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

پاسخ فعالیت‌ها و پرسش‌های کتاب

پایه یازدهم · شبکه و نرم‌افزار رایانه

شامل پاسخ فعالیت‌ها، پژوهش‌ها، کارهای گروهی و پرسش‌های کتاب

تهیه‌شده در ایران‌توشه، توشه‌ای برای موفقیت

irantooshe.ir

فهرست فصل‌ها

۱ پودمان اول — پیاده‌سازی پایگاه داده ۴

۲ پودمان دوم — مدیریت مجموعه داده ۴۴

۳ پودمان سوم — طراحی واسط گرافیکی ۸۳

۴ پودمان چهارم — توسعه‌ی واسط گرافیکی ۱۲۴

۵ پودمان پنجم — مدیریت پایگاه داده ۱۷۱

ایران توشه
توشه ای برای موفقیت

گام به گام توسعه برنامه‌سازی و پایگاه داده - پودمان اول

پیاده‌سازی پایگاه داده

فهرست

۱	صفحه ۳ · فعالیت گروهی — تاریخچه‌ی ذخیره‌سازی و بازیابی اطلاعات	۸
۲	صفحه ۴ · کنجکاوی — تعامل انسان با پایگاه داده	۹
۳	صفحه ۴ · فعالیت گروهی — داده‌های تکراری	۹
۴	صفحه ۵ · فعالیت گروهی — مشکلات جمع‌آوری پراکنده‌ی اطلاعات	۹
۵	صفحه ۶ · کنجکاوی — دفترچه‌ی تلفن بدون قاعده	۱۰
۶	صفحه ۷ · فعالیت گروهی — محیط‌های عملیاتی پیرامون	۱۱
۷	صفحه ۷ · کنجکاوی — میز و نیمکت به‌عنوان موجودیت	۱۱
۸	صفحه ۸ · پژوهش — تعریف موجودیت	۱۱
۹	صفحه ۸ · فعالیت کارگاهی — موجودیت‌های تاکسی تلفنی	۱۲
۱۰	صفحه ۸ · فعالیت کارگاهی — موجودیت‌های کتابخانه	۱۲
۱۱	صفحه ۸ · فعالیت کارگاهی — صفت‌های موجودیت	۱۲
۱۲	صفحه ۹ · کنجکاوی — معیار انتخاب صفت	۱۳
۱۳	صفحه ۹ · فعالیت گروهی — موجودیت، صفت و کلید	۱۳

ایران توشه
توشه‌ای برای موفقیت

۱۴ صفحه ۱۰ · پژوهش — معایب پایگاه داده ۱۴

۱۵ صفحه ۱۱ · فعالیت کارگاهی — ارتباط یک‌به‌یک ۱۴

۱۶ صفحه ۱۲ · فعالیت گروهی — ارتباط یک‌به‌چند ۱۵

۱۷ صفحه ۱۳ · کنجکاوی — نمایش گرافیکی رابطه‌ها ۱۵

۱۸ صفحه ۱۳ · فعالیت گروهی — ارتباط چندبه‌چند در کتابخانه ۱۶

۱۹ صفحه ۱۳ · فعالیت کارگاهی — نمودار ER فروشگاه و کافی‌نت ۱۶

۲۰ صفحه ۱۳ · فعالیت گروهی — نام ارتباط در مثال‌های چندبه‌چند ۱۷

۲۱ صفحه ۱۳ · فعالیت گروهی — تفاوت صفت کلیدی در نمودار ۱۸

۲۲ صفحه ۱۴ · فعالیت گروهی — تکمیل نمودار ER هنرآموز و هنرجو ۱۸

۲۳ صفحه ۱۴ · کنجکاوی — ارتباط موجودیت با خودش ۱۹

۲۴ صفحه ۱۶ · کنجکاوی — سیستم‌های مدیریت پایگاه داده ۱۹

۲۵ صفحه ۱۷ · فعالیت گروهی — موجودیت‌ها و فیلدهای هنرستان ۱۹

۲۶ صفحه ۱۸ · فعالیت کارگاهی — ایجاد پایگاه داده‌ی school ۲۰

۲۷ صفحه ۱۸ · پژوهش — قالب mdb و accdb ۲۱

۲۸ صفحه ۱۹ · کنجکاوی — ستون Click to Add ۲۱

۲۹ صفحه ۲۱ · فعالیت گروهی — انتخاب نوع داده‌ی فیلدها ۲۱

۳۰ صفحه ۲۲ · فعالیت کارگاهی — جدول Register ۲۲

۳۱ صفحه ۲۳ · فعالیت کارگاهی — جدول student در نمای Design ۲۳

ایران توشه
توشه ای برای موفقیت

۳۲ صفحه ۲۳ · کنجکاوی — روش‌های تعیین کلید اصلی ۲۴

۳۳ صفحه ۲۴ · کنجکاوی — پیام هشدار حذف فیلد ۲۴

۳۴ صفحه ۲۴ · فعالیت کارگاهی — کلید اصلی جدول student ۲۵

۳۵ صفحه ۲۵ · فعالیت کارگاهی — تغییر ساختار جدول student ۲۵

۳۶ صفحه ۲۵ · فعالیت کارگاهی — جدول‌های student، course، teacher و teach ۲۶

۳۷ صفحه ۲۸ · فعالیت کارگاهی — پایگاه داده‌ی Store ۲۸

۳۸ صفحه ۳۱ · کنجکاوی — روش‌های دیگر درج رکورد ۲۹

۳۹ صفحه ۳۴ · فعالیت کارگاهی — درج رکورد ۲۹

۴۰ صفحه ۳۴ · کارگاه ۲ — درج در جدول teach و کلید خارجی ۳۰

۴۱ صفحه ۳۵ · کنجکاوی — UPDATE بدون WHERE ۳۰

۴۲ صفحه ۳۵ · فعالیت کارگاهی — دستور UPDATE ۳۰

۴۳ صفحه ۳۶ · کنجکاوی — روش‌های دیگر حذف رکورد ۳۱

۴۴ صفحه ۳۷ · فعالیت کارگاهی — دستور DELETE ۳۱

۴۵ صفحه ۳۷ · کارگاه ۳ — حذف همه‌ی رکوردهای جدول ۳۲

۴۶ صفحه ۴۰ · فعالیت کارگاهی — پرس‌وجو با Wizard ۳۲

۴۷ صفحه ۴۲ · فعالیت کارگاهی — نام مستعار و ادغام فیلدها ۳۲

۴۸ صفحه ۴۳ · فعالیت کارگاهی — مرتب‌سازی نزولی ۳۳

۴۹ صفحه ۴۳ · کارگاه ۶ — آیا جدول اصلی مرتب می‌شود؟ ۳۳

ایران توشه
توشه‌ای برای موفقیت

۵۰ صفحه ۴۴ · فعالیت کارگاهی — مرتب‌سازی جدول course ۳۳

۵۱ صفحه ۴۴ · کارگاه ۷ — گروه‌بندی؛ تعیین فیلدها و جدول‌ها ۳۴

۵۲ صفحه ۴۵ · کارگاه ۷ — گروه‌بندی؛ شرط و WHERE ۳۴

۵۳ صفحه ۴۷ · فعالیت کارگاهی — گروه‌بندی بر اساس پایه و نام درس ۳۵

۵۴ صفحه ۵۰ · کارگاه ۸ — افزودن فیلد جدول دیگر به گزارش ۳۶

۵۵ صفحه ۵۱ · فعالیت کارگاهی — گزارش با Wizard ۳۶

۵۶ صفحه ۵۳ · فعالیت کارگاهی — ویرایش گزارش ثبت‌نام ۳۶

۵۷ صفحه ۵۴ · پروژه فاز ۱ — کتابخانه؛ موجودیت‌ها، صفات و نمودار ER ۳۷

۵۸ صفحه ۵۴ · پروژه فاز ۱ — کتابخانه؛ ایجاد پایگاه داده، جدول‌ها و ارتباط‌ها ۳۸

۵۹ صفحه ۵۴ · پروژه فاز ۱ — کتابخانه؛ ورود داده‌ها ۴۰

۶۰ صفحه ۵۴ · پروژه فاز ۱ — کتابخانه؛ پرس‌وجوهای ۱ تا ۴ ۴۰

۶۱ صفحه ۵۴ · پروژه فاز ۱ — کتابخانه؛ پرس‌وجوهای ۵ تا ۷ ۴۱

۶۲ صفحه ۵۴ · پروژه فاز ۱ — کتابخانه؛ گزارش‌ها ۴۳

ایران توشه
توشه ای برای موفقیت

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

۱ صفحه ۳ فعالیت گروهی

در خصوص موارد زیر در گروه گفت‌وگو کنید و نتیجه‌ی هر گروه را به کلاس ارائه دهید.

ـ نقش رایانه در ایجاد پایگاه‌های داده

ـ نگهداری اطلاعات به‌وسیله‌ی رایانه

ـ انفجار اطلاعات

ـ بشر از ابتدا تاکنون از چه ابزارهای ذخیره‌سازی استفاده کرده است؟

نقش رایانه در ایجاد پایگاه‌های داده

رایانه با سرعت بالا، حافظه‌ی بزرگ و قابلیت پردازش خودکار، ذخیره، جست‌وجو، ویرایش و حذف داده‌ها را بسیار آسان کرد. وجود رایانه سبب شد داده‌ها به شکل سازمان‌یافته (جدول، رکورد، فیلد) نگهداری شوند و پایگاه داده شکل بگیرد.

نگهداری اطلاعات به‌وسیله‌ی رایانه

اطلاعات در حافظه‌های جانبی (دیسک سخت، SSD، حافظه‌ی فلش) و امروزه در سرورها و فضای ابری ذخیره می‌شوند. مزیت‌ها: دسترسی سریع، حجم ذخیره‌سازی زیاد، کاهش مصرف کاغذ، امکان پشتیبان‌گیری و اشتراک‌گذاری در شبکه.

انفجار اطلاعات

یعنی رشد بسیار سریع حجم داده‌ها در دنیا (شبکه‌های اجتماعی، حسگرها، تراکنش‌های بانکی و...). این رشد باعث شده بدون پایگاه داده و ابزارهایی مانند داده‌کاوی، یافتن اطلاعات موردنیاز ممکن نباشد.

ابزارهای ذخیره‌سازی بشر

دیوار غارها و نقاشی، سنگ‌نوشته و لوح گلی، پاپیروس و پوست، کاغذ و کتاب، دفتر و پرونده‌های کاغذی، نوار و کارت‌های منگنه‌ای، دیسک‌های مغناطیسی و نوری (فلاپی، CD، DVD)، حافظه‌ی فلش، دیسک سخت و فضای ابری.

۲ صفحه ۴ کنجکاوی

چه فعالیت‌های دیگری را می‌توان نام برد که در آنها تعامل انسان با پایگاه داده وجود دارد. با دوستان خود گفت‌وگو کنید و حداقل دو مورد نام ببرید.

- خرید اینترنتی: ثبت و نمایش کالا، موجودی، سفارش و حساب کاربر در پایگاه داده‌ی فروشگاه.
- ثبت‌نام و دیدن کارنامه در سامانه‌ی مدرسه یا دانشگاه.
- رزرو بلیت قطار یا هواپیما و بررسی صندلی‌های خالی.
- جست‌وجو در موتورهای جست‌وجو، دیدن مخاطبین و پیام‌ها در تلفن همراه، شبکه‌های اجتماعی، نوبت‌دهی بیمارستان و دریافت قبض‌ها.

۳ صفحه ۴ فعالیت گروهی

مشکلی که در این روش جمع‌آوری اطلاعات وجود دارد چیست؟

هر فرد (معاون اجرایی، معاون آموزشی، هنرآموزان) اطلاعات یکسان هنرجویان را جداگانه از پرونده‌ها استخراج و ثبت می‌کند. مشکل این روش **تکرار داده‌ها (Redundancy)** است.

- وقت و نیروی انسانی بی‌جهت هدر می‌رود، زیرا یک کار چندبار انجام می‌شود.
- کاغذ و فضای ذخیره‌سازی بیشتری مصرف می‌شود.
- احتمال خطا و ناهمسانی اطلاعات بالا می‌رود.
- راه‌حل: اطلاعات یک بار ثبت شود و همه از یک منبع مشترک (پایگاه داده) استفاده کنند.

۴ صفحه ۵ فعالیت گروهی

مشکلات به‌وجود آمده در خصوص جمع‌آوری اطلاعات هنرجویان ناشی از چیست؟ چه راهکاری پیشنهاد می‌کنید؟ در این باره با هم‌کلاسی‌های خود گفت‌وگو کنید.

این مشکلات ناشی از آن است که هر فرد اطلاعات را به‌صورت جداگانه و در دفتر یا فهرست خودش نگهداری می‌کند؛ بنابراین سه مشکل پدید می‌آید:

- **تکرار داده‌ها:** یک اطلاعات چندبار ثبت می‌شود.
- **تداخل و مغایرت:** نمی‌توان تضمین کرد فهرست‌ها یکسان باشند (مثلاً نام‌خانوادگی کامل در یک فهرست و ناقص در فهرست دیگر).

ایران توشه
توشه‌ای برای موفقیت

- **دشواری بهروزرسانی:** برای اصلاح یک نام باید همه‌ی فهرست‌ها را اصلاح کرد و ممکن است کسی فراموش کند.

راهکار: ایجاد یک پایگاه داده‌ی مرکزی برای هنرستان. همه‌ی افراد از یک منبع مشترک داده می‌گیرند، هر اطلاعات فقط یک بار ثبت می‌شود و با یک بار اصلاح، در همه‌جا درست می‌شود. مسئول کتابخانه هم همیشه به اطلاعات به‌روز دسترسی دارد.

۵ صفحه ۶ کنجکاوی

دفترچه تلفنی را در نظر بگیرید که در آن نام و نام خانوادگی افراد و تلفن آنها پشت سر هم و بدون هیچ قاعده‌ای ثبت شده است. به نظر شما این روش چه مشکلی در پی خواهد داشت؟ دلیل خود را ذکر کرده، راه‌حل مورد نظر را ارائه دهید.

چون داده‌ها بدون ساختار و ترتیب نوشته شده‌اند:

- پیدا کردن یک شماره کند است، زیرا باید همه‌ی دفترچه را از ابتدا بخوانیم.
- نام، نام‌خانوادگی و تلفن از هم جدا نیستند؛ پس جست‌وجو و مرتب‌سازی بر اساس هرکدام ممکن نیست.
- ممکن است یک شخص چندبار ثبت شود و ویرایش یا حذف او دشوار است.

راه‌حل: داده‌ها را در جدولی با ستون‌های مشخص (نام، نام‌خانوادگی، تلفن) و به‌صورت مرتب‌شده بر اساس نام‌خانوادگی نگهداری کنیم؛ بهتر است برای هر فرد یک کد یکتا در نظر بگیریم و دفترچه را در یک پایگاه داده (مثلاً Access) ذخیره کنیم.

ایران توشه
توشه ای برای موفقیت

۶ صفحه ۷ فعالیت گروهی

محیط‌های عملیاتی پیرامون خود را شناسایی کنید و درباره‌ی آن با هم‌کلاسی‌های خود گفت‌وگو کنید.

محیط عملیاتی	مشکل و ضرورت استفاده از پایگاه داده
هنرستان	ثبت جداگانه‌ی اطلاعات هنرجویان، نمرات و حضور و غیاب؛ نیاز به گزارش سریع
کتابخانه	پیگیری امانت و بازگشت کتاب‌ها، جلوگیری از گم شدن اطلاعات اعضا
فروشگاه	کنترل موجودی کالا، صدور فاکتور و گزارش فروش
بانک	ثبت میلیون‌ها تراکنش، امنیت و دسترسی هم‌زمان چندین کاربر
داروخانه / بیمارستان	نگهداری پرونده‌ی بیماران و موجودی دارو، دقت بالا و دسترسی سریع
تاکسی تلفنی	ثبت سفارش مشتری‌ها و مدیریت رانندگان

۷ صفحه ۷ کنجکاوی

ـ آیا در محیطی مانند هنرستان، می‌توان میز و نیمکت‌ها را نیز به‌عنوان موجودیت در نظر گرفت؟

ـ برای پاسخ خود دلیل بیاورید.

ـ معیار انتخاب موجودیت چیست؟

بستگی به نیاز سیستم دارد. در پایگاه دادهای که کارش مدیریت هنرجویان، هنرآموزان و دروس است، میز و نیمکت موجودیت نیستند، زیرا نمی‌خواهیم درباره‌ی آنها اطلاعاتی ذخیره و بازیابی کنیم. اما اگر هدف سیستم «مدیریت اموال هنرستان» باشد، میز و نیمکت موجودیت هستند (با صفت‌هایی مانند کد اموال، تعداد، وضعیت و کلاس محل استقرار).

معیار انتخاب موجودیت: هر شخص، مکان، شیء یا مفهومی که در سناریوی محیط عملیاتی نقش دارد، قابل شناسایی از بقیه است و می‌خواهیم اطلاعاتی درباره‌ی آن ذخیره کنیم.

۸ صفحه ۸ پژوهش

در مورد تعریف موجودیت در منابع مختلف تحقیق کنید.

- **Entity** (موجودیت): چیزی در دنیای واقعی که وجود مستقل دارد، از دیگر اشیا قابل تشخیص است و می‌خواهیم داده‌هایی درباره‌ی آن نگهداری کنیم (مدل Chen).
- در مدل رابطه‌ای هر موجودیت به یک جدول تبدیل می‌شود؛ صفت‌های آن ستون‌ها و هر نمونه‌ی آن یک سطر (رکورد) است.

- موجودیت می‌تواند فیزیکی (هنرجو، کتاب، خودرو) یا مفهومی (درس، حساب بانکی، سفارش) باشد.
- موجودیت قوی بدون وابستگی به موجودیت دیگر وجود دارد؛ موجودیت ضعیف برای وجود خود به موجودیت دیگری وابسته است (مانند عضو خانواده‌ی کارمند).

۹ صفحه ۸ فعالیت کارگاهی

ـ با در نظر گرفتن محیط عملیاتی تاکسی‌تلفنی، جدول زیر را تکمیل کنید.

محیط عملیاتی | موجودیت

تاکسی‌تلفنی | راننده

محیط عملیاتی	موجودیت
تاکسی‌تلفنی	راننده
تاکسی‌تلفنی	مشتری (مسافر)
تاکسی‌تلفنی	خودرو (تاکسی)
تاکسی‌تلفنی	اپراتور
تاکسی‌تلفنی	سفر (درخواست)

۱۰ صفحه ۸ فعالیت کارگاهی

ـ موجودیت‌های محیط عملیاتی کتابخانه را نام ببرید؟

- عضو، کتاب، کتابدار، ناشر، نویسنده، موضوع (دسته‌بندی کتاب).
- امانت نیز در عمل یک ارتباط (یا جدول رابطه) میان عضو و کتاب است.

۱۱ صفحه ۸ فعالیت کارگاهی

جدول زیر را تکمیل کنید.

موجودیت | محیط عملیاتی | صفت‌ها

هنرجو

تاکسی تلفنی

مشتری فروشگاه

موجودیت	محیط عملیاتی	صفت‌ها
هنرجو	هنرستان	شماره‌ی دانش‌آموزی، نام، نام‌خانوادگی، کد ملی، تاریخ تولد، پایه، رشته، تلفن
راننده	تاکسی تلفنی	کد راننده، نام، نام‌خانوادگی، شماره‌ی گواهی‌نامه، پلاک خودرو، تلفن
مشتری	فروشگاه	کد مشتری، نام، نام‌خانوادگی، تلفن، نشانی

۱۲ صفحه ۹ **کنجکاوی**

ـ آیا می‌توان تمامی صفت‌های مربوط به یک موجودیت را در محیط عملیاتی مانند هنرستان در نظر گرفت؟

ـ معیار انتخاب صفت برای یک موجودیت چیست؟

ـ برای پاسخ خود دلیل بیاورید.

خیر. هر موجودیت صفت‌های بی‌شماری دارد (قد، وزن، گروه خونی، رنگ چشم، غذای مورد علاقه و...) و ذخیره‌ی همه‌ی آنها ممکن و لازم نیست؛ حجم پایگاه داده و کار ثبت داده را بی‌دلیل زیاد می‌کند.

معیار: فقط صفت‌هایی را انتخاب می‌کنیم که برای نیازهای اطلاعاتی آن محیط عملیاتی ضروری‌اند. در هنرستان قد و وزن هنرجو لازم نیست، اما نام، نام‌خانوادگی، شماره‌ی دانش‌آموزی و پایه لازم است. همچنین چون نام «علی» تکراری است و برای تشخیص یکتای هنرجو کافی نیست، صفتی مانند شماره‌ی دانش‌آموزی (کلید) لازم است.

۱۳ صفحه ۹ **فعالیت گروهی**

در جدول زیر برای هر محیط عملیاتی، یک موجودیت را انتخاب کرده، صفات آن را بنویسید. سپس یکی از این صفات را به‌عنوان کلید اصلی انتخاب کنید.

محیط عملیاتی | موجودیت | صفات موجودیت | صفت کلیدی

اداره

ایران توشه
توشه ای برای موفقیت

کتابخانه

محیط عملیاتی	موجودیت	صفات موجودیت	صفت کلیدی
اداره	کارمند	شماره‌ی کارمندی، نام، نام‌خانوادگی، کد ملی، سمت، تاریخ استخدام، حقوق	شماره‌ی کارمندی
کتابخانه	کتاب	شابک، عنوان، نویسنده، ناشر، موضوع، قیمت، سال انتشار	شابک (یا کد کتاب)

۱۴ صفحه ۱۰ **پژوهش**

تحقیق کنید استفاده از پایگاه داده چه معایبی دارد؟

- هزینه‌ی راه‌اندازی: خرید نرم‌افزار و سخت‌افزار مناسب، و آموزش یا استخدام متخصص.
- پیچیدگی طراحی و نگهداری، به‌ویژه برای پایگاه‌های بزرگ.
- تمرکز داده‌ها: اگر پایگاه داده خراب شود یا از کار بیفتد، کار همه‌ی کاربران متوقف می‌شود؛ پس پشتیبان‌گیری لازم است.
- آسیب‌پذیری امنیتی: دسترسی غیرمجاز به یک نقطه، همه‌ی داده‌ها را در معرض خطر قرار می‌دهد.
- برای کاربردهای کوچک و ساده (مثل یک فهرست کوتاه) پرهزینه و غیرضروری است.

۱۵ صفحه ۱۱ **فعالیت کارگاهی**

با در نظر گرفتن موجودیت‌های کشور و پایتخت، در جداول زیر هر کشور را به پایتخت خود متصل کنید.
نتیجه‌گیری خود را از انجام این فعالیت بنویسید.

پایتخت | نام کشور
آنکارا | ایران
تهران | چین
پکن | ژاپن
توکیو | ترکیه

اتصال کشورها به پایتخت

ایران توشه
توشه‌ای برای موفقیت

نام کشور	پایتخت
ایران	تهران
چین	پکن
ژاپن	توکیو
ترکیه	آنکارا

نتیجه: هر کشور فقط یک پایتخت دارد و هر پایتخت فقط متعلق به یک کشور است؛ بنابراین ارتباط بین این دو موجودیت **یک‌به‌یک (1:1)** است.

۱۶ صفحه ۱۲ **فعالیت گروهی**

در جدول‌های زیر نام دو هنرآموز خود را مشخص کرده، برای هرکدام در جدول دیگر عنوان‌های درسی را در نظر بگیرید. سپس اتصال صحیح را انجام دهید. نتیجه‌گیری خود را از انجام این فعالیت بنویسید.

عنوان‌های درسی | نام هنرآموز

نمونه‌ی اتصال (نام‌ها فرضی است)

هنرآموز	عنوان‌های درسی
هنرآموز اول (آقای احمدی)	پایگاه داده، برنامه‌سازی
هنرآموز دوم (خانم کریمی)	شبکه، طراحی وب، سیستم‌عامل

نتیجه: هر هنرآموز می‌تواند چند عنوان درسی را تدریس کند، اما هر عنوان درسی (در این فرض) فقط به یک هنرآموز تعلق دارد. پس ارتباط هنرآموز با درس **یک‌به‌چند (N:1)** است: هر نمونه از موجودیت اول با چند نمونه از موجودیت دوم ارتباط دارد، ولی هر نمونه از موجودیت دوم تنها با یک نمونه از اول.

ایران توشه
توشه ای برای موفقیت

۱۷ صفحه ۱۳ **کنجکاوی**

به نظر شما چگونه می‌توانیم رابطه بین موجودیت‌ها و عملکردی که بین آنها در یک محیط عملیاتی وجود دارد را به‌صورت گرافیکی نشان دهیم و یک مدل معنایی ایجاد کنیم که بتواند در طراحی پایگاه داده و برقراری ارتباط ما را یاری کند؟

با رسم **نمودار موجودیت–ارتباط (ERD)**: موجودیت‌ها با مستطیل، صفت‌ها با بیضی (صفت کلیدی با زیرخط)، ارتباط‌ها با لوزی و نوع ارتباط (1 یا N یا M) روی خط‌های اتصال نمایش داده می‌شود. این نمودار مدل معنایی محیط عملیاتی است و پایه‌ی طراحی جدول‌ها و ارتباط بین آنها قرار می‌گیرد.

