربر می‌تواند متن را تایپ هم بکند؛ در کادر فهرست (شکل ۴۴) چند عنصر همیشه به‌صورت یک فهرست در فرم نمایش داده می‌شوند و کاربر فقط از میان آنها انتخاب می‌کند (فضای بیشتری اشغال می‌کند).

این فایل توسط سایت ایران‌توشه تهیه شده است.

irantooshe.ir

۶۲ صفحه ۱۴۸ کارگاه ۷

۲ ـ فرم مناسب را طراحی کنید. در کنترل کادر ترکیبی نام ۱۵ نامزد را به دلخواه وارد کنید.

۳ ـ متد رویداد کلیک دکمه‌ی btnAdd را بنویسید. با اجرای برنامه چند نامزد را می‌توان به فهرست منتخبان اضافه کرد؟

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

۵ ـ متد رویداد کلیک دکمه‌ی «حذف» را بنویسید.

۶ ـ متد رویداد کلیک دکمه‌ی «پاک‌کردن فهرست» را بنویسید.

کنترل‌های فرم: cboCandidates (کادر ترکیبی با ۱۵ نام در ویژگی Items)، lstResults (کادر فهرست منتخبان)، و دکمه‌های btnAdd، btnDelete، btnClear.

```
private void btnAdd_Click(object sender, EventArgs e)
{
    if (lstResults.Items.Count < 5)
    {
        if (cboCandidates.Text.Length > 0)
            lstResults.Items.Add(cboCandidates.Text);
    }
    else
        MessageBox.Show("حداکثر ۵ نامزد را می‌توانید انتخاب کنید");
}
private void btnDelete_Click(object sender, EventArgs e)
{
    int index = lstResults.SelectedIndex;
    if (index >= 0)
        lstResults.Items.RemoveAt(index);
}
private void btnClear_Click(object sender, EventArgs e)
{
    lstResults.Items.Clear();
}
```

چون شرط Items.Count < 5 است، حداکثر ۵ نامزد به فهرست منتخبان اضافه می‌شود؛ درخواست ششم پیام خطا می‌دهد (مرحله‌ی ۴).

۶۳ صفحه ۱۴۸ فعالیت کارگاهی

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

ـ دستوراتی به پروژه اضافه کنید تا شماره‌ی عنصر انتخاب‌شده در فهرست را در یک کادر متن نمایش دهد.

ویژگی Sorted کادر ترکیبی را True می‌کنیم تا عناصر به‌صورت الفبایی مرتب نمایش داده شوند. برای نمایش اندیس عنصر انتخاب‌شده یک کادر متن txtIndex اضافه کرده، در رویداد SelectedIndexChanged کادر فهرست می‌نویسیم:

```
private void lstResults_SelectedIndexChanged(object sender, EventArgs e)
{
    txtIndex.Text = lstResults.SelectedIndex.ToString();
}
```

۶۴ صفحه ۱۴۹ پروژه فاز ۳

برای برنامه‌ی کتابخانه‌ی هنرستان واسط گرافیکی مانند فرم زیر طراحی کنید: سربرگ اعضا (نام و نام خانوادگی، کد عضویت، تلفن، نشانی، کد ملی و عکس)، سربرگ جست‌وجو، دکمه‌های ثبت، جدید و درج تصویر؛

ایران توشه
توشه ای برای موفقیت

سربرگ کتاب؛ نمایش تاریخ و ساعت در نوار عنوان؛ فرم Login برای مدیر و غیرفعال‌شدن یک‌دقیقه‌ای پس از بیش از ۳ بار ورود اشتباه؛ غیرفعال بودن سربرگ‌های اعضا و کتاب در شروع و فعال‌شدن پس از ورود مدیر.

طراحی فرم اصلی (MainForm)

- tabControl1 با سربرگ‌های «اعضا»، «کتاب» و «جست‌وجو»؛
- سربرگ اعضا: txtFirstName، txtLastName، txtCode، txtPhone، txtAddress، txtNationalCode، picMember (کادر تصویر) و دکمه‌های btnSave، btnNew، btnPicture؛
- سربرگ کتاب: txtBookName، txtAuthor، txtTranslator، txtPublisher، txtYear، cboSubject، txtIsbn، picBook و دکمه‌های مشابه؛
- سربرگ جست‌وجو: دکمه‌های رادیویی (نام کتاب، مؤلف، موضوع، سال چاپ)، txtSearch و btnSearch؛
- دکمه‌ی btnAdminLogin («ورود مدیر») و timerClock با Interval = 1000.

ایران توشه
توشه ای برای موفقیت

```
string[] firstName = new string[100], lastName = new string[100], picPath = new
string[100];
int count = 0;

private void MainForm_Load(object sender, EventArgs e)
{
    tabMember.Enabled = false;      // سربرگ‌های اعضا و کتاب در شروع غیرفعال
    tabBook.Enabled = false;
}
private void timerClock_Tick(object sender, EventArgs e)
{
    this.Text = "کتابخانه  " + DateTime.Now.ToString();   // تاریخ و ساعت در نوار عنوان
}
private void btnSave_Click(object sender, EventArgs e)
{
    firstName[count] = txtFirstName.Text;
    lastName[count] = txtLastName.Text;
    picPath[count] = picMember.ImageLocation;
    count++;
}
private void btnNew_Click(object sender, EventArgs e)
{
    txtFirstName.Clear(); txtLastName.Clear(); txtCode.Clear();
    txtPhone.Clear(); txtAddress.Clear(); txtNationalCode.Clear();
    picMember.Image = null;
}
private void btnPicture_Click(object sender, EventArgs e)
{
    OpenFileDialog dlg = new OpenFileDialog();
    dlg.Filter = "Images|*.jpg;*.png;*.bmp";
    if (dlg.ShowDialog() == DialogResult.OK)
        picMember.ImageLocation = dlg.FileName;   // آدرس در آرایه ذخیره می‌شود (btnSave)
}
private void btnAdminLogin_Click(object sender, EventArgs e)
{
    LoginForm frm = new LoginForm();
    frm.ShowDialog();
    if (frm.succeeded)
        tabMember.Enabled = tabBook.Enabled = true;
}
```

ایران توشه
توشه ای برای موفقیت

فرم Login (قفل یک‌دقیقه‌ای پس از بیش از ۳ بار اشتباه)

```
public bool succeeded;
int wrong = 0;
private void btnLogin_Click(object sender, EventArgs e)
{
    if (txtUserName.Text == "admin" && txtPassword.Text == "1234")
    { succeeded = true; this.Close(); }
    else
    {
        wrong++;
        MessageBox.Show("نام کاربری یا گذرواژه نادرست است");
        if (wrong > 3)                        // بیش از ۳ بار
        {
            txtUserName.Enabled = txtPassword.Enabled = btnLogin.Enabled = false;
            timerLock.Interval = 60000;       // یک دقیقه
            timerLock.Enabled = true;
        }
    }
}
private void timerLock_Tick(object sender, EventArgs e)
{
    timerLock.Enabled = false;
    txtUserName.Enabled = txtPassword.Enabled = btnLogin.Enabled = true;
    wrong = 0;
}
```

طرح دقیق فرم (شکل کتاب) در دسترس نبود؛ نام کنترل‌ها پیشنهادی هستند.

ایران توشه
توشه ای برای موفقیت