۱۸ صفحه ۱۳ فعالیت گروهی

در محیط عملیاتی کتابخانه در هنرستان یک نمونه ارتباط چند به چند را در نظر بگیرید و آن را به‌صورت دو جدول مجزا نمایش دهید سپس ارتباط‌های بین آنها را مشخص کنید. نتیجه‌گیری خود را از انجام این فعالیت بنویسید.

ارتباط «امانت» بین موجودیت‌های عضو و کتاب، چندبه‌چند است.

جدول عضو و کتاب‌هایی که امانت گرفته است

عضو	کتاب‌های امانت‌گرفته
علی احمدی	برنامه‌سازی #C، Access
سارا کریمی	برنامه‌سازی #C، شبکه
رضا مرادی	Access

جدول کتاب و اعضایی که آن را امانت گرفته‌اند

کتاب	اعضا
برنامه‌سازی #C	علی احمدی، سارا کریمی
Access	علی احمدی، رضا مرادی
شبکه	سارا کریمی

نتیجه: یک عضو می‌تواند چند کتاب را امانت بگیرد و یک کتاب نیز در زمان‌های مختلف توسط چند عضو امانت گرفته می‌شود؛ پس ارتباط **چندبه‌چند (N:M)** است. این ارتباط با دو جدول به‌تنهایی نمایش داده نمی‌شود و برای پیاده‌سازی به جدول سومی (جدول رابطه‌ی «امانت») نیاز داریم که کلید اصلی هر دو جدول را به‌عنوان کلید خارجی نگه می‌دارد.

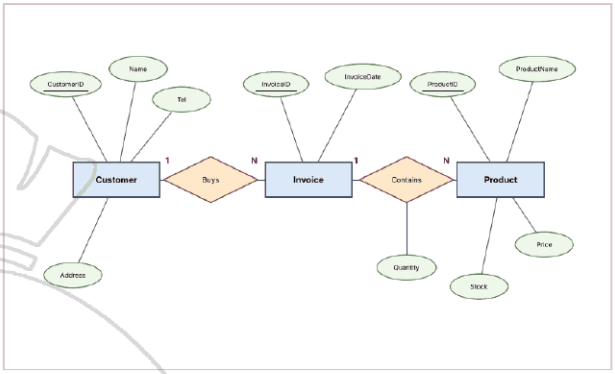
ایران توشه
توشه ای برای موفقیت

۱۹ صفحه ۱۳ فعالیت کارگاهی

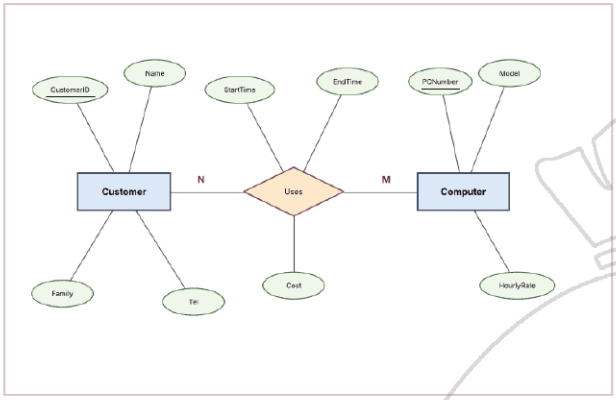
با در نظر گرفتن محیط‌های عملیاتی فروشگاه و کافی‌نت ارتباط موجودیت‌ها و نوع آنها را شناسایی کرده، نمودار آنها را رسم کنید.

ارتباط‌ها

محیط	موجودیت‌ها	ارتباط	نوع
فروشگاه	مشتری — فاکتور	خرید (Buys)	یک‌به‌چند (N:1)
فروشگاه	فاکتور — کالا	شامل (Contains)	چندبه‌چند (N:M)، با صفت تعداد
کافی‌نت	مشتری — رایانه	استفاده (Uses)	چندبه‌چند (N:M)، با صفت‌های ساعت شروع و پایان و هزینه



نمودار ER فروشگاه



نمودار ER کافی‌نت

۲۰ صفحه ۱۳ **فعالیت گروهی**

نام ارتباط یا عملکرد را در مثال‌های مطرح‌شده در ارتباطات چند به چند بالا شرح دهید؟ در این‌باره با هم‌کلاسی‌های خود گفت‌وگو کنید.

- هنرآموز و هنرستان: رابطه‌ی «تدریس در»؛ هنرآموز در چند هنرستان تدریس می‌کند.
- مترجم و کتاب: رابطه‌ی «ترجمه»؛ مترجم چند کتاب را ترجمه می‌کند و کتاب را چند مترجم ترجمه کرده‌اند.
- عضو و کتاب در کتابخانه: رابطه‌ی «امانت گرفتن».
- نام ارتباط معمولاً یک فعل است که عمل بین دو موجودیت را بیان می‌کند و داخل لوزی نوشته می‌شود.

ایران توشه
توشه‌ای برای موفقیت

۲۱ صفحه ۱۳ فعالیت گروهی

صفت شماره کارمندی و نام‌خانوادگی چه تفاوتی در نمودار دارند؟

شماره‌ی کارمندی **صفت کلیدی** است، زیرا برای هر کارمند منحصربه‌فرد است؛ در نمودار ER زیر آن **خط کشیده می‌شود**. نام‌خانوادگی ممکن است تکراری باشد و صفت معمولی است؛ پس بدون زیرخط نمایش داده می‌شود.

۲۲ صفحه ۱۴ فعالیت گروهی

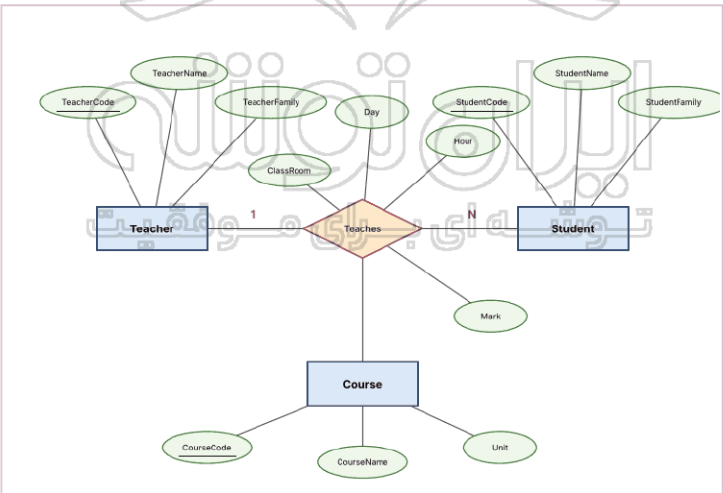
موارد خواسته‌شده را به کمک هم‌کلاسی‌های خود انجام دهید.

ـ در ارتباط تدریس بین هنرآموز و هنرجو، موجودیت درس و صفت‌های آن را تعریف کرده، با اضافه کردن آن، نمودار قبلی را کامل کنید.

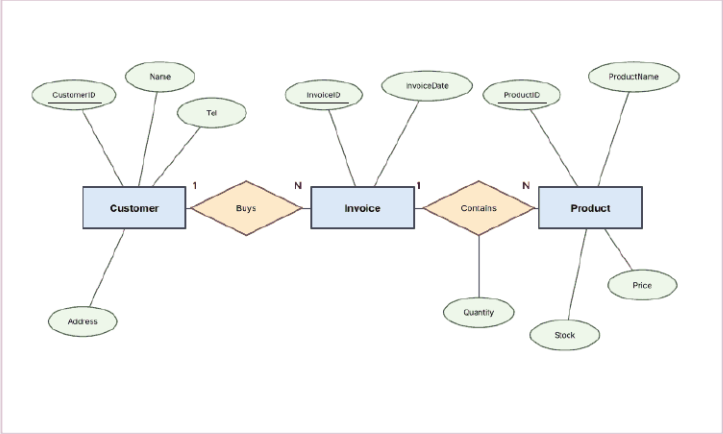
ـ صفت نمره را در محل مناسب، به نمودار ER اضافه کنید.

ـ برای محیط عملیاتی فروشگاه، نمودار ER رسم کنید.

- موجودیت درس (Course) با صفت‌های کد درس (کلید)، نام درس و تعداد واحد به نمودار اضافه و به لوزی «تدریس» وصل می‌شود.
- نمره مربوط به یک هنرجو در یک درس است، نه فقط به هنرجو یا درس؛ پس صفت **نمره** به **لوزی ارتباط** (کنار کلاس درس، روز درس و ساعت درس) وصل می‌شود.



نمودار ER فروشگاه (مشتری، فاکتور، کالا) در همین قسمت برای فعالیت قبل رسم شد:



۲۳ صفحه ۱۴ کنجکاوی

مثالی ارائه دهید که موجودیت با خودش در ارتباط باشد.

به این ارتباط، ارتباط بازگشتی (Recursive) می‌گویند. مثال‌ها:

- موجودیت کارمند: «مدیر بودن»؛ هر کارمند یک مدیر دارد و یک مدیر چند کارمند را مدیریت می‌کند (کارمند ← کارمند، N:1).
- موجودیت درس: «پیش‌نیاز بودن»؛ یک درس می‌تواند پیش‌نیاز چند درس دیگر باشد.
- موجودیت شخص: «ازدواج» (1:1).

۲۴ صفحه ۱۶ کنجکاوی

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

MySQL، Oracle Database، Microsoft SQL Server، PostgreSQL، SQLite، MariaDB، IBM DB2، Microsoft Access و (از نوع غیررابطه‌ای) MongoDB.

۲۵ صفحه ۱۷ فعالیت گروهی

با شناسایی تعدادی از موجودیت‌های محیط عملیاتی هنرستان، فیلدهای آن را مشخص و در جدولی مانند شکل ۱۳ نمایش دهید.

جدول student

ایران توشه
توشه ای برای موفقیت

شماره دانش‌آموزی	نام	نام خانوادگی	پایه	رشته
856479665	حسین	عزیزی	یازدهم	شبکه و نرم‌افزار رایانه
968214763	علی	محمدی	دهم	الکترونیک

جدول teacher

کد کارمندی	نام	نام خانوادگی	رشته	تلفن
1011	علی	رنجبر	رایانه	34558896
1012	سارا	کریمی	الکترونیک	22334455

جدول course

کد درس	نام درس	تعداد واحد
5563	پایگاه داده	3
5564	برنامه‌سازی	4

هر ستون یک فیلد و هر سطر یک رکورد است.

۲۶ صفحه ۱۸ **فعالیت کارگاهی**

پایگاه داده‌ای با نام school در پوشه‌ای به نام DB ایجاد کنید.

1. نرم‌افزار Access را اجرا کنید و روی Blank desktop database کلیک کنید.
2. در کادر File Name نام school را بنویسید.
3. روی نماد پوشه‌ی کنار کادر کلیک کنید، در پنجره‌ی باز شده محل دلخواه (مثلاً درایو D) را انتخاب کنید و با دکمه‌ی New Folder پوشه‌ای به نام DB بسازید و آن را باز کنید.
4. OK و سپس Create را بزنید. پرونده‌ی school.accdb ساخته می‌شود و با یک جدول پیش‌فرض باز می‌شود.

۲۷ صفحه ۱۸ پژوهش

درباره‌ی تغییر قالب پرونده‌های Access از mdb به accdb تحقیق کنید و تفاوت‌های این دو قالب را بررسی کنید.

ویژگی	mdb (تا 2003)	accdb (از 2007)
موتور	Jet	ACE (جدیدتر)
رمزگذاری	ساده‌تر و ضعیف‌تر	قوی‌تر با رمز عبور پایگاه داده
نوع داده‌ی جدید	ندارد	Attachment، فیلد چندمقداری، فیلد محاسباتی و Long Text با قالب‌بندی
حجم و کارایی	بیشینه‌ی 2 گیگابایت	بیشینه‌ی 2 گیگابایت ولی کارایی و پایداری بیشتر
سازگاری	با نسخه‌های قدیمی هم باز می‌شود	فقط در Access 2007 به بعد باز می‌شود

۲۸ صفحه ۱۹ کنجکاوی

به نظر شما، دلیل وجود ستون Click to Add چیست؟

این ستون برای **افزودن سریع فیلد جدید** در نمای Datasheet است؛ با کلیک روی آن فهرستی از نوع داده‌ها باز می‌شود، نوع فیلد را انتخاب می‌کنیم و ستون جدیدی به جدول افزوده می‌شود که می‌توان نام آن را تایپ کرد. به این ترتیب بدون رفتن به نمای Design می‌توان ساختار جدول را گسترش داد.

۲۹ صفحه ۲۱ فعالیت گروهی

در جدول زیر با توجه به مقادیر هر فیلد، نوع داده‌ای مناسب را در ستون نوع فیلد انتخاب کنید.

ایران توشه
توشه ای برای موفقیت

نام فیلد | نوع فیلد | توضیحات | دلیل انتخاب نوع داده‌ای

نام فیلد	نوع فیلد	توضیحات	دلیل انتخاب
کد ملی	Short Text	شامل ارقام	عملیات ریاضی روی آن انجام نمی‌شود و صفر ابتدایی نباید حذف شود
نام خانوادگی	Short Text	نام خانوادگی کامل هنرجو	متن کوتاه (تا 255 نویسه)
رشته	Lookup Wizard	فهرست رشته‌ها با قابلیت انتخاب	مقدار از یک فهرست آماده انتخاب می‌شود
وضعیت قبولی	Yes/No	دو وضعیت قبول یا مردود	فقط دو مقدار منطقی دارد
نشانی	Long Text	نشانی محل سکونت	ممکن است طولانی‌تر از 255 نویسه باشد
تلفن	Short Text	تلفن ثابت هنرجو	صفر ابتدایی و خط تیره دارد و محاسبه نمی‌شود
تاریخ تولد	Date/Time	سال، ماه و روز	مقدار تاریخ است و امکان محاسبه‌ی سن دارد
عکس	Attachment	عکس پرسنلی هنرجو	پرونده‌ی تصویری ضمیمه می‌شود (OLE Object هم ممکن است)
رایانامه	Hyperlink	ذخیره‌ی نشانی رایانامه	پیوند mailto ساخته می‌شود (Short Text هم قابل قبول است)
معدل	Number	معدل به‌صورت اعشاری	عدد اعشاری (Field Size برابر Single یا Double)
حقوق	Currency	حقوق دریافتی کارمند	مقدار پولی با قالب پولی و دقت بالا

ایران توشه
توشه‌ای برای موفقیت

۳۰ صفحه ۲۲ **فعالیت کارگاهی**

یک پایگاه داده ایجاد کرده و جدول Register آن را با توجه به فیلدهای جدول زیر ایجاد کنید.

نام فیلد | نوع داده | ویژگی‌ها

MemberID | Number | اعداد صحیح، نمایش «کد عضویت» به جای نام فیلد

Name | Short Text | حداکثر 20 نویسه، نمایش «نام» به جای نام فیلد، ورود داده اجباری

Family | Short Text | حداکثر 25 نویسه، نمایش «نام خانوادگی» به جای نام فیلد، ورود داده اجباری

birth_of_date | Short Text | نمایش «تاریخ تولد» به جای نام فیلد، قالب ورود داده به‌صورت 0000/00/00

Average | Number | اعداد اعشاری، نمایش «معدل» به جای نام فیلد، عدد وارد شده در محدوده‌ی 0 تا 20 باشد در غیر این‌صورت پیغام خطای مناسبی نمایش داده شود.

Field | Short Text | نمایش «رشته تحصیلی» به جای نام فیلد، مقدار پیش‌فرض «رایانه» در این فیلد قرار گیرد.

در زبانه‌ی Create روی Table Design کلیک کنید و فیلدها را با تنظیمات زیر در Field Properties وارد کنید، سپس جدول را با نام Register ذخیره کنید.

نام فیلد	نوع داده	ویژگی‌های تنظیم‌شده
MemberID	Number	Field Size = Long Integer؛ Caption = کد عضویت (کلید اصلی)
Name	Short Text	Field Size = 20؛ Caption = نام؛ Required = Yes
Family	Short Text	Field Size = 25؛ Caption = نام خانوادگی؛ Required = Yes
birth_of_date	Short Text	Caption = تاریخ تولد؛ Input Mask = 0000/00/00
Average	Number	Field Size = Single؛ Decimal Places = 2؛ Caption = معدل؛ Validation Rule = >=0 And <=20؛ Validation Text = معدل باید عددی بین 0 تا 20 باشد.
Field	Short Text	Caption = رشته تحصیلی؛ Default Value = "رایانه"

ایران توشه
توشه ای برای موفقیت

۳۱ صفحه ۲۳ فعالیت کارگاهی

جدولی با نام student در نمای Design برای پایگاه داده school ایجاد کرده، فیلدهای مورد نیاز هنرجویان را در آن وارد کرده، نوع داده‌ای هر یک از این فیلدها را تعیین کنید.

ساختار جدول student

نام فیلد	نوع داده	توضیح
StudentCode	(Long Integer) Number	شماره‌ی دانش‌آموزی (کلید اصلی)
StudentName	(20) Short Text	نام
StudentFamily	(25) Short Text	نام خانوادگی
NationalCode	(10) Short Text	کد ملی
StudentTel	(12) Short Text	تلفن
BirthDate	Date/Time	تاریخ تولد
Grade	(10) Short Text	پایه

جدول را از زبانه‌ی Create ← Table Design می‌سازیم، نام و نوع هر فیلد را وارد می‌کنیم و با Ctrl+S با نام student ذخیره می‌کنیم.

۳۲ صفحه ۲۳ کنجکاوی

ـ چه روش دیگری برای تعیین کلید اصلی وجود دارد؟

ـ اگر جدول را بدون ایجاد کلید اصلی ذخیره کنید چه اتفاقی می‌افتد؟

- روش دیگر: راست‌کلیک روی ناحیه‌ی خاکستری سمت چپ نام فیلد در نمای Design و انتخاب Primary Key. در نمای Datasheet هم فیلد ID به‌طور خودکار کلید اصلی است.
- اگر بدون کلید اصلی ذخیره کنیم، Access پیامی می‌دهد: «There is no primary key defined...» و می‌پرسد آیا اکنون کلید اصلی ساخته شود. با Yes، فیلد ID از نوع AutoNumber به‌عنوان کلید اصلی افزوده می‌شود؛ با No جدول بدون کلید ذخیره می‌شود (ممکن است رکورد تکراری وارد شود و ارتباط‌دادن جدول مشکل باشد)؛ با Cancel ذخیره لغو می‌شود.

۳۳ صفحه ۲۴ کنجکاوی

پیام هشدار هنگام حذف فیلد شامل داده را به همراه ترجمه‌ی آن بنویسید.

```
Do you want to permanently delete the selected field(s) and all the data
in the field(s)? To permanently delete the field(s), click Yes.
```

ترجمه: «آیا می‌خواهید فیلد (های) انتخاب‌شده و همه‌ی داده‌های موجود در آن فیلد (ها) را برای همیشه حذف کنید؟ برای حذف دائمی فیلد (ها) روی Yes کلیک کنید.»

حذف فیلد قابل بازگشت (Undo) نیست، پس قبل از آن از پایگاه داده نسخه‌ی پشتیبان تهیه کنید.

۳۴ صفحه ۲۴ فعالیت کارگاهی

برای جدول student، کلید اصلی تعیین کنید.

1. جدول student را در نمای Design باز کنید.
2. روی فیلد StudentCode کلیک کنید (هر هنرجو شماره‌ی دانش‌آموزی یکتا دارد).
3. در زبانه‌ی Design، گروه Tools، روی Primary Key کلیک کنید تا نماد کلید کنار فیلد ظاهر شود و جدول را ذخیره کنید.

۳۵ صفحه ۲۵ فعالیت کارگاهی

در جدول student:

ـ فیلد جدید معدل برای نگهداری مقادیر اعشاری با دقت دو رقم اعشار را اضافه کنید.

ـ کلید اصلی جدول را به شماره ملی هنرجو تغییر دهید.

ـ فیلدهای غیرضروری را حذف و موارد دیگر را به جدول اضافه کنید تا فیلدهای دانش‌آموز تکمیل شود.

ـ رکوردهایی را به جدول اضافه کنید.

ـ نوع فیلد معدل را به Integer تغییر دهید و نتیجه‌ی این کار را در کلاس شرح دهید.

- **افزودن معدل:** در نمای Design فیلدی با نام Average و نوع Number اضافه کنید، Field Size را Single (یا Double) و Decimal Places را 2 بگذارید.
- **تغییر کلید اصلی:** فیلد NationalCode (از نوع Short Text، 10 نویسه) را اضافه کنید، روی آن راست‌کلیک و Primary Key را انتخاب کنید. کلید قبلی خودکار برداشته می‌شود.
- **حذف فیلدهای غیرضروری** (مثلاً قد و وزن) با Delete Rows و افزودن فیلدهای لازم (پایه، رشته، تاریخ تولد و...).
- **افزودن رکورد:** در نمای Datasheet داده‌ها را در سطر آخر وارد کنید.

نتیجه‌ی تغییر نوع معدل به Integer: فیلد فقط عدد صحیح نگه می‌دارد؛ بنابراین بخش اعشاری داده‌ها گرد می‌شود و از بین می‌رود (مثلاً 17.65 به 18 و 14.2 به 14 تبدیل می‌شود). Access قبل از ذخیره هشدار «Some data may be lost» می‌دهد. پس تغییر نوع فیلد حاوی داده، خطر از دست رفتن اطلاعات دارد و باید با احتیاط و پس از پشتیبان‌گیری انجام شود.

ایران توشه
توشه‌ای برای موفقیت

۳۶ صفحه ۲۵ فعالیت کارگاهی

جدول‌های زیر را در پایگاه داده school، ایجاد و جدول student را نیز طبق جدول مربوطه اصلاح کنید. زیر فیلد کلید اصلی در هر جدول خط کشیده شده است. بین جدول‌های teacher با student، teach با teach و course با teach، ارتباط N:1 برقرار است. در این محیط عملیاتی فرض شده است که هر درس به‌وسیله‌ی چندین هنرآموز تدریس می‌شود و برعکس (ارتباط N:M) و هر هنرجو چندین هنرآموز دارد و برعکس (ارتباط N:M).

جدول teacher: TeacherCode، TeacherName، TeacherFamily، TeacherTel

جدول course: CourseCode، CourseName، Unit

جدول student: StudentCode، StudentName، StudentFamily، StudentTel

جدول teach: RowID، CourseCode، TeacherCode، StudentCode، EducationYear، Mark

ساختار جدول‌ها (کلید اصلی با ★)

ایران توشه
توشه ای برای موفقیت

جدول	فیلد	نوع داده	توضیح
teacher	★ TeacherCode	Long) Number (Integer	کد کارمندی
	TeacherName	(20) Short Text	نام هنرآموز
	TeacherFamily	(25) Short Text	نام خانوادگی هنرآموز
	TeacherTel	(12) Short Text	تلفن هنرآموز
course	★ CourseCode	Long) Number (Integer	کد درس
	CourseName	(40) Short Text	نام درس
	Unit	(Byte) Number	تعداد واحد
student	★ StudentCode	Long) Number (Integer	شماره‌ی دانش‌آموزی
	/ StudentName StudentFamily	(25 / 20) Short Text	نام و نام خانوادگی
	StudentTel	(12) Short Text	تلفن
	StudentGrade	(10) Short Text	پایه (در کارگاه‌های بعد اضافه می‌شود)
teach	★ RowID	AutoNumber	ردیف
	CourseCode	Long) Number (Integer	کلید خارجی ← course
	TeacherCode	Long) Number (Integer	کلید خارجی ← teacher
	StudentCode	Long) Number (Integer	کلید خارجی ← student
	EducationYear	(9) Short Text	سال تحصیلی
	Mark	(Single) Number	نمره

جدول teach جدول رابطه است و رابطه‌ی N:M میان هنرآموز، هنرجو و درس را به سه ارتباط N:1 تبدیل می‌کند. معادل SQL (هر دستور را جداگانه در نمای SQL اجرا کنید؛ ابتدا سه جدول اصلی و سپس teach):

ایران توشه
توشه ای برای موفقیت

```
CREATE TABLE teacher (TeacherCode LONG PRIMARY KEY, TeacherName TEXT(20),
  TeacherFamily TEXT(25), TeacherTel TEXT(12))
CREATE TABLE course (CourseCode LONG PRIMARY KEY, CourseName TEXT(40), Unit BYTE)
CREATE TABLE student (StudentCode LONG PRIMARY KEY, StudentName TEXT(20),
  StudentFamily TEXT(25), StudentTel TEXT(12), StudentGrade TEXT(10))
CREATE TABLE teach (RowID COUNTER PRIMARY KEY, CourseCode LONG, TeacherCode LONG,
  StudentCode LONG, EducationYear TEXT(9), Mark SINGLE,
  CONSTRAINT fk1 FOREIGN KEY (CourseCode) REFERENCES course(CourseCode),
  CONSTRAINT fk2 FOREIGN KEY (TeacherCode) REFERENCES teacher(TeacherCode),
  CONSTRAINT fk3 FOREIGN KEY (StudentCode) REFERENCES student(StudentCode))
```

برای برقراری ارتباط: Database Tools ← Relationships، افزودن چهار جدول، کشیدن CourseCode، TeacherCode و StudentCode از جدول‌های اصلی روی فیلد هم‌نام در teach و فعال کردن Enforce Referential Integrity.

۳۷ صفحه ۲۸ فعالیت کارگاهی

پایگاه داده‌ای با نام Store ایجاد کنید و با توجه به موجودیت‌های موجود در فروشگاه، جدول‌های مورد نظر را برای آنها تعریف کرده و ارتباط بین جدول‌ها را برقرار کنید.

موجودیت‌ها: مشتری، کالا، فاکتور؛ ارتباط فاکتور و کالا چندبه‌چند است و با جدول InvoiceItem پیاده‌سازی می‌شود.

جدول‌ها (کلید اصلی با ★)

جدول	فیلدها
Customer	CustomerID ★ (Number)، Name، Family، Tel، Address (همه Short Text)
Product	ProductID ★ (Number)، ProductName (Short Text)، Price (Currency)، Stock (Number)
Invoice	InvoiceID ★ (Number)، CustomerID (ک. خارجی)، InvoiceDate (Date/Time)
InvoiceItem	RowID ★ (AutoNumber)، InvoiceID (ک. خارجی)، ProductID (ک. خارجی)، Quantity (Number)

```
CREATE TABLE Customer (CustomerID LONG PRIMARY KEY, Name TEXT(20), Family TEXT(25),
  Tel TEXT(12), Address TEXT(100))
CREATE TABLE Product (ProductID LONG PRIMARY KEY, ProductName TEXT(50),
  Price CURRENCY, Stock LONG)
CREATE TABLE Invoice (InvoiceID LONG PRIMARY KEY, CustomerID LONG, InvoiceDate DATETIME,
  CONSTRAINT fk_c FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID))
CREATE TABLE InvoiceItem (RowID COUNTER PRIMARY KEY, InvoiceID LONG, ProductID LONG,
  Quantity LONG,
  CONSTRAINT fk_i FOREIGN KEY (InvoiceID) REFERENCES Invoice(InvoiceID),
  CONSTRAINT fk_p FOREIGN KEY (ProductID) REFERENCES Product(ProductID))
```

سپس در پنجره‌ی Relationships هر کلید خارجی را روی کلید اصلی متناظر بکشید و Enforce Referential Integrity را فعال کنید؛ نوع داده‌ی کلید خارجی باید با کلید اصلی یکسان باشد.

۳۸ صفحه ۳۱ **کنجکاوی**

دو راه دیگر برای درج رکورد پیدا کنید.

- استفاده از **فرم** (Create ← Form) و واردکردن اطلاعات در آن.
- **درون‌ریزی داده** از پرونده‌ی دیگر مانند Excel یا متنی با External Data ← New Data Source یا Import و نیز کپی و چسباندن (Paste) رکوردها از Excel به نمای Datasheet.

۳۹ صفحه ۳۴ **فعالیت کارگاهی**

ـ به‌دلخواه یکی از جدول‌های موجود در پایگاه داده را بازکرده و سه رکورد جدید به روش مستقیم از طریق واسط کاربری برنامه در آن ایجاد کنید.

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

جدول TeacherName، 1011 = TeacherCode :teacher = علی، TeacherFamily = رنجبر، TeacherTel = 34558896

جدول StudentName، 156 = StudentCode :student = حامد، StudentFamily = عزیزی، StudentTel = 35429902

روش مستقیم: جدول course را با دابل‌کلیک باز می‌کنیم و در سطر آخر (ستاره) سه رکورد وارد می‌کنیم؛ مثلاً (5563، پایگاه داده، 3)، (5564، برنامه‌سازی، 4) و (5565، شبکه، 2). با کلید Tab به فیلد بعدی می‌رویم.

با دستور SQL: زبانه‌ی Create ← Query Design ← SQL View و پس از نوشتن، دکمه‌ی Run. هر دستور را جداگانه اجرا کنید. (مقدارهای متنی داخل ' ' قرار می‌گیرند؛ نام‌های فارسی را هم می‌توان در همین قالب نوشت.)

```
INSERT INTO teacher (TeacherCode, TeacherName, TeacherFamily, TeacherTel)
VALUES (1011, 'Ali', 'Ranjbar', '34558896')

INSERT INTO student
VALUES (156, 'Hamed', 'Azizi', '35429902', '11')
```

ایران توشه
توشه‌ای برای موفقیت

۴۰ صفحه ۳۴ کارگاه ۲

ـ جدول teach را باز کرده و مقدار 5563 را برای فیلد کد درس، مقدار 1011 را برای فیلد کد کارمندی و مقدار 153 را برای فیلد شماره دانش‌آموزی وارد کنید. پس از ثبت این رکورد چه اتفاقی رخ می‌دهد؟

ـ امکان ورود مقادیر تکراری برای فیلدهای کلید خارجی را بررسی کنید.

ـ امکان خالی بودن فیلدهای کلید خارجی را بررسی نمایید.

- **ثبت رکورد:** چون Referential Integrity فعال است و در جدول‌های اصلی رکوردی با کد درس 5563 یا شماره‌ی دانش‌آموزی 153 نیست، Access رکورد را ذخیره نمی‌کند و پیام خطا می‌دهد: «You cannot add or change a record because a related record is required in table ...». رکورد زمانی ثبت می‌شود که ابتدا درس، هنرآموز و هنرجوی متناظر در جدول‌های اصلی وارد شده باشند.
- **مقدار تکراری:** مجاز است؛ چون فیلد کلید خارجی سمت «چند» ارتباط است (یک هنرآموز یا هنرجو می‌تواند در چندین رکورد teach بیاید). فقط مقدار کلید اصلی نباید تکراری باشد.
- **خالی بودن:** اگر Required = No باشد و فیلد جزو کلید اصلی نباشد، مقدار خالی (Null) پذیرفته می‌شود؛ ولی چنین رکوردی به هیچ رکورد اصلی وصل نیست. با Required = Yes خالی بودن ممنوع می‌شود.

۴۱ صفحه ۳۵ کنجکاوی

اگر از عبارت WHERE در دستور UPDATE صرف‌نظر شود، کدام رکوردهای جدول به‌روزرسانی خواهند شد؟

همه‌ی رکوردهای جدول. بدون WHERE شرطی برای انتخاب رکورد نیست، پس مقدار جدید در فیلد موردنظر همه‌ی رکوردها قرار می‌گیرد. مثلاً UPDATE student SET StudentGrade = '11' پایه‌ی همه‌ی هنرجویان را 11 می‌کند.

ایران توشه
توشه ای برای موفقیت

۴۲ صفحه ۳۵ فعالیت کارگاهی

با استفاده از دستور UPDATE:

ـ شماره تلفن هنرآموزی که در فعالیت‌های قبلی به جدول teacher اضافه شد را از 34558896 به 34557796 تغییر دهید.

ـ شماره دانش‌آموزی مربوط به هنرجویی با نام 'حامد عزیزی‌نیا' را به 53 تغییر دهید.

```
UPDATE teacher
SET TeacherTel = '34557796'
WHERE TeacherCode = 1011

UPDATE student
SET StudentCode = 53
WHERE StudentName = 'Hamed' AND StudentFamily = 'Azizinia'
```

در دستور دوم (نام فارسی را به‌جای 'Hamed' و 'Azizinia' بنویسید) اگر برای هنرجو رکوردی در جدول teach ثبت شده باشد، تغییر کلید اصلی فقط وقتی ممکن است که هنگام ایجاد ارتباط Cascade Update Related Fields را فعال کرده باشیم.

۴۳ صفحه ۳۶ کنجکاوی

چه روش دیگری برای حذف رکورد وجود دارد؟

- در نمای Datasheet روی نشانگر سطر (کادر خاکستری سمت چپ رکورد) راست‌کلیک کرده، Delete Record را انتخاب کنیم.
- رکورد را انتخاب کرده و در زبانه‌ی Home دکمه‌ی Delete را بزنیم.
- ساخت **Delete Query** در نمای Design پرس‌وجو (دستور DELETE) برای حذف چند رکورد با شرط.

۴۴ صفحه ۳۷ فعالیت کارگاهی

با استفاده از دستور DELETE :

ـ رکورد هنرآموزی با کد کارمندی 1011 را حذف کنید.

ـ اطلاعات تمامی هنرجویانی که شماره دانش‌آموزی آنها دو رقمی است را از جدول student حذف کنید.

```
DELETE
FROM teacher
WHERE TeacherCode = 1011

DELETE
FROM student
WHERE StudentCode >= 10 AND StudentCode <= 99
```

شرط دوم را با BETWEEN 10 AND 99 هم می‌توان نوشت. اگر برای این رکوردها در جدول teach رکوردی باشد، فقط هنگامی حذف می‌شوند که Cascade Delete Related Records فعال باشد، وگرنه Access مانع حذف می‌شود.

ایران توشه
توشه‌ای برای آینده موفقیت

۴۵ صفحه ۳۷ کارگاه ۳

همه رکوردهای جدول را حذف کنید و نتیجه را بررسی کنید. آیا جدول حذف می‌شود؟

```
DELETE
FROM student
```

خیر. فقط داده‌ها (رکوردها) حذف می‌شوند و جدول خالی با همه‌ی فیلدها و ویژگی‌هایش باقی می‌ماند و می‌توان دوباره رکورد وارد کرد. برای حذف خود جدول باید از DROP TABLE student یا حذف از پنجره‌ی اشیا استفاده کرد. رکوردهای حذف‌شده قابل بازگشت نیستند؛ پس پیش از حذف نسخه‌ی پشتیبان تهیه کنید.

۴۶ صفحه ۴۰ فعالیت کارگاهی

با استفاده از wizard، یک پرس‌وجوی ساده ایجاد کنید به‌طوری‌که نام، نام خانوادگی و شماره تلفن هنرآموزان را نمایش دهد.

1. Create ← Query Wizard ← Simple Query Wizard ← OK.
2. از Tables/Queries جدول teacher را انتخاب کنید.
3. فیلدهای TeacherName، TeacherFamily و TeacherTel را با دکمه‌ی > به Selected Fields ببرید و Next را بزنید.
4. گزینه‌ی Detail را نگه دارید، نام پرس‌وجو را (مثلاً Teacher List) بنویسید و Finish کنید.

معادل SQL این پرس‌وجو:

```
SELECT TeacherName, TeacherFamily, TeacherTel
FROM teacher
```

ایران توشه
توشه ای برای موفقیت

۴۷ صفحه ۴۲ فعالیت کارگاهی

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

```
SELECT DISTINCT student.StudentCode, StudentName + ' ' + StudentFamily AS Student
FROM student, teach
WHERE teach.StudentCode = student.StudentCode
```

جدول teach فقط شماره‌ی هنرجویانی را دارد که درس گرفته‌اند؛ DISTINCT از تکرار هنرجو (به‌ازای هر درس) جلوگیری می‌کند. نمونه‌ی خروجی:

Student	StudentCode
Hamed Azizi	156
Reza Moradi	157
Mina Ahmadi	158

۴۸ صفحه ۴۳ فعالیت کارگاهی

مثال بالا را به‌گونه‌ای تغییر دهید که خروجی به‌صورت نزولی (DESC) مرتب شوند. سپس نتایج حاصل را با نتایج مثال قبل مقایسه کنید.

```
SELECT StudentCode, StudentName, StudentFamily
FROM student
ORDER BY StudentFamily DESC, StudentName DESC
```

مقایسه: در مثال قبل نام‌خانوادگی‌ها از A تا Z (صعودی، مثلاً Ahmadi، Azizi، Karimi، Moradi) و در این مثال برعکس، از Z به A (Moradi، Karimi، Azizi، Ahmadi) نمایش داده می‌شوند. اگر نام‌خانوادگی دو هنرجو یکسان باشد، ترتیب بر اساس نام تعیین می‌شود (در حالت نزولی نیز نام‌ها از آخر به اول).

۴۹ صفحه ۴۳ کارگاه ۶

جدول مورد استفاده در پرس‌وجو را باز کنید. آیا اطلاعات آن مرتب‌شده است؟

خیر. عبارت ORDER BY فقط ترتیب نمایش رکوردها در نتیجه‌ی همان پرس‌وجو را عوض می‌کند و ترتیب رکوردها در خود جدول (معمولاً بر اساس کلید اصلی) ثابت می‌ماند. برای مرتب کردن جدول باید از واسط کاربری Access (دکمه‌های Ascending و Descending) استفاده کرد.

۵۰ صفحه ۴۴ فعالیت کارگاهی

ـ با استفاده از واسط کاربری برنامه Access، جدول course را براساس کد درس به‌صورت نزولی مرتب کنید.

ـ با استفاده از دستورات SQL و عبارت ORDER BY، جدول course را براساس نام دروس و سپس تعداد واحد آنها به‌صورت صعودی مرتب کنید.

1. جدول course را در نمای Datasheet باز کنید.
2. روی عنوان ستون CourseCode کلیک کنید تا انتخاب شود.

3. در زبانه‌ی Home، گروه Sort & Filter، روی Descending کلیک کنید و جدول را ذخیره کنید.

با SQL:

```
SELECT *
FROM course
ORDER BY CourseName ASC, Unit ASC
```

۵۱ صفحه ۴۴ کارگاه ۷

با استفاده از دستورات SQL، یک پرس‌وجو شامل نام خانوادگی، تعداد واحد دروس اخذ شده و پایه تحصیلی هنرجو ایجاد کنید.

برای این منظور چه فیلدهایی را باید در مقابل دستور SELECT، قرار دهیم؟ SELECT ..... , ... , ...

این فیلدها متعلق به کدام جدول‌ها هستند؟ teach , ... , ... FROM

اگر بخواهیم تنها هنرجویانی نمایش داده شوند که برای آنها انتخاب واحد انجام شده است، به کدام جدول باید مراجعه کنیم؟

برای ایجاد شرط مناسب، عبارت زیر را کامل کنید. .... = student.StudentCode

```
SELECT StudentFamily, Unit, StudentGrade
FROM student, course, teach
```

- فیلدهای SELECT: StudentFamily، Unit و StudentGrade.
- StudentFamily و StudentGrade متعلق به جدول student و Unit متعلق به جدول course است؛ پس در FROM: student، course و teach.
- برای هنرجویانی که انتخاب واحد کرده‌اند باید به جدول teach مراجعه کنیم، چون رکورد هر درس‌گرفتن در آن ثبت می‌شود.
- شرط: teach.StudentCode = student.StudentCode

ایران توشه
توشه‌ای برای موفقیت

۵۲ صفحه ۴۵ کارگاه ۷

حال اگر بخواهیم تنها واحدهای دروسی که به‌وسیله‌ی هنرجویان اخذ شده‌اند را در نظر بگیریم، به کدام جدول باید مراجعه کنیم؟

آیا این جدول در مقابل FROM وجود دارد؟ برای ایجاد شرط مناسب، عبارت زیر را کامل کنید. ..... .

..... = course.CourseCode

اکنون دو شرط داریم که باید آنها را با یکدیگر ترکیب کرده و در مقابل دستور WHERE قرار دهیم. این دو شرط باید هم‌زمان بررسی شوند، در نتیجه با استفاده از عملگر AND آنها را با یکدیگر ترکیب کرده، دستور WHERE را تکمیل کنید. WHERE .....

به نظر شما چرا جدول teach را مقابل FROM قرار دادیم؟

- جدول course (چون Unit در آن است). بله، course در FROM آمده است.
- شرط دوم: teach.CourseCode = course.CourseCode

```
SELECT StudentFamily, Unit, StudentGrade
FROM student, course, teach
WHERE teach.StudentCode = student.StudentCode AND
      teach.CourseCode = course.CourseCode
GROUP BY StudentGrade, StudentFamily, Unit
```

چرا teach؟ جدول teach جدول رابطه است و کلید خارجی StudentCode و CourseCode را دارد؛ فقط از طریق آن می‌توان هنرجو را به درس‌هایش وصل کرد. بدون آن، ارتباطی میان student و course نیست و نتیجه‌ی پرس‌وجو بی‌معنا (ضرب دکارتی) می‌شود. هر فیلدی که در SELECT است و تابع تجمعی نیست، باید در GROUP BY هم بیاید.

۵۳ صفحه ۴۷ **فعالیت کارگاهی**

دستور SELECT را به‌گونه‌ای بنویسید که نام و نام‌خانوادگی هنرجویان را براساس پایه تحصیلی و نام درس گروه‌بندی کرده به‌صورت توأم نمایش دهد.

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

نمونه‌ی خروجی

StudentFamily	StudentName	CourseName	StudentGrade
Ahmadi	Mina	Database	10
Ahmadi	Mina	Programming	10
Azizi	Hamed	Database	11
Moradi	Reza	Database	11

رکوردها ابتدا بر اساس پایه، سپس بر اساس نام درس و بعد نام هنرجو گروه‌بندی می‌شوند.

۵۴ صفحه ۵۰ کارگاه ۸

چگونه می‌توان فیلدهای سایر جدول‌ها یا پرس‌وجوها را به بخش فیلدهای انتخاب‌شده اضافه کرد؟

در همان مرحله‌ی انتخاب فیلدها، از فهرست کشویی Tables/Queries جدول یا پرس‌وجوی دیگر را انتخاب می‌کنیم؛ فیلدهای آن در Available Fields ظاهر می‌شود و با دکمه‌ی > یا >> به Selected Fields می‌رود. این کار را برای هر تعداد جدول می‌توان تکرار کرد (به شرط وجود ارتباط بین جدول‌ها). راه ساده‌تر این است که ابتدا پرس‌وجویی با همه‌ی فیلدهای لازم بسازیم و گزارش را روی آن بنویسیم.

۵۵ صفحه ۵۱ فعالیت کارگاهی

ـ با استفاده از wizard، گزارشی از فهرست هنرآموزان تهیه کنید.

ـ گزارشی از ثبت‌نام دانش‌آموزان به همراه تعداد واحدهای اخذ شده به‌وسیله‌ی هرکدام از آنها را که در مثال‌های قبل با استفاده از دستورات SQL انجام شد، تهیه کنید.

گزارش فهرست هنرآموزان: Create ← Report Wizard ← جدول teacher ← انتقال فیلدها با >> ← بدون گروه‌بندی ← مرتب‌سازی بر اساس TeacherFamily ← انتخاب Layout و جهت صفحه ← نام Teacher Report ← Finish.

گزارش واحدها: ابتدا پرس‌وجوی زیر را بسازید و با نام qryUnits ذخیره کنید:

```
SELECT StudentFamily, SUM(Unit) AS [Total Units], StudentGrade
FROM teach, student, course
WHERE teach.StudentCode = student.StudentCode AND
      teach.CourseCode = course.CourseCode
GROUP BY StudentGrade, StudentFamily
```

سپس Report Wizard را اجرا کنید، در Tables/Queries پرس‌وجوی qryUnits را انتخاب و همه‌ی فیلدها را منتقل کنید، گروه‌بندی را بر اساس StudentGrade بگذارید، مرتب‌سازی را بر اساس StudentFamily صعودی کنید و گزارش را با نام Units Report ذخیره کنید.

۵۶ صفحه ۵۳ فعالیت کارگاهی

گزارش ثبت‌نام دانش‌آموزان را که در فعالیت‌های قبل ایجاد کردید، در نمای Design بازکرده و موارد زیر را به آن اضافه کنید:

ـ شماره صفحه در سمت چپ پاصفحه قرارگرفته و تعداد کل صفحات را نیز نشان دهد.

ـ تاریخ و زمان را به گزارش اضافه کنید.

ـ لوگوی هنرستان را در گزارش قرار دهید.

ـ با استفاده از توابع تجمعی، تعداد رکوردها را در پاصفحه شمارش کنید.

ـ عنوان گزارش را با قلم و رنگ دلخواه در سر صفحه گزارش مشخص کنید.

روی برگه‌ی گزارش راست‌کلیک کنید و Design View را بزنید، سپس:

- **شماره‌ی صفحه:** Design ← Page Numbers؛ Format = Page N of M، Position = Bottom of Page (Footer)، Alignment = Left و OK. (معادل عبارت ="Page " & [Page] & " of " & [Pages] در یک Text Box پاصفحه.)
- **تاریخ و زمان:** Date and Time؛ گزینه‌های Include Date و Include Time را با قالب دلخواه فعال کنید.
- **لوگو:** Logo و انتخاب پرونده‌ی تصویر لوگوی هنرستان؛ در Report Header قرار می‌گیرد و با کشیدن جابه‌جا می‌شود.
- **شمارش رکوردها:** روی بخش Page Footer کلیک کنید، Totals ← Count Records (یا Text Box با مقدار =Count(*)).
- **عنوان:** روی برچسب عنوان در Report Header کلیک کنید و از زبانه‌ی Format نوع قلم، اندازه و رنگ (Font Color) را تعیین کنید.

۵۷ صفحه ۵۴ پروژه فاز ۱

می‌خواهیم برای کتابخانه‌ی یک هنرستان، پایگاه داده ایجاد کنیم. این پایگاه شامل موجودیت‌های عضو و کتاب است. با در نظر گرفتن این محیط عملیاتی موارد زیر را به ترتیب انجام دهید:

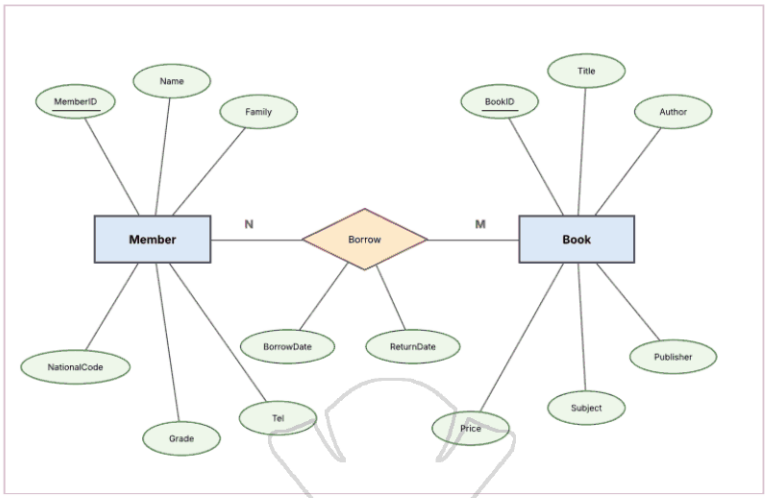
۱ـ موجودیت‌ها را شناسایی کرده، صفات و روابط بین آنها را بررسی کرده، نمودار ER آنها را به‌طور کامل رسم کنید.

موجودیت — صفات — کلید — ارتباط

موجودیت	صفات	کلید	ارتباط
عضو (Member)	MemberID، Name، Family، NationalCode، Grade، Tel	MemberID	امانت (Borrow) با کتاب: N:M
کتاب (Book)	BookID، Title، Author، Publisher، Subject، Price	BookID	امانت (Borrow) با عضو: N:M
امانت (Borrow)، صفت‌های ارتباط	BorrowDate (تاریخ امانت)، ReturnDate (تاریخ بازگشت؛ تا برگشت خالی)	BorrowID (در جدول)	کلید خارجی: MemberID و BookID

ایران توشه
توشه‌ای برای موفقیت

یک عضو می‌تواند چند کتاب را امانت بگیرد و یک کتاب در زمان‌های مختلف به چند عضو امانت داده می‌شود؛ پس ارتباط N:M است و در پیاده‌سازی به جدول سوم (امانت) تبدیل می‌شود. در نمودار، بیضی‌ها صفات (کلید با زیرخط)، مستطیل‌ها موجودیت‌ها و لوزی ارتباط را نشان می‌دهند.



۵۸ صفحه ۵۴ پروژه فاز ۱

۲ـ پایگاه داده را در Access ایجاد کنید.

۳ـ بر اساس نمودار ER، جدول‌های مورد نیاز را به همراه فیلدهای آنها ایجاد کنید و فیلد کلید اصلی و سایر جزئیات مربوط به فیلدها را در هر جدول تعیین کنید.

۴ـ ارتباط بین جدول‌ها را برقرار کنید.

ایجاد پایگاه داده: Blank desktop database با نام Library.accdb. سپس جدول‌ها را در نمای Design می‌سازیم:

ساختار جدول‌ها (کلید اصلی با ★)

ایران توشه
توشه ای برای موفقیت

جدول	فیلد	نوع و ویژگی
member	★ MemberID	Number (Long Integer)، Caption: کد عضویت
	Name / Family	Short Text (20 / 25)، Required = Yes
	NationalCode	Short Text (10)، Input Mask = 0000000000، Indexed، بدون تکرار
	Grade / Tel	Short Text (10 / 12)
book	★ BookID	Number (Long Integer)
	Title	Short Text (80)، Required = Yes
	/ Author / Publisher Subject	Short Text (40 / 40 / 30)
	Price	Currency
borrow	★ BorrowID	AutoNumber
	MemberID	Number (Long Integer)، کلید خارجی ← member
	BookID	Number (Long Integer)، کلید خارجی ← book
	BorrowDate	Date/Time، Default Value = Date()
	ReturnDate	Date/Time، Required = No (تا بازگشت کتاب خالی)

```
CREATE TABLE member (MemberID LONG PRIMARY KEY, Name TEXT(20) NOT NULL,
  Family TEXT(25) NOT NULL, NationalCode TEXT(10), Grade TEXT(10), Tel TEXT(12))
CREATE TABLE book (BookID LONG PRIMARY KEY, Title TEXT(80) NOT NULL, Author TEXT(40),
  Publisher TEXT(40), Subject TEXT(30), Price CURRENCY)
CREATE TABLE borrow (BorrowID COUNTER PRIMARY KEY, MemberID LONG NOT NULL,
  BookID LONG NOT NULL, BorrowDate DATETIME, ReturnDate DATETIME,
  CONSTRAINT fk_m FOREIGN KEY (MemberID) REFERENCES member(MemberID),
  CONSTRAINT fk_b FOREIGN KEY (BookID) REFERENCES book(BookID))
```

ارتباط‌ها: Database Tools ← Relationships؛ جدول‌ها را اضافه کنید، member.MemberID را روی borrow.MemberID و book.BookID را روی borrow.BookID بکشید و Enforce Referential Integrity را فعال کنید (ارتباط 1:N از هر جدول اصلی به borrow)؛ این دو ارتباط با هم همان ارتباط N:M عضو و کتاب را می‌سازند.

۵۹ صفحه ۵۴ پروژه فاز ۱

۵ـ داده‌ها را در جدول‌ها وارد کنید.

ابتدا جدول‌های اصلی (member و book) و سپس borrow را پر می‌کنیم (وگرنه کنترل یکپارچگی اجازه‌ی ثبت نمی‌دهد). می‌توان در نمای Datasheet تایپ کرد یا از INSERT INTO استفاده کرد؛ بازگشت‌نداده‌ها با ReturnDate خالی ثبت می‌شوند:

```
INSERT INTO member VALUES (1001, 'Ali', 'Ahmadi', '0012345678', '11', '09121112233')
INSERT INTO member VALUES (1002, 'Sara', 'Karimi', '0023456789', '10', '09123334455')
INSERT INTO member VALUES (1003, 'Reza', 'Moradi', '0034567890', '12', '09125556677')
INSERT INTO member VALUES (1004, 'Mina', 'Rezaei', '0045678901', '11', '09127778899')

INSERT INTO book VALUES (1, 'C# Programming', 'Alavi', 'Fanni', 'Programming', 95000)
INSERT INTO book VALUES (2, 'Access 2016', 'Hosseini', 'Fanni', 'Database', 80000)
INSERT INTO book VALUES (3, 'Computer Networks', 'Tanenbaum', 'Ney', 'Network', 120000)
INSERT INTO book VALUES (5, 'Python Programming', 'Rahimi', 'Elm', 'Programming', 85000)

INSERT INTO borrow (MemberID, BookID, BorrowDate, ReturnDate)
VALUES (1001, 1, #2025-10-01#, #2025-10-15#)
INSERT INTO borrow (MemberID, BookID, BorrowDate)
VALUES (1001, 2, #2025-11-02#)
INSERT INTO borrow (MemberID, BookID, BorrowDate)
VALUES (1002, 5, #2025-11-05#)
INSERT INTO borrow (MemberID, BookID, BorrowDate, ReturnDate)
VALUES (1003, 3, #2025-10-20#, #2025-11-01#)
INSERT INTO borrow (MemberID, BookID, BorrowDate, ReturnDate)
VALUES (1002, 1, #2025-09-10#, #2025-09-20#)
```

۶۰ صفحه ۵۴ پروژه فاز ۱

۶ـ برای موارد خواسته شده پرس‌وجو ایجاد کرده، ذخیره کنید:

فهرست اعضا

فهرست کتاب‌ها

فهرست کتاب‌هایی با موضوع برنامه نویسی

فهرست اعضایی که کتابی را به امانت گرفته‌اند.

هر دستور را در نمای SQL یک Query Design جدید بنویسید، با Run اجرا کنید و با نام‌های qryMembers، qryBooks، qryProgBooks و qryBorrowers ذخیره کنید.

فهرست اعضا

ایران توشه
توشه ای برای موفقیت

```
SELECT MemberID, Name, Family, Grade, Tel
FROM member
ORDER BY Family, Name
```

فهرست کتاب‌ها

```
SELECT BookID, Title, Author, Publisher, Subject, Price
FROM book
ORDER BY Title
```

کتاب‌هایی با موضوع برنامه‌نویسی

```
SELECT BookID, Title, Author
FROM book
WHERE Subject = 'Programming'
```

اعضایی که کتاب امانت گرفته‌اند

```
SELECT DISTINCT member.MemberID, Name, Family
FROM member, borrow
WHERE member.MemberID = borrow.MemberID
```

نتیجه‌ی پرس‌وجوی آخر با داده‌های نمونه: (1001، Ali Ahmadi)، (1002، Sara Karimi)، (1003، Reza Moradi). عضو 1004 هرگز امانت نگرفته و در نتیجه نیست. DISTINCT از تکرار عضوی که چند کتاب گرفته جلوگیری می‌کند.

۶۱ صفحه ۵۴ پروژه فاز ۱

فهرست اعضایی که هنوز کتاب امانت گرفته شده را برنگردانده‌اند.

فهرست اعضا و تعداد کتاب‌هایی که تاکنون امانت گرفته‌اند.

فهرست کتاب‌های امانت گرفته شده به همراه اسامی کسانی که کتاب را امانت گرفته‌اند.

ایران توشه
توشه‌ای برای موفقیت

اعضایی که کتاب را برنگردانده‌اند

```
SELECT DISTINCT member.MemberID, Name, Family
FROM member, borrow
WHERE member.MemberID = borrow.MemberID AND ReturnDate IS NULL
```

نمونه‌ی خروجی: Ali Ahmadi و Sara Karimi (کتاب‌هایشان هنوز بازنگشته است).

اعضا و تعداد کتاب‌های امانت‌گرفته

```
SELECT member.MemberID, Name, Family, COUNT(borrow.BookID) AS [Borrow Count]
FROM member, borrow
WHERE member.MemberID = borrow.MemberID
GROUP BY member.MemberID, Name, Family
```

نمونه‌ی خروجی

Borrow Count	Family	Name	MemberID
2	Ahmadi	Ali	1001
2	Karimi	Sara	1002
1	Moradi	Reza	1003

اگر بخواهیم اعضای بدون امانت نیز با عدد 0 بیایند، به‌جای FROM ... WHERE از FROM member LEFT JOIN borrow ON member.MemberID = borrow.MemberID استفاده می‌کنیم (عضو 1004 با تعداد 0 ظاهر می‌شود).

کتاب‌های امانت‌گرفته‌شده به همراه نام امانت‌گیرنده

```
SELECT Title, Name + ' ' + Family AS Member, BorrowDate
FROM book, borrow, member
WHERE borrow.BookID = book.BookID AND
      borrow.MemberID = member.MemberID
ORDER BY Title
```

نمونه‌ی خروجی

BorrowDate	Member	Title
2025-11-02	Ali Ahmadi	Access 2016
2025-10-01	Ali Ahmadi	C# Programming
2025-09-10	Sara Karimi	C# Programming
2025-10-20	Reza Moradi	Computer Networks
2025-11-05	Sara Karimi	Python Programming

ایران توشه
توشه ای برای موفقیت

۷ـ برای هر یک از پرس‌وجوهای مرحله‌ی قبل گزارشی تهیه و تنظیم کنید به گونه‌ای که در بالای هر گزارش نام کتابخانه به همراه لوگوی آن، تاریخ و ساعت گزارش قرار داشته باشد و شماره صفحه و مجموعه رکوردها در انتهای هر گزارش محاسبه شده باشد.

گزارش‌ها

گزارش	منبع	گروه‌بندی / مرتب‌سازی
Members Report	qryMembers	مرتب بر اساس Family
Books Report	qryBooks	گروه‌بندی بر اساس Subject
Programming Books	qryProgBooks	مرتب بر اساس Title
Borrowers Report	qryBorrowers	مرتب بر اساس Family
Not Returned Report	پرس‌وجوی ۵	مرتب بر اساس Family
Borrow Count Report	پرس‌وجوی ۶	مرتب بر اساس Borrow Count (نزولی)
Borrowed Books Report	پرس‌وجوی ۷	گروه‌بندی بر اساس Member

برای هر گزارش مراحل زیر را انجام می‌دهیم:

1. Create ← Report Wizard؛ در Tables/Queries پرس‌وجوی مربوط را انتخاب، فیلدها را منتقل، گروه‌بندی و مرتب‌سازی را تعیین، Layout را انتخاب و Finish کنید.
2. گزارش را در نمای Design باز کنید.
3. در Report Header: Logo (تصویر لوگوی کتابخانه)، Title (نام کتابخانه، مثلاً «کتابخانه‌ی هنرستان ...») و Date and Time را درج کنید.
4. در Page Footer: Page Numbers با قالب Page N of M را درج کنید.
5. رکوردها را بشمارید: روی Report Footer (یا Page Footer) کلیک کنید و Totals ← Count Records را بزنید (یا Text Box با مقدار =Count(*) و برچسب «تعداد رکوردها»).
6. با Print Preview نتیجه را ببینید و گزارش را ذخیره کنید.

برای اینکه نام و لوگو در همه‌ی گزارش‌ها یکسان باشد، یک گزارش را کامل کرده، با Save As کپی می‌کنیم و فقط Record Source آن را به پرس‌وجوی دیگر تغییر می‌دهیم.

ایران توشه
توشه‌ای برای موفقیت

گام به گام توسعه برنامه‌سازی و پایگاه داده - پودمان دوم

مدیریت مجموعه داده

فهرست

۶۳ صفحه ۵۹ · فعالیت گروهی — حلقه‌ی معین و نامعین ۴۸

۶۴ صفحه ۶۰ · فعالیت کارگاهی — تکمیل روندنمای ثبت فعالیت هنرجویان ۴۸

۶۵ صفحه ۶۱ · فعالیت کارگاهی — نوبت‌دهی مطب دندان‌پزشکی ۴۹

۶۶ صفحه ۶۲ · فعالیت کارگاهی — شمارش قبول‌شده‌ها ۴۹

۶۷ صفحه ۶۴ · کارگاه ۱ — تبدیل روندنما به برنامه ۵۰

۶۸ صفحه ۶۴ · کارگاه ۲ — بررسی شرایط حلقه‌ی تکرار معین ۵۰

۶۹ صفحه ۶۵ · کنجکاوی — نوع متغیر حلقه ۵۲

۷۰ صفحه ۶۵ · فعالیت گروهی — پیش‌بینی خروجی حلقه‌ها ۵۲

۷۱ صفحه ۶۵ · فعالیت منزل — گروه‌بندی هنرجویان ۵۲

۷۲ صفحه ۶۵ · کارگاه ۳ — کار با متغیر خارج از بلاک (بازی هپ) ۵۳

۷۳ صفحه ۶۶ · کارگاه ۴ — محاسبه‌ی مجموع زمان مطالعه ۵۳

۷۴ صفحه ۶۷ · کارگاه ۵ — خروج زودرس از حلقه (حدس ماه تولد) ۵۴

ایران توشه
توشه‌ای برای موفقیت

۷۵ صفحه ۶۸ · کارگاه ۵ — Trace و اصلاح برنامه‌ی حدس ماه ۵۵

۷۶ صفحه ۶۸ · فعالیت گروهی — پیش‌بینی خروجی حلقه‌های for ۵۶

۷۷ صفحه ۶۹ · فعالیت منزل — مسئله‌ی کاربردی برای حلقه‌ی گام ۳ ..... ۵۷

۷۸ صفحه ۶۹ · فعالیت کارگاهی — اجرای گام‌به‌گام MonthGuess و پنجره‌ی Local ۵۷

۷۹ صفحه ۶۹ · فعالیت منزل — شمارش هنرجویان هم‌نام حضرت معصومه (س) ۵۸

۸۰ صفحه ۷۰ · فعالیت گروهی — مثال‌هایی از حلقه‌ی نامعین ۵۹

۸۱ صفحه ۷۰ · کارگاه ۶ — به‌کارگیری حلقه‌ی while (میانگین نمرات) ۵۹

۸۲ صفحه ۷۱ · فعالیت کارگاهی — Trace حلقه‌های while ۶۰

۸۳ صفحه ۷۲ · کارگاه ۷ — یافتن بزرگ‌ترین مقدار (روندنما) ۶۱

۸۴ صفحه ۷۳ · کارگاه ۷ — برنامه‌ی Max ۶۱

۸۵ صفحه ۷۳ · فعالیت کارگاهی — ماشین‌حساب کنسولی ۶۳

۸۶ صفحه ۷۴ · فعالیت گروهی — توصیف شکل حلقه‌های متداخل ۶۳

۸۷ صفحه ۷۵ · کارگاه ۸ — جدول‌ضرب ۵ در ۵ با حلقه‌ی متداخل ۶۴

۸۸ صفحه ۷۵ · فعالیت گروهی — پرداخت ۵۰ هزار تومان در خودپرداز ۶۵

۸۹ صفحه ۷۵ · کارگاه ۹ — توسعه‌ی نوبت‌دهی مطب دندان‌پزشکی ۶۵

۹۰ صفحه ۷۶ · فعالیت منزل — توسعه‌ی برنامه‌ی ATM ۶۶

۹۱ صفحه ۷۶ · فعالیت منزل — توسعه‌ی برنامه‌ی میانگین ۶۶

۹۲ صفحه ۷۹ · فعالیت کارگاهی — نمایش اسامی پنج هنرجو از آخر به اول ۶۷

ایران توشه
توشه‌ای برای موفقیت

۹۳ صفحه ۸۰ · فعالیت کارگاهی — اعلان آرایه و بررسی محتوای عناصر ۶۸

۹۴ صفحه ۸۱ · فعالیت کارگاهی — اعلان آرایه‌ها ۶۸

۹۵ صفحه ۸۱ · کنجکاوی — چرا اندیس ۲ برای خانه‌ی سوم ۶۹

۹۶ صفحه ۸۲ · کارگاه ۱ — تعریف و مقداردهی آرایه (Computer) ۶۹

۹۷ صفحه ۸۳ · کارگاه ۱ — روش‌های مقداردهی در هنگام اعلان ۷۰

۹۸ صفحه ۸۵ · کنجکاوی — نمایش عناصر آرایه ۷۰

۹۹ صفحه ۸۵ · کارگاه ۲ — نمایش نام روز هفته (Week) ۷۰

۱۰۰ صفحه ۸۶ · فعالیت کارگاهی — پیمایش آرایه‌ی week ۷۱

۱۰۱ صفحه ۸۷ · کارگاه ۳ — قیمت کل قطعات رایانه (PriceOfComputer) ۷۱

۱۰۲ صفحه ۸۸ · کارگاه ۴ — پیمایش نویسه‌های رشته (StringChar) ۷۲

۱۰۳ صفحه ۸۹ · کارگاه ۵ — ویرایش عناصر آرایه (ValueAddedTax) ۷۲

۱۰۴ صفحه ۸۹ · کنجکاوی — متغیر حلقه‌ی foreach ۷۳

۱۰۵ صفحه ۸۹ · فعالیت منزل — شمارش هنرجویان با سطح شایستگی ۱ ۷۴

۱۰۶ صفحه ۹۰ · فعالیت منزل — هزینه‌ی روزانه‌ی یک ماه ۷۴

۱۰۷ صفحه ۹۱ · فعالیت گروهی — جست‌وجوی خطی یا دودویی ۷۵

۱۰۸ صفحه ۹۱ · فعالیت گروهی — تعداد مقایسه در جست‌وجوی خطی (۱۰۰ هنرجو) ۷۵

۱۰۹ صفحه ۹۲ · کارگاه ۶ — پیاده‌سازی جست‌وجوی خطی (LinearSearch) ۷۵

۱۱۰ صفحه ۹۳ · فعالیت منزل — نمره‌ی ۲۰ هنرجویان ۷۷

ایران توشه
توشه ای برای موفقیت

۱۱۱ صفحه ۹۳ · فعالیت گروهی — بازی حدس عدد (جست‌وجوی دودویی) ۷۷

۱۱۲ صفحه ۹۳ · فعالیت منزل — تکمیل جدول جست‌وجوی دودویی با عدد ۱۲ ..... ۷۸

۱۱۳ صفحه ۹۴ · کارگاه ۷ — متدهای کلاس Array (ArrayClass) ۷۸

۱۱۴ صفحه ۹۵ · فعالیت منزل — برنامه‌ی منودار عملیات آرایه ۷۹

۱۱۵ صفحه ۹۶ · پروژه فاز ۲ — برنامه‌ی کتابخانه‌ی هنرستان ۸۰

ایران توشه
توشه ای برای موفقیت

۶۳ صفحه ۵۹ فعالیت گروهی

به مثال‌های زیر توجه کرده و با کمک هنرآموز خود، مشخص کنید کدام‌یک از آنها معین یا نامعین هستند.

نوع حلقه

مثال	نوع حلقه (معین یا نامعین)
یک نجار روزانه ۱۲ صندلی می‌سازد.	معین (تعداد تکرار ۱۲ است)
تا زمانی که دبیر ورزش سوت پایان را بزند همه‌ی هنرجویان چندین نرمش کششی انجام می‌دهند.	نامعین (زمان سوت معلوم نیست)
در یک نانوایی بربری، نانوا در هر تنور ۶۰ عدد نان می‌پزد.	معین (۶۰ بار)
یک نقاش ساختمان در یک روز تعدادی دیوار با ابعاد مختلف را رنگ می‌زند.	نامعین (تعداد دیوارها از پیش معلوم نیست)

۶۴ صفحه ۶۰ فعالیت کارگاهی

روندنمای بالا را با توجه به الگوریتم برنامه تکمیل کنید.

روندنما بر اساس مراحل الگوریتم به‌صورت زیر کامل می‌شود. بدنه‌ی حلقه (مراحل ۳ تا ۷) همان بلوک‌های «دریافت فعالیت»، «بررسی»، «ثبت گزارش» و «تحویل» به‌همراه افزایش i است.

```
Start
  |
  v
 i <- 1
  |
  v
<i <= 22 ?> --False--> End
  | True
  v
[Receive student i's activity]
[Check the activity]
[Record the report in the evaluation book]
[Return the activity to the student]
[i <- i + 1]
  |
  +--> back to the condition (i <= 22)
```

قسمت‌های پررنگ‌شونده‌ی حلقه: شرط i <= 22، چهار بلوک بدنه و بلوک i ← i + 1 همراه با پیکان بازگشت.

ایران توشه
توشه ای برای موفقیت

۶۵ صفحه ۶۱ فعالیت کارگاهی

ـ اگر تعداد بیماران ۳۰ نفر باشد، چه تغییری لازم است؟

ـ روندنما و الگوریتم را تغییر دهید تا تعداد بیماران را از ورودی دریافت کند.

با توجه به روندنما الگوریتم زیر را کامل کنید. (مراحل ۲ تا ۴)

تکمیل الگوریتم

1. اگر i <= 20 است برو به مرحله‌ی بعد، وگرنه برو به مرحله‌ی ۵.
2. مقدار i را نمایش بده (شماره‌ی نوبت).
3. به i یک واحد اضافه کن و به مرحله‌ی ۲ برو.

۳۰ بیمار

فقط شرط حلقه عوض می‌شود: به‌جای i <= 20 می‌نویسیم i <= 30 (در الگوریتم نیز در مرحله‌ی ۲).

دریافت تعداد بیماران از ورودی

یک متغیر n تعریف می‌کنیم و بلافاصله پس از شروع، مقدار آن را دریافت می‌کنیم؛ شرط حلقه به i <= n تغییر می‌کند.

1. شروع
2. مقدار n (تعداد بیماران) را دریافت کن.
3. مقدار i را برابر یک قرار بده.
4. اگر i <= n است برو به مرحله‌ی بعد، وگرنه برو به مرحله‌ی ۷.
5. i را نمایش بده.
6. به i یک واحد اضافه کن و به مرحله‌ی ۴ برو.
7. پایان

ایران توشه
توشه ای برای موفقیت

۶۶ صفحه ۶۲ فعالیت کارگاهی

مسئله: شرط قبولی در یک آزمون کسب حداقل نمره‌ی ۱۲ از ۲۰ است. می‌خواهیم نمرات هنرجویان یک کلاس ۲۰ نفره را دریافت کرده، تعداد قبول‌شده‌ها را مشخص کنیم.

الگوریتم را کامل کنید (مراحل ۳، ۴ و ۶).

1. شروع
2. در شمارنده (count) مقدار صفر قرار بده.
3. در i مقدار یک قرار بده.

4. اگر i <= 20 است برو به مرحله‌ی ۴ وگرنه برو به مرحله‌ی ۸. (مرحله‌ی ۳)
5. نمره‌ی هنرجوی nام را دریافت کن (mark). (مرحله‌ی ۴)
6. اگر نمره بزرگ‌تر یا مساوی ۱۲ بود، به شمارنده یکی اضافه کن.
7. به i یک واحد اضافه کن. (مرحله‌ی ۶)
8. برو به مرحله‌ی ۳.
9. شمارنده را نمایش بده.
10. پایان

در متن کتاب شماره‌ی مرحله‌ها هنگام استخراج جابه‌جا شده است؛ ترتیب بالا بر پایه‌ی روندنما نوشته شده است.

---

۶۷ صفحه ۶۴ کارگاه ۱

برنامه را طوری تغییر دهید که برای یک درمانگاه خیریه با تعداد بیماران بیشتر قابل استفاده باشد.

شماره‌های مورد نیاز از ۱۰۰ تا ۹۹۹ است.

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

مرحله‌ی ۳: شماره‌ها از ۱۰۰ تا ۹۹۹

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

مرحله‌ی ۴: دریافت حداکثر شماره از ورودی

```
Console.Write("Max patient number: ");
int max = int.Parse(Console.ReadLine());
for (int i = 100; i <= max; i++)
    Console.Write(i + " ");
```

نمونه‌ی اجرا (ورودی ۱۱۰):

```
Max patient number: 110
100 101 102 103 104 105 106 107 108 109 110
```

ایران توشه
توشه ای برای موفقیت

---

۶۸ صفحه ۶۴ کارگاه ۲

۳ پس از پرانتز دستور for علامت ; قرار داده، خروجی را بررسی کنید. نتیجه را برای هم‌گروهی و هنرآموز خود توضیح دهید.

۴ قطعه کد مرحله‌ی ۲ را به‌صورت زیر تغییر داده، نتیجه را با مرحله‌ی قبل مقایسه کنید.

```
for (int i = 2; i <= 30; i++)
{
```

```
    i++;
    Console.Write("{0,5}", i);
```

{

۵ بررسی کنید در صورتی که آکولادهای قطعه کد بالا حذف شوند خروجی چه تغییری می‌کند؟

۶ قطعه کد مرحله‌ی ۲ را طوری تغییر دهید تا شماره‌ی هنرجویانی که باید کارگاه را نظافت کنند چاپ کند.

۷ در عبارت منطقی، علامت => را به =< تبدیل کنید، خروجی چه تغییری می‌کند؟

کد مرحله‌ی ۲ (شماره‌های زوج، مخصوص نظافت کلاس) این است: for (int i = 2; i <= 30; i += 2) Console.Write("{0,5}", i); و خروجی آن ۲، ۴، ...، ۳۰ است.

مرحله‌ی ۳: ; بعد از for

```
for (int i = 2; i <= 30; i += 2) ;
Console.Write("{0,5}", i);   // error CS0103: 'i' does not exist
```

; یک دستور خالی است و آن را به‌عنوان بدنه‌ی حلقه می‌گیرد؛ پس حلقه ۱۵ بار بدون هیچ کاری تکرار می‌شود و چاپ‌کردن بیرون از حلقه قرار می‌گیرد. چون i فقط درون for شناخته می‌شود، خطای کامپایل رخ می‌دهد. اگر i را پیش از حلقه تعریف کنیم، فقط یک بار و با مقدار نهایی (32) چاپ می‌شود.

مرحله‌ی ۴

```
    3    5    7    9   11   13   15   17   19   21   23   25   27   29   31
```

در هر دور i یک بار در سرصفحه‌ی for و یک بار در بدنه زیاد می‌شود (۲ واحد در هر دور)، پس بدنه روی اعداد فرد چاپ می‌کند. برای i=2 ابتدا i می‌شود ۳ و چاپ می‌شود؛ آخرین چاپ ۳۱ است (چون شرط فقط در ابتدای دور بررسی می‌شود).

مرحله‌ی ۵: حذف آکولادها

فقط ;++i بدنه‌ی حلقه است و Console.Write بعد از حلقه می‌آید که i را نمی‌شناسد؛ برنامه خطای کامپایل CS0103 می‌دهد و چیزی چاپ نمی‌شود.

مرحله‌ی ۶: شماره‌های فرد (نظافت کارگاه)

```
for (int i = 1; i <= 30; i += 2)
    Console.Write("{0,5}", i);
//     1    3    5 ... 29
```

مرحله‌ی ۷: i >= 30

شرط از همان ابتدا (30 =< 2) نادرست است، پس بدنه‌ی حلقه هیچ‌گاه اجرا نمی‌شود و خروجی خالی است.

۶۹ صفحه ۶۵ کنجکاوی

آیا متغیر حلقه می‌تواند از نوع byte، float یا char باشد؟

بله؛ متغیر حلقه می‌تواند هر نوع عددی یا نویسه‌ای باشد که عملگرهای مقایسه و ++ یا =+ را بپذیرد. اجرای آزمایشی:

```
for (byte b = 250; b < 255; b++) Console.Write(b + " ");        // 250 251 252 253 254
for (float f = 0; f <= 1; f += 0.25f) Console.Write(f + " ");  // 0 0.25 0.5 0.75 1
for (char c = 'a'; c <= 'e'; c++) Console.Write(c + " ");       // a b c d e
```

دقت کنید در byte اگر شرط تا ۲۵۵ برود (مثلاً b <= 255) حلقه هرگز تمام نمی‌شود، چون b از ۲۵۵ به ۰ برمی‌گردد. در float هم به‌دلیل خطای گردکردن، شرط‌های برابری توصیه نمی‌شود.

۷۰ صفحه ۶۵ فعالیت گروهی

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

حلقه‌ها و خروجی آنها

حلقه	خروجی برنامه (پیش‌بینی و اجرا)
for (int i=5; i>=1; i--) Console.Write(i);	54321
for (int i=0; i<10; i+=3) Console.WriteLine(i);	0 3 6 9 (هر عدد در یک خط)
for (int i=10; i>=0; i-=2) Console.WriteLine(i);	10 8 6 4 2 0 (هر عدد در یک خط)

ایران توشه
توشه ای برای موفقیت

۷۱ صفحه ۶۵ فعالیت منزل

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

```
for (int i = 1; i <= 5; i++)
    Console.WriteLine("Group {0}: student {1} and student {2}", i, i, 11 - i);
```

```
Group 1: student 1 and student 10
Group 2: student 2 and student 9
Group 3: student 3 and student 8
Group 4: student 4 and student 7
Group 5: student 5 and student 6
```

۷۲ صفحه ۶۵ کارگاه ۳

برنامه را اجرا کنید. عملکرد برنامه چیست؟

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

```
Console.Write("Final i :" + i);
```

۴ برنامه را برای مضارب ۳ بازنویسی کنید.

عملکرد برنامه

اعداد ۱ تا ۱۰۰ را چاپ می‌کند و به‌جای هر مضرب ۵ (۵، ۱۰، ۱۵، ...) کلمه‌ی Hop می‌نویسد: 1 2 3 4 Hop 6 7 8 9 Hop ...

دلیل خطا و راه‌حل

خطای CS0103: The name 'i' does not exist in the current context؛ چون i داخل پرانتز for تعریف شده و دامنه‌ی آن (بلاک) فقط خود حلقه است و بعد از پایان حلقه از بین می‌رود. راه‌حل: تعریف i پیش از حلقه.

```
int i;
for (i = 1; i <= 100; i++)
{
    if (i % 5 == 0) Console.WriteLine("Hop");
    else Console.WriteLine(i);
}
Console.Write("Final i :" + i);   // Final i :101
```

مرحله‌ی ۴: مضارب ۳

```
int i;
for (i = 1; i <= 100; i++)
    if (i % 3 == 0) Console.WriteLine("Hop");
    else Console.WriteLine(i);
```

۷۳ صفحه ۶۶ کارگاه ۴

۱ روندنمای مسئله را کامل کنید.

۳ قطعه کد زیر را در متد Main وارد کرده، برنامه را کامل کنید.

ایران توشه
توشه‌ای برای موفقیت

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

۴ برنامه را طوری تغییر دهید که مجموع ساعات مطالعه‌ی ماهانه‌ی هنرجو را نمایش دهد.

۵ برنامه را توسعه دهید تا میانگین ساعت مطالعه‌ی ماهانه‌ی هنرجو را نمایش دهد.

مرحله‌ی ۱: روندنما

بلوک‌های خالی: داخل حلقه پس از «دریافت زمان مطالعه‌ی روزانه» بلوک sum ← sum + minute و سپس بلوک i ← i + 1 قرار می‌گیرد؛ خروجی مسیر «نادرست» به بلوک «نمایش sum» و سپس پایان می‌رود.

مرحله‌ی ۳: کد کامل

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

با ورودی‌های ۳۰، ۴۵، ۶۰، ۰، ۹۰، ۱۲۰، ۱۵ خروجی نهایی: you studied for 360 minutes a week

مرحله‌ی ۴ و ۵: ماهانه (۳۰ روز)

```
int minute, sum = 0;
for (int i = 1; i <= 30; i++)
{
    Console.Write("Day {0} (minutes)= ", i);
    minute = int.Parse(Console.ReadLine());
    sum += minute;
}
Console.WriteLine("monthly study: {0:F1} hours", sum / 60.0);
Console.WriteLine("average per day: {0:F2} hours", sum / 60.0 / 30);
```

اگر هر روز ۶۰ دقیقه وارد شود: monthly study: 30.0 hours و average per day: 1.00 hours . تقسیم بر 60.0 (اعشاری) لازم است تا بخش اعشاری حذف نشود.

۷۴ صفحه ۶۷ کارگاه ۵

۲ کد زیر را تکمیل کرده، در متد Main وارد کنید و برنامه را اجرا کنید. در کد زیر متغیرهای مورد نیاز برنامه را اعلان کرده، دستورات دریافت آنها از ورودی را تکمیل کنید.

```
.............................
int score = 5;
Console.Write("Enter number your month:");
month = ....................................
Console.Clear();
for (int i = 1; i <= 5; i++)
{
    Console.Write("your guess?");
    guess = ....................................
    ...
```

{

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

هر دو متغیر month و guess عددی صحیح‌اند و ورودی کنسول رشته است، پس با int.Parse تبدیل می‌شود. break به‌محض حدس درست، حلقه را رها می‌کند.

ایران توشه
توشه ای برای موفقیت

۷۵ صفحه ۶۸ کارگاه ۵

۳ خروجی برنامه را با مقادیر زیر Trace کنید. (ماه ۱۱ با حدس‌های ۳، ۸، ۲، ۴، ۵ و ماه ۵ با حدس‌های ۳، ۸، ۵)

۴ برنامه را اصلاح کنید. پس از اجرای برنامه و Trace متوجه شدید که با حدس اشتباه در بار پنجم مشابه دفعات قبل پیام ! try again دیده می‌شود؛ برنامه را طوری تغییر دهید که پس از ۵ بار حدس اشتباه کاربر دوم، پیام مناسب دیگری چاپ شده، شماره‌ی ماه تولد را نمایش دهد.

Trace با month = 11 (مقدار score پس از پایان هر دور)

ماه ۱۱

خروجی	score	i	guess
! try again	4	1	3
! try again	3	2	8
! try again	2	3	2
! try again	1	4	4
! try again	0	5	5

Trace با month = 5

ماه ۵

خروجی	score	i	guess
! try again	4	1	3
! try again	3	2	8
you win / your score: 3	3	3	5

مرحله‌ی ۴: اصلاح

```
else
{
    score -= 1;
    if (i < 5)
        Console.WriteLine("try again !");
    else
        Console.WriteLine("you lose! The month number was {0}", month);
}
```

اجرای واقعی با ماه ۱۱ و حدس‌های بالا، در دور پنجم پیام you lose! The month number was 11 چاپ می‌شود و امتیاز نهایی ۰ است.

۷۶ صفحه ۶۸ **فعالیت گروهی**

جدول زیر را به کمک هم‌گروهی خود تکمیل کنید.

قطعه کدها و خروجی

قطعه کد	خروجی برنامه پس از اجرا
for (int k = 10; k <= 15;) { Console.WriteLine(k); k++; }	10 11 12 13 14 15 (هر عدد در یک خط)
= int i = 1; Console.ForegroundColor ConsoleColor.Cyan; for (; i<=10; i++) { ;++Console.WriteLine("Iran"); i }	کلمه‌ی Iran پنج بار (با رنگ فیروزه‌ای)؛ i در هر دور دو واحد زیاد می‌شود (۱، ۳، ۵، ۷، ۹) و در پایان ۱۱ است.
(;;) Console.ForegroundColor = ConsoleColor.Cyan; for { ;Console.WriteLine("Iran") }	حلقه‌ی بی‌پایان: Iran بی‌نهایت بار چاپ می‌شود (بدون شرط، حلقه هرگز تمام نمی‌شود؛ با Ctrl+C متوقف می‌شود).

هر سه بخش سرصفحه‌ی for (مقداردهی اولیه، شرط، تغییر) اختیاری است و در صورت حذف، ; باید بماند. نبودن شرط یعنی شرط همواره درست است.

۷۷ صفحه ۶۹ **فعالیت منزل**

برای قطعه کد زیر پس از اجرا و بررسی نتیجه، یک مسئله‌ی کاربردی بنویسید.

```
for (int i = 0; i < 100; i += 3)
    Console.WriteLine("{0,3}", i);
```

خروجی: اعداد ۰، ۳، ۶، ...، ۹۹ (۳۴ عدد) هر کدام در یک خط.

مسئله‌ی نمونه: یک کلاس ۱۰۰ نفره از هنرجویان داریم که از نفر اول، هر سومین نفر برای شرکت در مسابقه انتخاب می‌شود. برنامه‌ای بنویسید که شماره‌ی هنرجویان انتخاب‌شده (با شمارش از صفر) را چاپ کند.

۷۸ صفحه ۶۹ **فعالیت کارگاهی**

پس از مشاهده‌ی فیلم، برنامه‌ی MonthGuess را با کلید F10 گام به گام اجرا کنید. تغییرات متغیرها را در جدول زیر یادداشت کنید. (month، guess، i، score، خروجی)

برنامه‌ی کارگاه ۳ را با استفاده از پنجره‌ی Local گام به گام اجرا کنید. چرا پس از اجرا و اتمام کار حلقه، متغیر i از پنجره‌ی Local حذف می‌شود؟

نمونه برای month = 11 و حدس‌های ۳، ۸، ۲، ۴، ۵ (مقدارها پس از اجرای هر دور):

گام‌به‌گام

ایران توشه
توشه‌ای برای موفقیت

خروجی	score	i	guess	month
! try again	4	1	3	11
! try again	3	2	8	11
! try again	2	3	2	11
! try again	1	4	4	11
! try again	0	5	5	11

حذف i از پنجره‌ی Local

متغیر i در سرصفحه‌ی for تعریف شده، پس دامنه‌ی آن (scope) فقط همان حلقه است. با خروج از حلقه، حافظه‌ی آن آزاد می‌شود و دیگر در پنجره‌ی Local دیده نمی‌شود (دلیل خطای CS0103 هم همین است).

۷۹ صفحه ۶۹ فعالیت منزل

برنامه‌ای طراحی کنید که با دریافت نام هنرجویان یک هنرستان تعداد هنرجویانی که هم‌نام حضرت معصومه (س) هستند را نمایش دهد. تعداد کل هنرجویان هنرستان در ابتدای برنامه از کاربر دریافت شود.
برنامه را با کمک کلید F10 گام به گام اجرا کرده، با واردکردن نام ۵ هنرجو مقدار متغیرهای ورودی و خروجی را مشاهده کرده، در جدول زیر وارد کنید. (مقدار متغیر حلقه i / name / count / خروجی)

```
Console.Write("Number of students: ");
int n = int.Parse(Console.ReadLine());
int count = 0;
for (int i = 1; i <= n; i++)
{
    Console.Write("Enter name {0}: ", i);
    string name = Console.ReadLine();
    if (name == "Masoumeh")
        count++;
}
Console.WriteLine("Number of students named Masoumeh: {0}", count);
```

ورودی نمونه: Maryam، Ali، Masoumeh، Zahra، Masoumeh (نتیجه‌ی اجرای واقعی: ۲)

گام‌به‌گام

ایران توشه
توشه ای برای موفقیت

خروجی	count	name	(i) مقدار متغیر حلقه
—	1	Masoumeh	1
—	1	Zahra	2
—	2	Masoumeh	3
—	2	Ali	4
Number of students named Masoumeh: 2	2	Maryam	5

۸۰ صفحه ۷۰ فعالیت گروهی

با کمک هم‌گروهی و هنرآموز خود دو مثال دیگر برای حلقه‌ی نامعین بزنید.

- وارد کردن رمز عبور تا زمانی که رمز درست وارد شود (تعداد تلاش‌ها معلوم نیست).
- پرداخت قبوض یا خرید در فروشگاه تا زمانی که موجودی کارت تمام شود.
- خواندن داده‌ها از یک فایل تا رسیدن به انتهای آن.

۸۱ صفحه ۷۰ کارگاه ۶

۱ روندنمای مسئله را رسم کنید.

۳ قطعه کد زیر را در متد Main وارد کنید و برنامه را تکمیل کنید.

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

{

کاربرد متغیر count چیست؟

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

مرحله‌ی ۱: روندنما

ایران توشه
توشه ای برای موفقیت

```
Start
  sum <- 0 , count <- 0
  Receive mark
  <mark >= 0 ?> --False--> avg <- sum / count ; Show avg ; End
     | True
  sum <- sum + mark
  count <- count + 1
  Receive mark
  +--> back to the condition
```

مرحله‌ی ۳ و ۴: برنامه‌ی کامل

```
float sum = 0, avg;
int count = 0;
Console.Write("Enter mark {0} =", count + 1);
float mark = float.Parse(Console.ReadLine());
while (mark >= 0)
{
    sum = sum + mark;
    count++;
    Console.Write("Enter mark {0} =", count + 1);
    mark = float.Parse(Console.ReadLine());
}
if (count > 0)
{
    avg = sum / count;
    Console.WriteLine("\nCount = {0}, Average = {1:F2}", count, avg);
}
```

با ورودی ۱۸، ۱۵٫۵، ۱۲، ۲۰ و ۱-: Count = 4, Average = 16.38

شرط حلقه mark >= 0 است (ورود عدد منفی یعنی پایان). متغیر count تعداد نمرات واردشده را می‌شمارد و هم برای نمایش شماره‌ی نمره‌ی بعدی (count + 1) و هم برای مخرج میانگین به کار می‌رود. شرط count > 0 از تقسیم بر صفر در حالتی که هیچ نمره‌ای وارد نشده جلوگیری می‌کند.

ایران توشه
توشه ای برای موفقیت

۸۲ صفحه ۷۱ فعالیت کارگاهی

قطعه کدهای زیر را Trace کرده، اولین و آخرین مقدار چاپ‌شده را در جدول زیر بنویسید.

```
int i = 99;
while (i >= 1)
    Console.WriteLine(--i);
int i = 100;
while (i <= 1000)
    Console.WriteLine(++i);
```

Trace

دستور	اولین مقدار در خروجی	آخرین مقدار در خروجی
;int i = 99; while (i >= 1) Console.WriteLine(--i)	98	0
+)int i = 100; while (i <= 1000) Console.WriteLine ;(i+	101	1001

در i-- ابتدا کاهش و سپس چاپ انجام می‌شود؛ وقتی i برابر ۱ است شرط درست است، کاهش به ۰ می‌رسد و ۰ چاپ می‌شود. در حالت دوم هنگامی که i = 1000 است حلقه یک بار دیگر اجرا شده، ۱۰۰۱ چاپ می‌شود. هر دو نتیجه با اجرای واقعی تأیید شد.

۸۳ صفحه ۷۲ **کارگاه ۷**

در روندنما قسمت‌های مربوط به حلقه را پررنگ کنید.

در الگوریتم کارگاه، مراحل ۳ تا ۶ (بررسی شرط hour > 0، دریافت ساعت کاری نفر بعدی، مقایسه با max و بازگشت به مرحله‌ی ۳) بدنه‌ی حلقه هستند. در روندنما باید این بخش‌ها پررنگ شوند:

- لوزی شرط hour > 0 (در صورت «نادرست» خروج از حلقه و رفتن به نمایش max)،
- متوازی‌الاضلاع «دریافت ساعت کاری نفر بعدی (hour)»،
- لوزی شرط hour > max و مستطیل max ← hour (در صورت «درست»)،
- پیکان برگشت از انتهای بدنه به لوزی شرط hour > 0.

بخش‌های «شروع»، «دریافت ساعت کاری نفر اول»، «max ← hour» اولیه، «نمایش max» و «پایان» خارج از حلقه‌اند و پررنگ نمی‌شوند.

ایران توشه
توشه ای برای موفقیت

۸۴ صفحه ۷۳ **کارگاه ۷**

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

شرط خروج از حلقه در این برنامه چیست؟

۳ ـ برنامه را طوری تغییر دهید که کمترین ساعت کاری را نمایش دهد.

۴ ـ برنامه را طوری تغییر دهید که کمترین و بیشترین ساعت کاری را نمایش دهد.

تکمیل برنامه

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

با ورودی ۱۶۰، ۱۷۵، ۱۴۰ و ۰ خروجی max= 175 است.

شرط خروج از حلقه

حلقه تا وقتی ادامه می‌یابد که hour > 0 درست باشد؛ بنابراین با وارد کردن عدد صفر (یا عدد منفی) شرط نادرست شده، حلقه پایان می‌یابد.

کمترین ساعت کاری

فقط مقدار اولیه و شرط مقایسه تغییر می‌کند (برای همان ورودی‌ها خروجی ۱۴۰ است):

```
min = hour;
while (hour > 0)
{
    if (hour < min)
        min = hour;
    Console.Write("Enter hour (0 for Exit): ");
    hour = int.Parse(Console.ReadLine());
}
Console.WriteLine("min= {0}", min);
```

کمترین و بیشترین با هم

```
int hour, max, min;
Console.Write("Enter hour: ");
hour = int.Parse(Console.ReadLine());
max = min = hour;
while (hour > 0)
{
    if (hour > max) max = hour;
    if (hour < min) min = hour;
    Console.Write("Enter hour (0 for Exit): ");
    hour = int.Parse(Console.ReadLine());
}
Console.WriteLine("max= {0}, min= {1}", max, min);
```

ایران توشه
توشه ای برای موفقیت

۸۵ صفحه ۷۳ فعالیت کارگاهی

ماشین‌حساب کنسولی همانند شکل روبه‌رو ایجاد کنید.

برنامه دو عدد و یک عملگر را از ورودی می‌گیرد، نتیجه را چاپ می‌کند و تا زمانی که کاربر پایان را انتخاب نکرده است تکرار می‌شود (شکل کتاب در دسترس نبود):

```
char op = '+';
while (op != 'e')
{
    Console.Write("Enter first number: ");
    double a = double.Parse(Console.ReadLine());
    Console.Write("Enter operator (+ - * /): ");
    op = char.Parse(Console.ReadLine());
    Console.Write("Enter second number: ");
    double b = double.Parse(Console.ReadLine());
    switch (op)
    {
        case '+': Console.WriteLine("= {0}", a + b); break;
        case '-': Console.WriteLine("= {0}", a - b); break;
        case '*': Console.WriteLine("= {0}", a * b); break;
        case '/':
            if (b != 0) Console.WriteLine("= {0}", a / b);
            else Console.WriteLine("Division by zero!");
            break;
    }
    Console.Write("Press e to exit, any other key to continue: ");
    op = char.Parse(Console.ReadLine());
}
```

این جواب ممکن است دقیق نباشد.

۸۶ صفحه ۷۴ فعالیت گروهی

به کمک هم‌گروهی خود شکل را توصیف کنید.

در این مثال‌ها یک کار تکراری (حلقه‌ی بیرونی) انجام می‌شود و در هر بار تکرار آن، کار تکرارشونده‌ی دیگری (حلقه‌ی درونی) به‌طور کامل اجرا می‌شود؛ مانند عقربه‌های ساعت: به ازای هر یک حرکت عقربه‌ی کند، عقربه‌ی تند چندین بار می‌چرخد.

- حلقه‌ی بیرونی: ۵ دور دویدن دور حیاط، یا ۱۰ طبقه‌ی مجتمع.
- حلقه‌ی درونی: ۳ حرکت بارفیکس در پایان هر دور، یا ۸ واحد هر طبقه.
- تعداد کل تکرار کارِ درونی = تعداد تکرار حلقه‌ی بیرونی × تعداد تکرار حلقه‌ی درونی (مثلاً ۵×۳ = ۱۵ حرکت بارفیکس و ۱۰×۸ = ۸۰ غذای نذری).

ایران توشه
توشه‌ای برای موفقیت

این جواب ممکن است دقیق نباشد.

۸۷ صفحه ۷۵ کارگاه ۸

۲ ـ ... برنامه را اجرا کنید. آیا خروجی یک جدول‌ضرب را نشان می‌دهد؟

۳ ـ با توجه به اعداد مشخص‌شده یک تا پنج، در کد مرحله ۲ با استفاده از دستور حلقه‌ی for دستورهای Console.Write را به یک دستور تبدیل کنید.

۴ ـ برای تنظیم نمایش جدول‌ضرب به جای جانگهدار از دستور SetCursorPosition استفاده کنید.

۵ ـ برنامه را با کمک کلید F10 خط به خط اجرا کنید.

بله؛ هر بار اجرای حلقه‌ی for، یک سطر جدول (ضرب‌های ۱ تا ۵ برای i) را در ستون‌هایی با عرض ۴ می‌نویسد و Console.WriteLine() به سطر بعد می‌رود. خروجی (اجرای واقعی):

```
   1   2   3   4   5
   2   4   6   8  10
   3   6   9  12  15
   4   8  12  16  20
   5  10  15  20  25
```

تبدیل پنج دستور Write به یک حلقه (حلقه‌ی متداخل)

```
for (int i = 1; i <= 5; i++)
{
    for (int j = 1; j <= 5; j++)
        Console.Write("{0,4}", i * j);
    Console.WriteLine();
}
```

استفاده از SetCursorPosition

```
for (int i = 1; i <= 5; i++)
    for (int j = 1; j <= 5; j++)
    {
        Console.SetCursorPosition(j * 4, i);   // ستون، سطر
        Console.Write(i * j);
    }
Console.SetCursorPosition(0, 7);
```

در SetCursorPosition(left, top) پارامتر اول شماره‌ی ستون و دومی شماره‌ی سطر است و مکان‌نما را قبل از چاپ هر عدد جابه‌جا می‌کند.

با F10 هر بار یک دستور اجرا می‌شود؛ مشاهده می‌شود به ازای هر مقدار i، حلقه‌ی درونی j پنج بار (۱ تا ۵) کامل اجرا شده و سپس i یک واحد زیاد می‌شود.

ایران توشه
توشه ای برای موفقیت

۸۸ صفحه ۷۵ فعالیت گروهی

اگر خودپردازی دارای اسکناس‌های ۲ و ۵ هزار تومانی باشد و شما درخواست ۵۰ هزار تومان وجه نقد کنید، چگونه پرداخت را انجام می‌دهد؟ چه روش‌های پرداختی می‌تواند داشته باشد؟ جدول روبه‌رو را تکمیل کنید.

اگر x تعداد اسکناس ۲ هزار تومانی و y تعداد اسکناس ۵ هزار تومانی باشد، باید 2000x + 5000y = 50000 شود. همه‌ی حالت‌ها:

جدول پرداخت ۵۰ هزار تومان

تعداد اسکناس ۵ هزار تومانی	تعداد اسکناس ۲ هزار تومانی
۱۰	۰
۸	۵
۶	۱۰
۴	۱۵
۲	۲۰
۰	۲۵

خودپرداز معمولاً با توجه به موجودی اسکناس‌ها، ترجیحاً کمترین تعداد اسکناس (۱۰ اسکناس ۵ هزار تومانی) را می‌پردازد.

۸۹ صفحه ۷۵ کارگاه ۹

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

کافی است نوبت‌دهی یک روز را داخل یک حلقه‌ی بیرونی با شش تکرار (شش روز کاری) قرار دهیم:

```
string[] days = {"Saturday", "Sunday", "Monday", "Tuesday", "Wednesday", "Thursday"};
for (int d = 0; d < 6; d++)
{
    Console.WriteLine("--- {0} ---", days[d]);
    for (int hour = 16; hour < 20; hour++)          // ساعت‌های کاری
        for (int min = 0; min < 60; min += 30)       // هر نوبت ۳۰ دقیقه
            Console.WriteLine("{0:00}:{1:00}", hour, min);
}
```

این جواب ممکن است دقیق نباشد.

ایران توشه
توشه ای برای موفقیت

۹۰ صفحه ۷۶ فعالیت منزل

برنامه را اجرا کنید. آیا با جدولی که شما کامل کرده‌اید مطابقت دارد؟ فکر می‌کنید خودپرداز کدام پرداخت را انجام می‌دهد؟

چرا مقدار نهایی X مقدار ۲۵ و مقدار نهایی Y مقدار ۱۰ در نظر گرفته شده است؟

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

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

۵ ـ برنامه را طوری تغییر دهید که وجه نقد درخواستی کاربر را از ورودی دریافت کنیم.

خروجی برنامه (اجرای واقعی): X=0,Y=10، X=5,Y=8، X=10,Y=6، X=15,Y=4، X=20,Y=2 و X=25,Y=0 است که دقیقاً با شش سطر جدول بالا یکی است. خودپرداز معمولاً حالتی را انتخاب می‌کند که تعداد اسکناس کمتر باشد، یعنی X=0 و Y=10.

بیشترین تعداد اسکناس ۲ هزار تومانی برای ۵۰ هزار تومان ۵۰,۰۰۰ ÷ ۲,۰۰۰ = ۲۵ و بیشترین تعداد اسکناس ۵ هزار تومانی ۵۰,۰۰۰ ÷ ۵,۰۰۰ = ۱۰ است؛ بنابراین حدّ بالای حلقه‌ها همین مقادیر است.

حلقه‌ی بیرونی ۲۶ بار (X از ۰ تا ۲۵) و درونی برای هر بار ۱۱ بار (Y از ۰ تا ۱۰) اجرا می‌شود؛ یعنی ۲۶ × ۱۱ = ۲۸۶ حالت آزمایش می‌شود (با اجرای واقعی تأیید شد).

با اسکناس ۱۰ هزار تومانی (Z) و گرفتن مبلغ از ورودی

```
Console.Write("Enter amount: ");
int amount = int.Parse(Console.ReadLine());
for (int X = 0; X <= amount / 2000; X++)
    for (int Y = 0; Y <= amount / 5000; Y++)
        for (int Z = 0; Z <= amount / 10000; Z++)
            if (2000 * X + 5000 * Y + 10000 * Z == amount)
                Console.WriteLine("2000:{0}  5000:{1}  10000:{2}", X, Y, Z);
```

ایران توشه
توشه ای برای موفقیت

۹۱ صفحه ۷۶ فعالیت منزل

ـ برنامه‌ی کارگاه ۶ را توسعه دهید تا با دریافت نمرات بقیه‌ی دروس پایه‌ی یازدهم، میانگین آنها را محاسبه و نمایش دهد.

ـ کمترین و بیشترین میانگین نمره را نمایش دهید.

برای هر درس میانگین نمرات محاسبه می‌شود (پایان ورود نمره با عدد منفی) و میانگین‌ها با کمترین و بیشترین مقایسه می‌شوند: حلقه‌ی بیرونی روی دروس و حلقه‌ی درونی (while) برای ورود نمرات هر درس.

```
Console.Write("Number of courses: ");
int n = int.Parse(Console.ReadLine());
float max = float.MinValue, min = float.MaxValue;
for (int c = 1; c <= n; c++)
{
    float sum = 0; int count = 0;
    Console.Write("Course {0}, enter mark (negative to stop): ", c);
    float mark = float.Parse(Console.ReadLine());
    while (mark >= 0)
    {
        sum += mark; count++;
        Console.Write("Enter mark: ");
        mark = float.Parse(Console.ReadLine());
    }
    if (count == 0) continue;
    float avg = sum / count;
    Console.WriteLine("Average of course {0} = {1:F2}", c, avg);
    if (avg > max) max = avg;
    if (avg < min) min = avg;
}
Console.WriteLine("Max average = {0:F2}, Min average = {1:F2}", max, min);
```

۹۲ صفحه ۷۹ **فعالیت کارگاهی**

روندنمایی رسم کنید که اسامی پنج هنرجو را دریافت کرده، سپس اسامی را از آخر به اول نمایش دهد.

برنامه‌ی این الگوریتم را به زبان سی‌شارپ بنویسید.

در این مثال، اگر تعداد اسامی ۳۰ نفر باشد: چه تعداد دستور خواندن از ورودی و نوشتن روی صفحه‌نمایش لازم است؟ آیا می‌توان از دستور for برای خواندن ورودی‌ها و نوشتن اسامی استفاده کرد؟ برای چنین مسئله‌هایی که با داده‌های زیاد سروکار دارند، چه روشی را پیشنهاد می‌کنید؟

ایران توشه
توشه ای برای موفقیت

روندنما (شرح)

1. شروع
2. دریافت نام اول تا پنجم و قرار دادن در متغیرهای name1 تا name5 (پنج متوازی‌الاضلاع ورودی)
3. نمایش name5، سپس name2، name3، name4 و name1 (پنج متوازی‌الاضلاع خروجی)
4. پایان

برنامه‌ی C#

```
string name1, name2, name3, name4, name5;
name1 = Console.ReadLine();
name2 = Console.ReadLine();
name3 = Console.ReadLine();
name4 = Console.ReadLine();
name5 = Console.ReadLine();
Console.WriteLine(name5);
Console.WriteLine(name4);
Console.WriteLine(name3);
Console.WriteLine(name2);
Console.WriteLine(name1);
```

برای ۳۰ نفر به ۳۰ متغیر، ۳۰ دستور خواندن و ۳۰ دستور نوشتن (جمعاً ۶۰ دستور) نیاز است. چون هر نام در یک متغیر جدا با نام متفاوت قرار دارد نمی‌توان با for به متغیرها دسترسی یافت. راه‌حل مناسب استفاده از آرایه است: یک نام مشترک و شماره‌ی اندیس برای هر عنصر؛ در این صورت حلقه‌ی for با اندیس به همه‌ی عناصر دسترسی می‌یابد.

۹۳ صفحه ۸۰ **فعالیت کارگاهی**

یک پروژه‌ی جدید ایجاد کنید و دستور اعلان آرایه و دستور تخصیص حافظه به آرایه‌ی name را بنویسید.

بررسی کنید محتوای هر عنصر آرایه چیست؟

```
string[] name;          // اعلان
name = new string[5];   // تخصیص حافظه برای ۵ عنصر
```

با اجرای برنامه با F10 و دیدن محتوای آرایه، هر پنج عنصر (name[0] تا name[4]) برابر null هستند؛ زیرا حافظه تخصیص یافته ولی هنوز مقداری در عناصر قرار نگرفته است (مقدار پیش‌فرض عناصر رشته‌ای null است).

ایران توشه
توشه ای برای موفقیت

۹۴ صفحه ۸۱ **فعالیت کارگاهی**

ـ آرایه‌ای به نام salary برای نگهداری حقوق ۳۰۰ کارمند اعلان کنید.

ـ آرایه‌ای به نام vowels برای نگهداری حروف صدادار انگلیسی اعلان کنید.

ـ آرایه‌ای به نام lamp برای نگهداری وضعیت روشن و خاموش بودن پنج لامپ اعلان کنید.

ـ آرایه‌ای با نام دلخواه برای نگهداری اسامی ماه‌های سال اعلان کنید.

ـ آرایه‌ای با نام دلخواه برای نگهداری معدل هنرجویان کلاس اعلان کنید.

```
long[] salary = new long[300];
char[] vowels = {'a', 'e', 'i', 'o', 'u'};
bool[] lamp = new bool[5];                  // true = روشن ، false = خاموش
string[] months = new string[12];           // اسامی ماه‌های سال
float[] gpa = new float[30];                // معدل هنرجویان (مثلاً ۳۰ نفر)
```

نوع داده‌ی آرایه باید با نوع داده‌ی عناصر آن هم‌خوان باشد: حقوق عدد صحیح بزرگ، حروف char، وضعیت دوحالته bool، نام ماه string و معدل عدد اعشاری.

۹۵ صفحه ۸۱ کنجکاوی

چرا برای مراجعه به خانه‌ی سوم آرایه، از اندیس ۲ استفاده شده است؟

چون در سی‌شارپ (و بیشتر زبان‌های برنامه‌نویسی) اندیس اولین عنصر آرایه صفر است. پس عنصر اول اندیس ۰، عنصر دوم اندیس ۱ و عنصر سوم اندیس ۲ دارد؛ یعنی اندیس هر عنصر یک واحد کمتر از شماره‌ی ترتیبی آن است.

۹۶ صفحه ۸۲ کارگاه ۱

۴ ـ برای ذخیره کردن اسامی قطعات دیگر، چه تغییری در کد باید بدهیم؟

چرا مقدار برخی از عناصر برابر null است؟

۶ ـ برای نگهداری قیمت قطعات رایانه، آرایه‌ی price را متناظر با آرایه‌ی نام قطعات اعلان کنید.

۷ ـ عناصر آرایه‌ی price را مقداردهی کنید.

۸ ـ محتوای آرایه‌ی price چیست؟ چرا؟

ایران توشه
توشه ای برای موفقیت

برای ذخیره‌ی نام قطعات بیشتر باید اندازه‌ی آرایه را بزرگ‌تر کنیم (مثلاً new string[8]) و سپس به عناصر جدید مقدار بدهیم؛ زیرا اندازه‌ی آرایه پس از ساخته شدن ثابت است. عناصری که مقداردهی نشده‌اند مقدار null دارند، چون هنوز مقداری در آنها قرار نگرفته است.

```
long[] price = new long[5];
price[0] = 200000;
price[1] = 150000;
price[2] = 300000;
price[3] = 450000;
price[4] = 100000;
```

هنگام اجرا با F10 تا رسیدن به دستور مقداردهی price[2]: price[0] برابر ۲۰۰۰۰۰ و price[1] برابر ۱۵۰۰۰۰ است و عناصر ۲ تا ۴ برابر ۰ هستند؛ زیرا مقدار پیش‌فرض عناصر عددی صفر است و هنوز مقداردهی نشده‌اند.

۹۷ صفحه ۸۳ کارگاه ۱

۱۰ ـ دستور زیر را جایگزین کد مرحله ۹ کنید ... به‌جای عدد ۵ عدد ۴ را قرار دهید. خطای رخ‌داده چیست؟

۱۲ ـ دستوری برای تغییر مقدار عنصر شماره‌ی ۳ آرایه‌ی computer به "Keyboard" به کد اضافه کنید.

۱۳ ـ دستوری بنویسید که قیمت Keyboard را از ورودی دریافت کرده و در عنصر شماره‌ی ۳ آرایه‌ی price قرار دهد.

اگر در new string[4]{"CPU", "MainBoard", "RAM", "HDD", "DVD"} اندازه ۴ باشد ولی ۵ مقدار داده شود، خطای زمان کامپایل رخ می‌دهد: تعداد مقادیر با اندازه‌ی آرایه برابر نیست (Array initializers must have exactly the declared number of elements). برای جلوگیری از این خطا بهتر است آرایه بدون new و اندازه تعریف شود.

منظور از «عنصر شماره‌ی ۳» اندیس ۳ است (چهارمین عنصر، HDD):

```
computer[3] = "Keyboard";
price[3] = long.Parse(Console.ReadLine());
```

۹۸ صفحه ۸۵ کنجکاوی

آیا می‌توان با یک متد WriteLine() تمام عناصر آرایه را نمایش داد؟ حاصل اجرای دستور Console.WriteLine(price); چیست؟

خیر؛ Console.WriteLine(price) عناصر را چاپ نمی‌کند، بلکه نام نوع آرایه را نمایش می‌دهد؛ برای مثال برای آرایه‌ی long[] خروجی System.Int64[] است. برای نمایش عناصر باید از حلقه (یا string.Join) استفاده کرد.

۹۹ صفحه ۸۵ کارگاه ۲

۲ ـ اعلان آرایه‌ی week را تکمیل کنید.

۴ ـ برنامه را با عدد ۷ اجرا کنید. چه خطایی رخ می‌دهد؟ علت آن چیست؟

۵ ـ اگر بخواهیم به‌جای اعداد ۰ تا ۶، اعداد ۱ تا ۷ دریافت شود، چه تغییری باید در برنامه ایجاد کنیم؟

ایران توشه
توشه‌ای برای موفقیت

۶ ـ دستوراتی به برنامه اضافه کنید که اگر شماره‌ی واردشده در محدوده‌ی مجاز نبود، پیام مناسب نمایش داده شود.

```
string[] week = {"saturday", "sunday", "monday", "tuesday", "wednesday", "thursday",
"friday"};
Console.Write("Enter the number one day of week (0-6): ");
byte num = byte.Parse(Console.ReadLine());
Console.WriteLine(week[num]);
```

با عدد ۷ خطای زمان اجرا IndexOutOfRangeException (Index was outside the bounds of the array) رخ می‌دهد؛ زیرا آرایه‌ی week هفت عنصر با اندیس‌های ۰ تا ۶ دارد و اندیس ۷ وجود ندارد.

برای دریافت عدد ۱ تا ۷، از شماره‌ی واردشده یک واحد کم می‌کنیم: week[num – 1].

```
Console.Write("Enter the number of day (1-7): ");
byte num = byte.Parse(Console.ReadLine());
if (num >= 1 && num <= 7)
    Console.WriteLine(week[num - 1]);
else
    Console.WriteLine("Invalid number! Enter a number between 1 and 7.");
```

۱۰۰ صفحه ۸۶ فعالیت کارگاهی

برنامه‌ای بنویسید که اسامی روزهای زوج آرایه‌ی week را نمایش دهد.

منظور روزهایی است که اندیس آنها زوج است (۰، ۲، ۴، ۶): شنبه، دوشنبه، چهارشنبه و جمعه.

```
string[] week = {"saturday", "sunday", "monday", "tuesday", "wednesday", "thursday",
"friday"};
for (int i = 0; i < week.Length; i += 2)
    Console.WriteLine(week[i]);
```

خروجی: saturday، monday، wednesday، friday. برای آرایه‌ی ۱۰۰ عنصری هم کافی است حد حلقه ۱۰۰ باشد: for (int i = 0; i < 100; i++).

۱۰۱ صفحه ۸۷ کارگاه ۳

۲ ـ تعداد قطعات رایانه را از ورودی دریافت کنید و دو آرایه با نام‌های computer و price اعلان کنید.

۴ ـ کد مرحله‌ی قبل را تغییر دهید تا علاوه بر عناصر computer، قیمت قطعات را هم دریافت کنید.

ایران توشه
توشه‌ای برای موفقیت

۶ ـ برنامه را طوری تکمیل کنید که نام، قیمت قطعات و قیمت کل را نمایش دهد.

```
Console.Write("Number of parts: ");
int n = int.Parse(Console.ReadLine());
string[] computer = new string[n];
long[] price = new long[n];
long sumPrice = 0;
for (int i = 0; i < computer.Length; i++)
{
    Console.Write("Enter the name of a computer part: ");
    computer[i] = Console.ReadLine();
    Console.Write("Enter its price: ");
    price[i] = long.Parse(Console.ReadLine());
    sumPrice += price[i];
}
for (int i = 0; i < computer.Length; i++)
    Console.WriteLine("{0,-12}{1,10}", computer[i], price[i]);
Console.WriteLine("Total price = {0}", sumPrice);
```

۱۰۲ صفحه ۸۸ کارگاه ۴

۳ ـ ... برنامه را اجرا کنید. خروجی آن چیست؟

۴ ـ کد زیر را جایگزین مرحله‌ی ۳ کنید (foreach) ... نتیجه و کد را با مرحله‌ی ۳ مقایسه کنید.

۵ ـ برای نمایش معکوس رشته، کد زیر را تکمیل کنید. آیا می‌توان این کد را با دستور foreach نوشت؟

```
Console.Write("Enter a string: ");
string myStr = Console.ReadLine();
for (int i = 0; i < myStr.Length; i++)
    Console.WriteLine(myStr[i]);
```

با ورودی Ali نویسه‌های A، l و i هر یک در یک سطر چاپ می‌شوند. کد foreach (char c in myStr) Console.WriteLine(c); نیز دقیقاً همان خروجی را می‌دهد، ولی بدون استفاده از اندیس و با کد کوتاه‌تر.

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

خیر؛ foreach همیشه عناصر را از اول تا آخر پیمایش می‌کند و امکان پیمایش از آخر به اول ندارد.

۱۰۳ صفحه ۸۹ کارگاه ۵

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

۳ ـ دستورات محاسبه‌ی مالیات ارزش‌افزوده را بنویسید ... در این کد نقش متغیر i و j چیست؟

ایران توشه
توشه‌ای برای موفقیت

۴ ـ در قطعه کد مرحله‌ی ۳ تبدیل صریح به int را حذف کنید. چه خطایی رخ می‌دهد؟ چرا؟

```
int[] priceList = new int[50];
int i, temp;
for (i = 0; i < 50; i++)
{
    Console.Write("Enter the price of {0} the product: ", i + 1);
    temp = int.Parse(Console.ReadLine());
    if (temp > 0)
        priceList[i] = temp;
    else
        break;
}
Console.WriteLine("Product price plus VAT:");
for (int j = 0; j < i; j++)
{
    priceList[j] = (int)(priceList[j] + 0.09 * priceList[j]);
    Console.WriteLine("Product{0}: {1}", j, priceList[j]);
}
```

متغیر i تعداد قیمت‌های معتبر واردشده را نگه می‌دارد (در حلقه‌ی اول شمارنده‌ی ورودی‌ها و پس از خروج با break تعداد محصولات است). متغیر j اندیس پیمایش عناصر پُر آرایه (از ۰ تا i–1) برای محاسبه‌ی قیمت با مالیات است. مثلاً برای قیمت‌های ۱۰۰۰، ۲۰۰۰ و ۳۵۰۰ نتیجه ۱۰۹۰، ۲۱۸۰ و ۳۸۱۵ می‌شود (اجرای واقعی).

اگر (int) حذف شود، خطای زمان کامپایل Cannot implicitly convert type 'double' to 'int' رخ می‌دهد؛ چون عبارت priceList[j] * 0.09 + [priceList[j] نتیجه‌ی double دارد و تبدیل ضمنی double به int چون ممکن است بخش اعشاری از بین برود مجاز نیست؛ باید تبدیل صریح انجام شود.

۱۰۴ صفحه ۸۹ کنجکاوی

ـ در دستور foreach منظور از متغیر حلقه چیست؟

ـ آیا می‌توان به‌وسیله‌ی عملگر انتساب (=) مقداری را داخل متغیر حلقه‌ی foreach قرار داد؟

متغیر حلقه‌ی foreach در هر بار تکرار یک عنصر آرایه را (به ترتیب از اولین تا آخرین عنصر) نگه می‌دارد و نوع آن باید با نوع عناصر آرایه یکی باشد.

خیر؛ متغیر حلقه‌ی foreach فقط‌خواندنی است و انتساب به آن خطای کامپایل می‌دهد (Cannot assign to ... because it is a 'foreach iteration variable'). بنابراین با foreach نمی‌توان عناصر آرایه را تغییر داد.

ایران توشه
توشه ای برای موفقیت

۱۰۵ صفحه ۸۹ فعالیت منزل

برنامه‌ای بنویسید که سطح شایستگی هنرجویان یک کلاس را به‌صورت عددی دریافت کرده، به‌وسیله‌ی دستور foreach تعداد هنرجویانی که سطح شایستگی لازم را کسب نکرده‌اند یعنی دارای سطح شایستگی ۱ هستند نمایش دهد. بررسی شود که سطح شایستگی دریافت‌شده یکی از سه سطح ۱، ۲ و ۳ باشد.

```
Console.Write("Number of students: ");
int n = int.Parse(Console.ReadLine());
int[] level = new int[n];
for (int i = 0; i < n; i++)
{
    do
    {
        Console.Write("Level of student {0} (1-3): ", i + 1);
        level[i] = int.Parse(Console.ReadLine());
    } while (level[i] < 1 || level[i] > 3);   // فقط سطح ۱، ۲ یا ۳ پذیرفته می‌شود
}
int count = 0;
foreach (int l in level)
    if (l == 1) count++;
Console.WriteLine("Not competent: {0}", count);
```

۱۰۶ صفحه ۹۰ فعالیت منزل

برنامه‌ای بنویسید که هزینه‌ی روزانه‌ی یک ماه شما را دریافت کرده، در یک آرایه ذخیره کند. سپس هزینه‌ی ماهانه و میانگین هزینه‌ی روزانه‌ی شما را محاسبه کرده، نمایش دهد. سپس تعداد روزهایی که هزینه‌ی روزانه بیشتر از میانگین شده‌است را محاسبه کرده و نمایش دهد.

```
long[] cost = new long[30];
long sum = 0;
for (int i = 0; i < cost.Length; i++)
{
    Console.Write("Cost of day {0}: ", i + 1);
    cost[i] = long.Parse(Console.ReadLine());
    sum += cost[i];
}
double avg = (double)sum / cost.Length;
int above = 0;
foreach (long c in cost)
    if (c > avg) above++;
Console.WriteLine("Monthly cost = {0}", sum);
Console.WriteLine("Average daily cost = {0:F1}", avg);
Console.WriteLine("Days above average = {0}", above);
```

ایران توشه
توشه ای برای موفقیت

۱۰۷ صفحه ۹۱ فعالیت گروهی

در هر یک از موارد زیر تعیین کنید از روش جست‌وجوی خطی یا دودویی استفاده می‌شود؟ انتخاب روش بر اساس چه معیاری است؟

نوع جست‌وجو

مورد	روش	دلیل
۱ ـ یافتن کارنامه‌ی یک هنرجو بین کارنامه‌های مرتب‌شده بر اساس نام	دودویی	داده‌ها بر اساس نام مرتب‌اند
۲ ـ پیدا کردن یک کتاب بر اساس نام آن در کتابخانه	دودویی	کتاب‌ها معمولاً بر اساس حروف الفبا مرتب‌اند
۳ ـ پیدا کردن یک کتاب بر اساس کد کتاب در کتابخانه	دودویی	کدها مرتب (ترتیبی) هستند
۴ ـ پیدا کردن نام هنرجویی که بالاترین معدل را دارد، اگر اسامی بر اساس معدل مرتب باشد	نیازی به جست‌وجو نیست	بالاترین معدل در ابتدا یا انتهای فهرست مرتب قرار دارد؛ در حالت کلی دودویی

معیار انتخاب: اگر داده‌ها مرتب باشند، جست‌وجوی دودویی سریع‌تر است؛ اگر مرتب نباشند فقط جست‌وجوی خطی ممکن است.

۱۰۸ صفحه ۹۱ فعالیت گروهی

برای پیدا کردن نام یک هنرجو در این آرایه به روش جست‌وجوی خطی چند مقایسه نیاز است؟ جدول زیر را کامل کنید.

تعداد مقایسه

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

۱۰۹ صفحه ۹۲ کارگاه ۶

۲ ـ برای خودروها و قیمت آنها دو آرایه‌ی متناظر اعلان کرده، مقداردهی کنید.

۳ ـ ... برنامه را اجرا کنید. خروجی برنامه چیست؟

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

ایران توشه
توشه ای برای موفقیت

۵ ـ مقدار item را برابر Hyundai قرار دهید. خروجی چیست؟

نقش متغیر found در این کد چیست؟ به‌جای شرط found==false عبارت !found را قرار دهید و برنامه را اجرا کنید.

۷ ـ برنامه را طوری تغییر دهید که نام خودرو را از ورودی دریافت کند. با دو مقدار Renault و Hyundai نتیجه‌ی برنامه را مشاهده کنید.

```
string[] car = {"Pride", "Mazda", "Renault", "Peugeot", "Toyota"};
long[] price = {2500, 6000, 4500, 5000, 7000};   // قیمت‌ها (فرضی)
Console.Write("Enter car name: ");
string item = Console.ReadLine();
bool found = false;
for (int i = 0; i < car.Length; i++)
    if (item == car[i])
    {
        found = true;
        Console.WriteLine("price {0}={1}", item, price[i]);
        break;
    }
if (!found)
    Console.WriteLine("item Not Found");
```

با "item = "Pride خروجی مرحله‌ی ۳ Pride Found in 0 است (اولین عنصر آرایه) و در مرحله‌ی ۴ قیمت آن (price Pride=2500) چاپ می‌شود. با Hyundai چون در آرایه نیست در مرحله‌ی ۵ هیچ خروجی‌ای چاپ نمی‌شود و در مرحله‌ی ۶ پیام item Not Found نمایش داده می‌شود. با Renault خروجی price Renault=4500 است.

متغیر found پرچم (flag) است: مقدار اولیه‌اش false است و فقط وقتی عنصر پیدا شود true می‌شود تا بعد از حلقه معلوم باشد جست‌وجو موفق بوده یا نه. عبارت !found دقیقاً همان معنی found == false را دارد و خروجی یکسان است.

قیمت‌های آرایه‌ی price در کتاب نیامده و فرضی هستند؛ هر مقدار دلخواه پذیرفته است.

ایران توشه
توشه ای برای موفقیت

۱۱۰ صفحه ۹۳ فعالیت منزل

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

```
Console.Write("Number of students: ");
int n = int.Parse(Console.ReadLine());
string[] name = new string[n];
float[] mark = new float[n];
for (int i = 0; i < n; i++)
{
    Console.Write("Name: ");
    name[i] = Console.ReadLine();
    Console.Write("Mark: ");
    mark[i] = float.Parse(Console.ReadLine());
}
Console.WriteLine("Students with mark 20:");
for (int i = 0; i < n; i++)
    if (mark[i] == 20)
        Console.WriteLine(name[i]);
```

۱۱۱ صفحه ۹۳ فعالیت گروهی

ـ شما چه الگوریتمی برای برنده شدن پیشنهاد می‌دهید؟ بهتر است اولین حدس چه عددی باشد؟

ـ هنرجویی عدد ۵۹ را یادداشت کرده است. در روش جست‌وجوی دودویی حداکثر با چند مقایسه عدد مورد نظر حدس زده خواهد شد؟ جدول زیر را کامل کنید.

ـ در جست‌وجوی دودویی اگر عنصر مورد جست‌وجو در آرایه نباشد، الگوریتم چگونه پایان می‌یابد؟

بهترین الگوریتم، حدس زدن عدد وسط بازه و نصف کردن بازه‌ی باقی‌مانده بر اساس پاسخ «بزرگ‌تر/کوچک‌تر» است. اولین حدس باید وسط بازه‌ی ۱ تا ۱۰۰ یعنی 50 باشد. برای عدد ۵۹ پنج حدس لازم است:

جدول حدس‌ها

مرحله	اولین عدد	آخرین عدد	عدد حدس‌زده‌شده (وسط)	نتیجه‌ی مقایسه
۱	۱	۱۰۰	۵۰	عدد بزرگ‌تر است
۲	۵۱	۱۰۰	۷۵	عدد کوچک‌تر است
۳	۵۱	۷۴	۶۲	عدد کوچک‌تر است
۴	۵۱	۶۱	۵۶	عدد بزرگ‌تر است
۵	۵۷	۶۱	۵۹	برابر است (پیدا شد)

ایران توشه
توشه ای برای موفقیت

حداکثر تعداد مقایسه برای بازه‌ی ۱۰۰ عددی حدود ۷ است (چون ۲ به توان ۷ برابر ۱۲۸ است). اگر عدد در آرایه نباشد، با هر مقایسه بازه نصف می‌شود تا اولین اندیس از آخرین اندیس بزرگ‌تر شود (بازه خالی)؛ آنگاه الگوریتم با پیام «پیدا نشد» پایان می‌یابد.

۱۱۲ صفحه ۹۳ فعالیت منزل

با در نظر گرفتن عدد انتخابی ۱۲، جدول بالا را پر کنید.

جست‌وجوی دودویی عدد ۱۲

مرحله	اولین عدد	آخرین عدد	عدد حدس‌زده‌شده (وسط)	نتیجه‌ی مقایسه
۱	۱	۱۰۰	۵۰	عدد کوچک‌تر است
۲	۱	۴۹	۲۵	عدد کوچک‌تر است
۳	۱	۲۴	۱۲	برابر است (پیدا شد)

۱۱۳ صفحه ۹۴ کارگاه ۷

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

۴ ـ اگر x=60 قرار دهیم خروجی چه می‌شود؟

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

۷ ـ با دستور foreach عناصر آرایه‌ی a را در خروجی نمایش دهید. عملکرد متد Sort چیست؟

۸ ـ مقدار متغیر found چیست؟ (BinarySearch)

۹ ـ ... عملکرد متد Reverse چیست؟

ایران توشه
توشه ای برای موفقیت

۱۰ ـ مقدار متغیر found چیست؟

```
int[] a = {20, 18, 10, 15, 50, 80, 10, 37};
int x = 10;
int found = Array.IndexOf(a, x);
Console.WriteLine(found);          // 2
found = Array.LastIndexOf(a, x);
Console.WriteLine(found);          // 6
Array.Sort(a);
foreach (int v in a) Console.Write(v + " ");   // 10 10 15 18 20 37 50 80
found = Array.BinarySearch(a, x);
Console.WriteLine(found);          // 1
Array.Reverse(a);
foreach (int v in a) Console.Write(v + " ");   // 80 50 37 20 18 15 10 10
found = Array.BinarySearch(a, x);
Console.WriteLine(found);          // -1
```

نتایج اجرای واقعی

دستور	نتیجه	توضیح
Array.IndexOf(a, 10)	۲	اندیس اولین عنصر برابر ۱۰ را برمی‌گرداند
Array.IndexOf(a, 60)	۱-	عنصر وجود ندارد؛ مقدار منفی ۱ برمی‌گردد
Array.LastIndexOf(a, 10)	۶	اندیس آخرین عنصر برابر ۱۰ (IndexOf از ابتدا و LastIndexOf از انتها جست‌وجو می‌کند)
Array.Sort(a)	10 10 15 18 20 37 50 80	عناصر را به‌صورت صعودی مرتب می‌کند
Array.BinarySearch(a, 10)	۱	روی آرایه‌ی مرتب‌شده‌ی صعودی؛ اندیس یکی از عنصرهای ۱۰ (با وجود تکرار)
Array.Reverse(a)	80 50 37 20 18 15 10 10	ترتیب عناصر را وارونه می‌کند
Array.BinarySearch(a, 10)	۱-	آرایه دیگر صعودی نیست؛ جست‌وجوی دودویی نتیجه‌ی درست نمی‌دهد (عنصر پیدا نمی‌شود)

جست‌وجوی دودویی فقط روی آرایه‌ی مرتب‌شده‌ی صعودی درست کار می‌کند.

ایران توشه
توشه ای برای موفقیت

۱۱۴ صفحه ۹۵ **فعالیت منزل**

می‌خواهیم برنامه‌ای بنویسیم که با استفاده از یک فهرست انتخاب (منو) عملیات زیر را روی آرایه انجام دهد و برنامه تا زمانی که کاربر گزینه‌ی خروج را انتخاب نکرده است، ادامه یابد. گزینه‌های منو به رنگ سفید روی

زمینه‌ی آبی نمایش داده شوند. ۱ دریافت عناصر آرایه ۲ نمایش محتوای عناصر آرایه ۳ مرتب کردن ۴ معکوس کردن ۵ جست‌وجوی خطی ۶ جست‌وجوی دودویی ۷ خروج

```
int[] a = new int[0];
int choice = 0;
while (choice != 7)
{
    Console.BackgroundColor = ConsoleColor.Blue;
    Console.ForegroundColor = ConsoleColor.White;
    Console.Clear();
    Console.WriteLine("1 Input   2 Show   3 Sort   4 Reverse");
    Console.WriteLine("5 Linear Search   6 Binary Search   7 Exit");
    Console.Write("Your choice: ");
    choice = int.Parse(Console.ReadLine());
    switch (choice)
    {
        case 1:
            Console.Write("Number of elements: ");
            a = new int[int.Parse(Console.ReadLine())];
            for (int i = 0; i < a.Length; i++) a[i] = int.Parse(Console.ReadLine());
            break;
        case 2: foreach (int v in a) Console.Write(v + " "); break;
        case 3: Array.Sort(a); break;
        case 4: Array.Reverse(a); break;
        case 5:
            Console.Write("Value: ");
            Console.WriteLine("Index = " + Array.IndexOf(a,
int.Parse(Console.ReadLine())));
            break;
        case 6:
            Console.Write("Value (array must be sorted): ");
            Console.WriteLine("Index = " + Array.BinarySearch(a,
int.Parse(Console.ReadLine())));
            break;
    }
    if (choice != 7) { Console.WriteLine("\nPress Enter..."); Console.ReadLine(); }
}
```

ایران توشه
توشه ای برای موفقیت

۱۱۵ صفحه ۹۶ پروژه فاز ۲

برنامه‌ی کتابخانه‌ی هنرستان را با استفاده از دستوراتی که در این پودمان آموخته‌اید بنویسید. تعداد کتاب‌ها و اعضا را از ورودی دریافت کرده، آرایه‌های متناظر برای آنها اعلان کنید ... منوی ۱ Input book ۲ Input user

۳ Exit ۷ EditUser ۶ Subject Book ۵ Show Book ۴ Show User ... یک نوع داده‌ی شمارشی به نام MenuItem برای تشخیص منوی انتخاب‌شده تعریف کنید.

```
enum MenuItem { InputUser = 1, InputBook, ShowUser, ShowBook, SubjectBook, EditUser,
Exit }
// داخل Main:
Console.Write("Number of members: ");
int u = int.Parse(Console.ReadLine());
Console.Write("Number of books: ");
int b = int.Parse(Console.ReadLine());
string[] firstName = new string[u], lastName = new string[u];
string[] title = new string[b], subject = new string[b];
MenuItem m = 0;
while (m != MenuItem.Exit)
{
    Console.WriteLine("\n1 Input user  2 Input book  3 Show User  4 Show Book");
    Console.WriteLine("5 Subject Book  6 EditUser  7 Exit");
    m = (MenuItem)int.Parse(Console.ReadLine());
    switch (m)
    {
        case MenuItem.InputUser:
            for (int i = 0; i < u; i++)
            { Console.Write("First, Last name of member {0}: ", i);
              firstName[i] = Console.ReadLine(); lastName[i] = Console.ReadLine(); }
            break;
        case MenuItem.InputBook:
            for (int i = 0; i < b; i++)
            { Console.Write("Title, Subject of book {0}: ", i);
              title[i] = Console.ReadLine(); subject[i] = Console.ReadLine(); }
            break;
```

ایران توشه
توشه ای برای موفقیت

```
        case MenuItem.ShowUser:
            for (int i = 0; i < u; i++)
                Console.WriteLine("{0}: {1} {2}", i, firstName[i], lastName[i]);
            break;
        case MenuItem.ShowBook:
            for (int i = 0; i < b; i++)
                Console.WriteLine("{0}: {1} ({2})", i, title[i], subject[i]);
            break;
        case MenuItem.SubjectBook:
            Console.Write("Subject: ");
            string s = Console.ReadLine();
            for (int i = 0; i < b; i++)
                if (subject[i] == s) Console.WriteLine(title[i]);
            break;
        case MenuItem.EditUser:
            Console.Write("Membership code: ");
            int code = int.Parse(Console.ReadLine());
            if (code >= 0 && code < u)
            { firstName[code] = Console.ReadLine(); lastName[code] =
Console.ReadLine(); }
            else Console.WriteLine("Invalid code");
            break;
    }
}
```

اندیس آرایه‌ها همان کد عضویت (و کد کتاب) است.

تکمیل پروژه: مقایسه‌ی کد ۱ و ۲

کد شماره‌ی ۱ (Thread.Sleep(50) + foreach) هر نویسه‌ی رشته‌ی C# Programing را با ۵۰ میلی‌ثانیه توقف جداگانه چاپ می‌کند، بنابراین متن به‌صورت «تایپ‌شونده» و تدریجی دیده می‌شود؛ کد شماره‌ی ۲ کل رشته را یک‌جا و بی‌درنگ چاپ می‌کند. برای زیباسازی پروژه (مثلاً چاپ نام برنامه در ابتدای اجرا) از کدی مشابه کد ۱ استفاده می‌کنیم و برای استفاده از Sleep باید using System.Threading; اضافه شود.

ایران توشه
توشه ای برای موفقیت

گام به گام توسعه برنامه‌سازی و پایگاه داده - پودمان سوم

طراحی واسط گرافیکی

فهرست

۱۱۶ صفحه ۱۰۲ · فعالیت کارگاهی — اجزای IDE پروژه‌های ویندوزی ۸۷

۱۱۷ صفحه ۱۰۴ · کارگاه ۱ — با RightToLeft=Yes چه تغییری می‌کند؟ ۸۸

۱۱۸ صفحه ۱۰۴ · کارگاه ۱ — جدول ۲: BackgroundImageLayout ۸۸

۱۱۹ صفحه ۱۰۵ · فعالیت گروهی — جدول ۳: FormBorderStyle ۸۸

۱۲۰ صفحه ۱۰۵ · فعالیت گروهی — جدول ۴: ویژگی‌های فرم ۸۹

۱۲۱ صفحه ۱۰۵ · فعالیت گروهی — RightToLeftLayout ۹۰

۱۲۲ صفحه ۱۰۶ · کارگاه ۲ — عرض برچسب و RightToLeft برچسب ۹۰

۱۲۳ صفحه ۱۰۷ · کارگاه ۲ — مقادیر TextAlign ۹۰

۱۲۴ صفحه ۱۰۷ · کارگاه ۳ — کادر متن: Name، Text، Ctrl+کشیدن، PasswordChar ۹۱

۱۲۵ صفحه ۱۰۸ · کارگاه ۳ — Visible، Enabled، Cursor، MaxLength ۹۱

۱۲۶ صفحه ۱۰۸ · فعالیت گروهی — جدول ویژگی‌های کنترل‌ها ۹۲

۱۲۷ صفحه ۱۰۹ · فعالیت منزل — فرم مشخصات کاربر ۹۳

ایران توشه
توشه‌ای برای موفقیت

۱۲۸ صفحه ۱۰۹ · فعالیت منزل — ماشین حساب شبیه ویندوز ۱۰ ۹۳

۱۲۹ صفحه ۱۱۰ · فعالیت کارگاهی — خروج از پروژه Login ۹۴

۱۳۰ صفحه ۱۱۲ · کارگاه ۴ — خروج با this.Close ۹۴

۱۳۱ صفحه ۱۱۳ · کارگاه ۴ — پاک‌کردن کادرها و Parse ۹۴

۱۳۲ صفحه ۱۱۳ · کارگاه ۴ — رویداد کلیک ضرب، تفریق و تقسیم ۹۵

۱۳۳ صفحه ۱۱۴ · کارگاه ۵ — افزودن تصویر به منابع ۹۵

۱۳۴ صفحه ۱۱۵ · فعالیت کارگاهی — گزینه Add New Image ۹۶

۱۳۵ صفحه ۱۱۶ · کارگاه ۶ — Import تصویر: Local resource و Project resource file ۹۶

۱۳۶ صفحه ۱۱۶ · کارگاه ۶ — SizeMode و ویژگی Image ۹۶

۱۳۷ صفحه ۱۱۶ · کارگاه ۶ — دکمه‌های نمایش تصویر ۱ و ۲ ۹۷

۱۳۸ صفحه ۱۱۷ · فعالیت منزل — کادر تصویر در فرم‌ها ۹۷

۱۳۹ صفحه ۱۱۷ · کارگاه ۷ — بزرگ‌نمایی و محل تصویر ۹۸

۱۴۰ صفحه ۱۱۸ · کارگاه ۷ — Left و Top و بزرگ/کوچک‌نمایی ۹۸

۱۴۱ صفحه ۱۱۸ · کارگاه ۷ — ImageLocation ۹۸

۱۴۲ صفحه ۱۱۹ · کارگاه ۷ — پیمایش تصاویر: if و پوشه Debug ۹۸

۱۴۳ صفحه ۱۱۹ · کارگاه ۷ — دکمه‌های قبلی و بعدی ۹۹

۱۴۴ صفحه ۱۱۹ · فعالیت کارگاهی — Show و Hide ۹۹

۱۴۵ صفحه ۱۱۹ · فعالیت منزل — بازی ساده: حرکت آدمک ۱۰۰

ایران توشه
توشه ای برای موفقیت

۱۴۶ صفحه ۱۲۱ · کارگاه ۸ — OpenFileDialog: پرونده غیرتصویری ۱۰۰
۱۴۷ صفحه ۱۲۲ · کارگاه ۸ — انتخاب‌نکردن پرونده و DialogResult ۱۰۰
۱۴۸ صفحه ۱۲۲ · کارگاه ۹ — پروژه PictureCity: فرم و انتخاب عکس ۱۰۱
۱۴۹ صفحه ۱۲۳ · کارگاه ۹ — عنوان کادر محاوره‌ای (Title) ۱۰۲
۱۵۰ صفحه ۱۲۴ · فعالیت گروهی — جدول ۵: پارامترهای MessageBox.Show و شمارشی‌ها ۱۰۲
۱۵۱ صفحه ۱۲۴ · کارگاه ۱۰ — اجرای پیام خطای نام ۱۰۳
۱۵۲ صفحه ۱۲۵ · کارگاه ۱۰ — پیام خطا برای نام خانوادگی؛ حذف عنوان ۱۰۳
۱۵۳ صفحه ۱۲۵ · کارگاه ۱۰ — پاک‌کردن نام با Retry ۱۰۳
۱۵۴ صفحه ۱۲۵ · فعالیت کارگاهی — OK/Cancel با دکمه پیش‌فرض Cancel ۱۰۴
۱۵۵ صفحه ۱۲۵ · فعالیت منزل — فرم ورود کاربر با پیام ۱۰۵
۱۵۶ صفحه ۱۲۹ · کارگاه ۱ — زمان‌سنج و بازی پک‌من ۱۰۵
۱۵۷ صفحه ۱۳۰ · کارگاه ۱ — حرکت خودرو با زمان‌سنج (MoveCar) ۱۰۵
۱۵۸ صفحه ۱۳۰ · فعالیت منزل — توقف در لبه‌ی فرم و حرکت به عقب ۱۰۶
۱۵۹ صفحه ۱۳۱ · کارگاه ۲ — زمان‌سنج ساده (Chronometer) ۱۰۷
۱۶۰ صفحه ۱۳۱ · فعالیت منزل — نمایش ساعت سیستم ۱۰۸
۱۶۱ صفحه ۱۳۳ · فعالیت گروهی — جدول ۶: ویژگی‌های RadioButton و CheckBox ۱۰۸
۱۶۲ صفحه ۱۳۴ · کارگاه ۳ — نمایش گذرواژه و نوع کاربر (Login) ۱۰۹
۱۶۳ صفحه ۱۳۵ · فعالیت گروهی — انتخاب هم‌زمان دو دکمه‌ی رادیویی ۱۱۰

ایران توشه
توشه‌ای برای موفقیت

۱۶۴ صفحه ۱۳۶ · کارگاه ۴ — کنترل کادر گروهی (Tour): ویژگی‌ها ۱۱۰

۱۶۵ صفحه ۱۳۶ · کارگاه ۴ — رویداد CheckedChanged کادرهای علامت ۱۱۱

۱۶۶ صفحه ۱۳۷ · کارگاه ۴ — جدول ۷: شکل‌های متد Next و تغییر تصویر ۱۱۱

۱۶۷ صفحه ۱۳۸ · کارگاه ۴ — دکمه‌ی پاسخ صحیح (switch) ۱۱۲

۱۶۸ صفحه ۱۳۸ · فعالیت گروهی — تغییر برنامه‌ی Tour با کلیک برچسب ۱۱۳

۱۶۹ صفحه ۱۳۸ · فعالیت گروهی — جدول کنترل‌ها، ویژگی‌ها و رویدادها ۱۱۳

۱۷۰ صفحه ۱۴۰ · کارگاه ۵ — کنترل سربرگ (PersonalInfo) ۱۱۴

۱۷۱ صفحه ۱۴۲ · کارگاه ۶ — پروژه‌ی چندفرمی: فرم شروع و فیلدها ۱۱۴

۱۷۲ صفحه ۱۴۴ · کارگاه ۶ — رویداد کلیک ورود و رویداد Load فرم صفحه‌ی شخصی ۱۱۵

۱۷۳ صفحه ۱۴۵ · فعالیت کارگاهی — نمایش فرم: Show یا ShowDialog ۱۱۶

۱۷۴ صفحه ۱۴۵ · فعالیت منزل — قفل ۳۰ ثانیه‌ای پس از سه بار ورود اشتباه ۱۱۷

۱۷۵ صفحه ۱۴۵ · پژوهش — معایب سطح دسترسی public و انتقال داده بین فرم‌ها ۱۱۸

۱۷۶ صفحه ۱۴۶ · فعالیت گروهی — کادر فهرست یا کادر ترکیبی ۱۱۹

۱۷۷ صفحه ۱۴۸ · کارگاه ۷ — انتخابات شورای دانش‌آموزی (StudentElection) ۱۱۹

۱۷۸ صفحه ۱۴۸ · فعالیت کارگاهی — مرتب‌سازی کادر ترکیبی و نمایش اندیس انتخاب‌شده ۱۲۰

۱۷۹ صفحه ۱۴۹ · پروژه فاز ۳ — واسط گرافیکی برنامه‌ی کتابخانه‌ی هنرستان ۱۲۰

ایران توشه
توشه‌ای برای موفقیت

۱۱۶ 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.

ایران توشه
توشه‌ای برای موفقیت

۱۱۷ 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.

۱۱۸ 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

۱۱۹ 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.

۱۲۰ 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

۱۲۱ 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.

۱۲۲ 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.

۱۲۳ 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.

۱۲۴ 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.

۱۲۵ 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.

۱۲۶ 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

۱۲۷ 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

۱۲۸ 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.

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

۱۲۹ 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.

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

۱۳۰ 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.

۱۳۱ 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.

۱۳۲ 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.

۱۳۳ 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

۱۳۴ 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 (PNG، BMP، GIF، JPEG 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.

۱۳۵ 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

۱۳۶ 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.

۱۳۷ 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.

۱۳۸ 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

۱۳۹ 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.

۱۴۰ 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

۱۴۱ 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).

۱۴۲ 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.

۱۴۳ 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.

۱۴۴ 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).

۱۴۵ 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

۱۴۶ 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.

۱۴۷ 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).

۱۴۸ 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

۱۴۹ 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.

۱۵۰ 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

ایران توشه
توشه‌ای برای موفقیت

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

۱۵۱ 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.

۱۵۲ 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.

۱۵۳ 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).

۱۵۴ 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

۱۵۵ 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();
}
```

۱۵۶ 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

۱۵۷ 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

۱۵۸ 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.

۱۵۹ 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.

ایران توشه
توشه‌ای برای موفقیت

```
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;
    }
}
```

۱۶۰ 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();
}
```

۱۶۱ 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

۱۶۲ 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

۱۶۳ 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.

۱۶۴ 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

۱۶۵ 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;.

۱۶۶ 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
}
```

۱۶۷ 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

۱۶۸ 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;
}
```

۱۶۹ 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، Enabled ،Text	CheckedChanged، Click
کادر علامت (CheckBox)	انتخاب یک یا چند گزینه (علامت‌گذاری)	CheckAlign، Checked، Enabled ،Text	CheckedChanged، Click
کادر گروهی (GroupBox)	گروه‌بندی و دسته‌بندی کنترل‌ها با عنوان	Enabled، Font، RightToLeft ،Text	Click ،Leave ،Enter
پنل (Panel)	گروه‌بندی کنترل‌ها بدون عنوان (با امکان نوار پیمایش)	AutoScroll، BorderStyle، Enabled ،BackColor	Enter ،Click ،Paint

ایران توشه
توشه ای برای موفقیت

۱۷۰ 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

۱۷۱ 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.

۱۷۲ 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 = "کاربر گرامی " + frm.userName + " خوش آمدید";
    }
    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

۱۷۳ 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.

۱۷۴ 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;
}
```

۱۷۵ 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

۱۷۶ 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).

۱۷۷ 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).

۱۷۸ 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();
}
```

۱۷۹ 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

گام به گام توسعه برنامه‌سازی و پایگاه داده - پودمان چهارم

توسعه‌ی واسط گرافیکی

فهرست

۱۸۰ صفحه ۱۵۵ · فعالیت گروهی — عملکرد ماوس در ویندوز و برنامه‌های کاربردی ۱۲۸

۱۸۱ صفحه ۱۵۶ · فعالیت گروهی — جدول ۱: رویدادهای ماوس وابسته به دکمه ۱۲۸

۱۸۲ صفحه ۱۵۶ · کارگاه ۱ — واکنش برنامه به رویدادهای ماوس (MouseEvent) ۱۲۹

۱۸۳ صفحه ۱۵۸ · کارگاه ۲ — MouseDown و MouseUp (ParkCar) ۱۳۰

۱۸۴ صفحه ۱۵۹ · فعالیت منزل — حرکت ماشین به نسبت زمان نگه‌داشتن دکمه ۱۳۱

۱۸۵ صفحه ۱۵۹ · کنجکاوی — تغییر شکل نمادها با حرکت ماوس ۱۳۲

۱۸۶ صفحه ۱۶۰ · کارگاه ۳ — MouseEnter و MouseLeave (EnglishLearning) ۱۳۲

۱۸۷ صفحه ۱۶۱ · فعالیت کارگاهی — جدول ۲: رویدادهای ماوس مستقل از دکمه ۱۳۳

۱۸۸ صفحه ۱۶۲ · فعالیت منزل — دکمه‌ی گرافیکی سه‌حالته و آموزش واژگان فارسی ۱۳۳

۱۸۹ صفحه ۱۶۳ · کارگاه ۴ — آرگومان‌های رویدادهای ماوس (MouseSimulation) ۱۳۴

۱۹۰ صفحه ۱۶۴ · فعالیت کارگاهی — نمایش واژه‌ی Computer و مختصات ماوس ۱۳۵

۱۹۱ صفحه ۱۶۴ · فعالیت منزل — شبیه‌ساز ماوس: راست‌دست و چپ‌دست ۱۳۶

ایران توشه
توشه‌ای برای موفقیت

۱۹۲ صفحه ۱۶۴ · فعالیت کارگاهی — جدول ۳: ویژگی‌های آرگومان e ۱۳۶

۱۹۳ صفحه ۱۶۵ · پژوهش — تخصیص متد رویداد با برنامه‌نویسی و دستور as ۱۳۷

۱۹۴ صفحه ۱۶۵ · فعالیت منزل — یک متد رویداد برای همه‌ی برچسب‌ها ۱۳۷

۱۹۵ صفحه ۱۶۵ · فیلم — بازی سرعت کلیک ۱۳۸

۱۹۶ صفحه ۱۶۶ · فیلم — بازی اعصاب‌سنج ۱۳۹

۱۹۷ صفحه ۱۶۷ · کنجکاوی — تفاوت منوها ۱۴۰

۱۹۸ صفحه ۱۶۸ · فعالیت گروهی — منوها و کلیدهای میان‌بر در ویندوز ۱۴۰

۱۹۹ صفحه ۱۶۹ · کارگاه ۶ — ایجاد منو با MenuStrip (MenuBar) ۱۴۱

۲۰۰ صفحه ۱۷۰ · کارگاه ۷ — سفارشی کردن منو ۱۴۲

۲۰۱ صفحه ۱۷۰ · فعالیت منزل — طراحی منوی Notepad ۱۴۳

۲۰۲ صفحه ۱۷۱ · فعالیت کارگاهی — جدول ۴: ویژگی‌های کنترل منو ۱۴۳

۲۰۳ صفحه ۱۷۱ · کنجکاوی — جهت نمایش متن گزینه‌های منو ۱۴۴

۲۰۴ صفحه ۱۷۲ · پژوهش — رویدادهای دیگر منو ۱۴۴

۲۰۵ صفحه ۱۷۲ · کارگاه ۸ — رویداد برای گزینه‌های منو (TextEditor) ۱۴۵

۲۰۶ صفحه ۱۷۳ · فعالیت کارگاهی — منوی استاندارد ۱۴۷

۲۰۷ صفحه ۱۷۴ · کارگاه ۹ — طراحی منوی زمینه (ContextMenuStrip) ۱۴۷

۲۰۸ صفحه ۱۷۵ · کارگاه ۱۰ — پروژه‌ی منودار چندرسانه‌ای (MyIran) ۱۴۸

۲۰۹ صفحه ۱۸۰ · فعالیت کارگاهی — عملکرد صفحه‌کلید در ویندوز ۱۵۰

ایران توشه
توشه‌ای برای موفقیت

۲۱۰ صفحه ۱۸۱ · کارگاه ۱ — رویدادهای صفحه‌کلید برای فرم (KeyBoardEvent1) ۱۵۱

۲۱۱ صفحه ۱۸۲ · کارگاه ۲ — ترتیب رخداد رویدادهای صفحه‌کلید (KeyboardEvent2) ۱۵۲

۲۱۲ صفحه ۱۸۴ · کارگاه ۳ — رویداد KeyDown و KeyUp (ShootBall) ۱۵۲

۲۱۳ صفحه ۱۸۵ · کارگاه ۴ — آرگومان e در KeyDown و KeyPress (AsciiCode) ۱۵۳

۲۱۴ صفحه ۱۸۶ · کارگاه ۴ — ویژگی KeyData و رویداد KeyPress ۱۵۴

۲۱۵ صفحه ۱۸۷ · کنجکاوی — جدول ۶ و ویژگی‌های آرگومان e در KeyPress ۱۵۵

۲۱۶ صفحه ۱۸۷ · فعالیت منزل — کد حروف فارسی ۱۵۶

۲۱۷ صفحه ۱۸۹ · کارگاه ۵ — کلیدهای ترکیبی (ZoomPicture) ۱۵۶

۲۱۸ صفحه ۱۹۰ · فعالیت منزل — برنامه‌ی کارگاه ۵ با KeyData ۱۵۷

۲۱۹ صفحه ۱۹۰ · کارگاه ۶ — فوکوس (Focus) و KeyPreview ۱۵۸

۲۲۰ صفحه ۱۹۱ · فعالیت منزل — فیلم: برنامه‌ی یافتن نویسه در صفحه‌کلید ۱۵۸

۲۲۱ صفحه ۱۹۲ · کارگاه ۷ — نوع داده‌ی شمارشی Keys (MoveSpacecraft) ۱۵۹

۲۲۲ صفحه ۱۹۳ · کارگاه ۸ — انتقال فوکوس با کلید Enter (FocusByEnter) ۱۶۰

۲۲۳ صفحه ۱۹۴ · پژوهش — تغییر رنگ زمینه‌ی کادر متن با فوکوس ۱۶۱

۲۲۴ صفحه ۱۹۵ · کارگاه ۹ — متدهای نوع داده‌ی char (FilterInputData) ۱۶۱

۲۲۵ صفحه ۱۹۶ · کارگاه ۹ — فیلتر حروف با KeyDown و KeyPress ۱۶۲

۲۲۶ صفحه ۱۹۷ · فعالیت کارگاهی — جدول ۷: متدهای نوع داده‌ی char ۱۶۳

۲۲۷ صفحه ۱۹۸ · کنجکاوی — کلاس String ۱۶۳

ایران توشه
توشه‌ای برای موفقیت

۲۲۸ صفحه ۱۹۹ · کارگاه ۱۰ — بررسی متدهای رشته‌ای (StringMethod) ۱۶۴

۲۲۹ صفحه ۲۰۰ · کارگاه ۱۱ — متدهای CompareTo و Trim (LoginApp) ۱۶۵

۲۳۰ صفحه ۲۰۱ · فعالیت کارگاهی — جدول ۸: شرح متدهای رشته‌ای ۱۶۶

۲۳۱ صفحه ۲۰۱ · فیلم — برنامه‌ی جست‌وجو و جایگزینی واژه (کارگاه ۱۲) ۱۶۸

۲۳۲ صفحه ۲۰۲ · پروژه فاز ۴ — منو برای برنامه‌ی کتابخانه ۱۶۹

ایران توشه
توشه ای برای موفقیت

۱۸۰ صفحه ۱۵۵ فعالیت گروهی

برای بررسی عملکرد ماوس در ویندوز و برنامه‌های کاربردی به پرسش‌های زیر پاسخ دهید:

برای اجرای یک برنامه در نوار وظیفه چگونه از ماوس استفاده می‌کنید؟

برای اجرای یک برنامه از روی میزکار چگونه از ماوس استفاده می‌کنید؟

برای ایجاد نسخه‌ی مشابه یک پرونده یا پوشه چگونه از ماوس استفاده می‌کنید؟

وقتی که اشاره‌گر ماوس روی نمادهای میزکار قرار می‌گیرد، آیا شکل نماد تغییر می‌کند؟

در برنامه‌ی نقاشی ویندوز برای ترسیم شکل چگونه از ماوس استفاده می‌کنید؟

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

- نوار وظیفه: یک بار کلیک چپ روی نماد برنامه.
- میزکار: دوبار کلیک چپ (دابل‌کلیک) روی نماد برنامه.
- نسخه‌ی مشابه: نگه داشتن کلید Ctrl و کشیدن (Drag) پرونده یا پوشه با ماوس، یا راست‌کلیک ← Copy و سپس Paste.
- بله؛ با قرار گرفتن اشاره‌گر روی نماد، ظاهر نماد (پس‌زمینه‌ی آن) تغییر می‌کند و روشن یا برجسته می‌شود.
- در نقاشی: با نگه داشتن دکمه‌ی چپ و حرکت دادن ماوس (کشیدن) شکل ترسیم می‌شود.

کاربردهای ماوس

کاربرد ماوس در ویندوز	کاربرد ماوس در برنامه‌های کاربردی
انتخاب و اجرای نمادها و پنجره‌ها با کلیک و دابل‌کلیک	انتخاب ابزار و ترسیم شکل (کشیدن) در برنامه‌ی نقاشی
جابه‌جایی پنجره‌ها و تغییر اندازه‌ی آنها (کشیدن)	انتخاب متن در برنامه‌ی واژه‌پرداز (کشیدن یا دابل‌کلیک)
باز کردن منوی میان‌بر با راست‌کلیک	کلیک روی دکمه‌ها و گزینه‌های منو در نرم‌افزار
نمایش راهنمای ابزار (ToolTip) با نگه داشتن اشاره‌گر روی نماد	هدف‌گیری و شلیک در بازی‌های رایانه‌ای
تغییر شکل اشاره‌گر (دست، ساعت شنی، ...)	بزرگ‌نمایی و جابه‌جایی با چرخ ماوس

۱۸۱ صفحه ۱۵۶ فعالیت گروهی

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

ایران توشه
توشه ای برای موفقیت

با انجام عملیات زیر به‌وسیله‌ی ماوس چه رویدادی رخ می‌دهد؟ (نگه‌داشتن دکمه‌ی راست یا چپ؛ رها کردن دکمه‌ی راست یا چپ)

کدام رویداد ماوس وابسته به دکمه‌ی دیگری را قبلاً استفاده کرده‌اید؟

جدول ۱

نام رویداد	شرح رویداد
MouseClick	زمانی رخ می‌دهد که روی کنترل با ماوس کلیک شود.
MouseDoubleClick	زمانی رخ می‌دهد که روی کنترل با ماوس دابل‌کلیک (دو بار پشت سر هم کلیک) شود.
MouseDown	زمانی رخ می‌دهد که اشاره‌گر ماوس روی کنترل است و یکی از دکمه‌های ماوس فشرده می‌شود.
MouseUp	زمانی رخ می‌دهد که اشاره‌گر ماوس روی کنترل است و یکی از دکمه‌های ماوس رها می‌شود.

عملیات و رویداد

عملیات	رویداد
نگه‌داشتن (فشردن) دکمه‌ی راست یا چپ	MouseDown
رها کردن دکمه‌ی راست یا چپ	MouseUp

رویدادهای Click (برای دکمه‌ها) و MouseClick پیش‌تر استفاده شده‌اند.

ایران توشه
توشه‌ای برای موفقیت

۱۸۲ صفحه ۱۵۶ کارگاه ۱

۳ ـ دستوری در رویداد MouseClick بنویسید که رشته‌ی «به سی‌شارپ خوش‌آمدید» را در کادر پیام نمایش دهد. برنامه را اجرا کنید و روی دکمه کلیک کنید.

۴ ـ برای خروج از برنامه می‌توانید از متد Application.Exit() استفاده کنید. آیا روش دیگری می‌شناسید؟

۵ ـ در متد رویداد MouseDown دکمه‌ی btnRun کد زیر را بنویسید ... با نگه داشتن دکمه‌ی ماوس، رنگ دکمه‌ی btnRun چه تغییری می‌کند؟

۶ ـ در متد رویداد MouseUp دکمه‌ی btnRun رنگ دکمه را به حالت اول برگردانید.

```
private void btnRun_MouseClick(object sender, MouseEventArgs e)
{
    MessageBox.Show("به سی‌شارپ خوش‌آمدید");
}
private void btnRun_MouseDown(object sender, MouseEventArgs e)
{
    btnRun.BackColor = Color.Red;     // با نگه داشتن دکمه‌ی ماوس، رنگ دکمه قرمز می‌شود
}
private void btnRun_MouseUp(object sender, MouseEventArgs e)
{
    btnRun.BackColor = Color.Blue;    // بازگشت به رنگ زمینه‌ی اولیه (آبی)
}
private void Form1_DoubleClick(object sender, EventArgs e)
{
    Application.Exit();               // خروج با دابل‌کلیک روی فرم
}
```

روش دیگر خروج از برنامه استفاده از this.Close() (بستن فرم اصلی) است؛ Environment.Exit(0) هم برنامه را به پایان می‌رساند.

۱۸۳ صفحه ۱۵۸ کارگاه ۲

چرا در این برنامه نیاز به زمان‌سنج داریم؟

ویژگی Right چه مقداری را نگهداری می‌کند؟

برنامه‌ی پارک ماشین را اجرا کنید. با چه شرایطی، کاربر برنده می‌شود؟

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

حرکت ماشین باید تا زمانی که دکمه‌ی ماوس نگه داشته شده است به‌طور مداوم و تدریجی انجام شود. رویداد MouseDown فقط یک بار رخ می‌دهد؛ پس برای تکرار حرکت در فواصل زمانی کوتاه به زمان‌سنج نیاز است که با MouseDown فعال و با MouseUp غیرفعال می‌شود.

ایران توشه
توشه‌ای برای موفقیت

```
private void timer1_Tick(object sender, EventArgs e)
{
    picCar.Left += 5;
}
private void Form1_MouseDown(object sender, MouseEventArgs e)
{
    timer1.Enabled = true;
}
private void Form1_MouseUp(object sender, MouseEventArgs e)
{
    timer1.Enabled = false;
    if ((picCar.Right + 10 > picBarrier.Left) && (picCar.Right < picBarrier.Left))
        MessageBox.Show("آفرین، برنده شدی");
    else
        MessageBox.Show("متأسفانه، بازنده شدی");
}
```

ویژگی Right مختصات افقی لبه‌ی راست کنترل را (فاصله‌ی لبه‌ی راست کادر تصویر از سمت چپ فرم) نگه می‌دارد و برابر Left + Width است.

شرط برنده شدن: لبه‌ی راست ماشین باید کمتر از ۱۰ پیکسل با مانع فاصله داشته باشد و به مانع نرسیده باشد: picBarrier.Left – 10 < picCar.Right < picBarrier.Left.

برای شروع بازی جدید، مکان اولیه‌ی ماشین را بازمی‌گردانیم (مثلاً با یک دکمه‌ی «بازی جدید» یا پس از نمایش پیام):

```
private void btnNewGame_Click(object sender, EventArgs e)
{
    picCar.Left = 20;      // مکان اولیه‌ی ماشین
    timer1.Enabled = false;
}
```

۱۸۴ صفحه ۱۵۹ **فعالیت منزل**

بازی کارگاه ۲ را به‌صورت زیر تغییر دهید: بعد از رها کردن ماوس، ماشین به نسبت زمان نگه‌داشتن دکمه‌ی ماوس، حرکت کند. به‌طور مثال اگر کاربر یک ثانیه دکمه‌ی ماوس را نگه دارد، ماشین ۱۰۰ پیکسل حرکت کند.

با زمان‌سنج با Interval = 100، شمارنده‌ی ticks را در زمان نگه‌داشتن افزایش می‌دهیم (هر ثانیه ۱۰ تیک)؛ پس از رها کردن ماوس ماشین به اندازه‌ی ticks * 10 پیکسل (یک ثانیه ۱۰۰ پیکسل) حرکت می‌کند:

```
int ticks = 0;
private void timer1_Tick(object sender, EventArgs e)   // Interval = 100
{
    ticks++;
}
private void Form1_MouseDown(object sender, MouseEventArgs e)
{
    ticks = 0;
    timer1.Enabled = true;
}
private void Form1_MouseUp(object sender, MouseEventArgs e)
{
    timer1.Enabled = false;
    picCar.Left += ticks * 10;     // ۱۰ پیکسل = ۱۰۰ ثانیه = ۱ تیک
}
```

۱۸۵ صفحه ۱۵۹ کنجکاوی

با حرکت ماوس روی نمادهای برنامه در میزکار و نوار وظیفه چه تغییراتی در شکل نمادها مشاهده می‌کنید؟

لحظه‌ای ماوس را بدون آنکه کلیک کنید، روی یکی از نمادهای نوار وظیفه نگه دارید. چه مشاهده می‌کنید؟

با حرکت ماوس روی نماد، پس‌زمینه‌ی نماد برجسته (روشن یا رنگی) می‌شود. اگر ماوس چند لحظه روی نماد نوار وظیفه بماند، پیش‌نمایش پنجره‌ی برنامه یا متن راهنما (نام برنامه) نمایش داده می‌شود. اینها نمونه‌ی رویدادهای ماوس مستقل از دکمه (MouseEnter، MouseLeave و MouseHover) هستند.

۱۸۶ صفحه ۱۶۰ کارگاه ۳

چرا ویژگی Visible کادرتصویرها برابر با false است؟

۴ ـ در متد رویداد MouseLeave_lblBook دستوری بنویسید که picBook پنهان شود و رنگ lblBook مشکی شود.

۵ ـ متد رویداد MouseHover_lblBook را ایجاد کنید ... برنامه را اجرا کنید. عملکرد متد SetToolTip چیست؟

۶ ـ برای آموزش همه‌ی واژگان، برنامه را کامل کنید.

تصویر هر واژه باید در شروع برنامه دیده نشود و فقط وقتی اشاره‌گر ماوس روی واژه‌ی مربوط می‌آید نمایش داده شود؛ برای همین Visible = false است و در MouseEnter برابر true می‌شود.

ایران توشه
توشه ای برای موفقیت

```
private void lblBook_MouseEnter(object sender, EventArgs e)
{
    lblBook.ForeColor = Color.Red;
    picBook.Visible = true;
    lblBook.Cursor = Cursors.Hand;
}
private void lblBook_MouseLeave(object sender, EventArgs e)
{
    picBook.Visible = false;
    lblBook.ForeColor = Color.Black;
}
private void lblBook_MouseHover(object sender, EventArgs e)
{
    ToolTip tip = new ToolTip();
    tip.SetToolTip(this.lblBook, "کتاب");
}
// نیز سه متد مشابه (ساعت) lblClock و (قلم) lblPen، (ماشین) lblCar، (سیب) lblApple برای
نوشته می‌شود.
```

متد SetToolTip(control, text) متن راهنمای (ToolTip) را به کنترل مشخص‌شده نسبت می‌دهد تا وقتی ماوس چند لحظه روی آن ماند، متن در یک کادر مستطیلی کوچک (اینجا معنی فارسی واژه) نمایش داده شود.

۱۸۷ صفحه ۱۶۱ فعالیت کارگاهی

ستون شرح رویدادها در جدول ۲ را تکمیل کنید.

جدول ۲

نام رویداد	شرح رویداد
MouseEnter	زمانی رخ می‌دهد که ماوس به بخشی از کنترل (قسمت قابل‌مشاهده‌ی آن) وارد شود.
MouseLeave	زمانی رخ می‌دهد که اشاره‌گر ماوس از کنترل خارج شود.
MouseMove	زمانی رخ می‌دهد که اشاره‌گر ماوس روی کنترل حرکت داده شود.
MouseHover	زمانی رخ می‌دهد که اشاره‌گر ماوس برای چند لحظه روی کنترل ساکن بماند.

۱۸۸ صفحه ۱۶۲ فعالیت منزل

ـ در برنامه‌های کاربردی برخی از دکمه‌های گرافیکی سه حالت up، down و over دارند. به کمک کنترل PictureBox یک دکمه‌ی گرافیکی سه‌حالته طراحی کنید.

ـ برنامه‌ی آموزشی کارگاه ۳ را برای آموزش واژگان فارسی استفاده کنید. کاربر با ورود ماوس به روی تصویر، واژه‌ی فارسی معادل آن را ببیند.

سه تصویر هم‌اندازه (حالت عادی، روی ماوس، فشرده) به‌صورت فایل‌های up.png، over.png و down.png ساخته و در پوشه‌ی Debug قرار می‌دهیم. حالت‌ها را با رویدادهای ماوس تغییر می‌دهیم:

```
private void picButton_MouseEnter(object sender, EventArgs e)
{ picButton.ImageLocation = "over.png"; }
private void picButton_MouseLeave(object sender, EventArgs e)
{ picButton.ImageLocation = "up.png"; }
private void picButton_MouseDown(object sender, MouseEventArgs e)
{ picButton.ImageLocation = "down.png"; }
private void picButton_MouseUp(object sender, MouseEventArgs e)
{ picButton.ImageLocation = "over.png"; }
```

برای واژگان فارسی، رویداد MouseEnter هر کادر تصویر، برچسب فارسی همان واژه را نمایش می‌دهد و MouseLeave آن را پنهان می‌کند:

```
private void picBook_MouseEnter(object sender, EventArgs e)
{
    lblPersian.Text = "کتاب";
    lblPersian.Visible = true;
}
private void picBook_MouseLeave(object sender, EventArgs e)
{
    lblPersian.Visible = false;
}
```

۱۸۹ صفحه ۱۶۳ کارگاه ۴

۴ ـ قطعه کدی بنویسید که با رها کردن دکمه‌ی ماوس، رنگ دکمه‌ها مشکی شود.

ایران توشه
توشه ای برای موفقیت

۶ ـ کد مرحله‌ی ۵ را تغییر دهید تا به‌جای تصویر اشاره‌گر، شکل نمادین ماوس حرکت کند.

```
private void pnlMousePanel_MouseDown(object sender, MouseEventArgs e)
{
    if (e.Button == MouseButtons.Left)
        picLeft.BackColor = Color.White;
    else if (e.Button == MouseButtons.Middle)
        picMiddle.BackColor = Color.White;
    else if (e.Button == MouseButtons.Right)
        picRight.BackColor = Color.White;
}
private void pnlMousePanel_MouseUp(object sender, MouseEventArgs e)
{
    picLeft.BackColor = Color.Black;
    picMiddle.BackColor = Color.Black;
    picRight.BackColor = Color.Black;
}
private void pnlMousePanel_MouseMove(object sender, MouseEventArgs e)
{
    picCursor.Location = e.Location;    // حرکت تصویر اشاره‌گر
}
```

برای مرحله‌ی ۶ (حرکت «شکل نمادین ماوس» به‌جای تصویر اشاره‌گر) کافی است picCursor را به‌جای تصویر اشاره‌گر با تصویر کوچک شکل ماوس مقداردهی کنیم (ویژگی Image) و همان دستور picCursor.Location = e.Location را نگه داریم.

۱۹۰ صفحه ۱۶۴ **فعالیت کارگاهی**

ـ برنامه‌ای بنویسید که با حرکت ماوس در فضای فرم واژه‌ی Computer به همراه اشاره‌گر ماوس نمایش داده شود.

ـ برنامه را به‌گونه‌ای ویرایش کنید که به‌جای واژه‌ی Computer مختصات ماوس به‌صورت (x,y) به همراه اشاره‌گر ماوس نمایش داده شود.

ایران توشه
توشه ای برای موفقیت

```
private void Form1_MouseMove(object sender, MouseEventArgs e)
{
    lblText.Location = e.Location;        // lblText.Text = "Computer"
}

// حالت دوم: نمایش مختصات
private void Form1_MouseMove(object sender, MouseEventArgs e)
{
    lblText.Text = "(" + e.X + "," + e.Y + ")";
    lblText.Location = e.Location;
}
```

۱۹۱ صفحه ۱۶۴ فعالیت منزل

برنامه‌ی شبیه‌ساز را اجرا کنید. با راست‌کلیک، کلیک چپ، کلیک وسط، دابل‌کلیک و حرکت ماوس روی پنل ماوس، برنامه را آزمایش کنید.

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

با کلیک چپ، دکمه‌ی picLeft، با کلیک وسط picMiddle و با راست‌کلیک picRight سفید می‌شود و با رها کردن مشکی برمی‌گردد؛ با حرکت ماوس تصویر اشاره‌گر همراه ماوس حرکت می‌کند.

یک کادر علامت chkLeftHanded اضافه می‌کنیم. در حالت چپ‌دست، نقش دکمه‌ها جابه‌جا می‌شود:

```
private void pnlMousePanel_MouseDown(object sender, MouseEventArgs e)
{
    MouseButtons primary = chkLeftHanded.Checked ? MouseButtons.Right :
MouseButtons.Left;
    MouseButtons secondary = chkLeftHanded.Checked ? MouseButtons.Left :
MouseButtons.Right;
    if (e.Button == primary)
        picLeft.BackColor = Color.White;
    else if (e.Button == MouseButtons.Middle)
        picMiddle.BackColor = Color.White;
    else if (e.Button == secondary)
        picRight.BackColor = Color.White;
}
```

۱۹۲ صفحه ۱۶۴ فعالیت کارگاهی

در VS با قرار دادن اشاره‌گر ماوس روی هر ویژگی، نوع و عملکرد آن به‌صورت راهنما در اختیار برنامه‌نویس قرار می‌گیرد (شکل ۶). با توجه به این راهنما جدول ۳ را تکمیل کنید.

ایران توشه
توشه ای برای موفقیت

جدول ۳ـ ویژگی‌های آرگومان e

نام ویژگی	نوع داده	شرح
Button	نوع داده‌ی شمارشی MouseButtons	مشخص می‌کند کدام دکمه‌ی ماوس فشار داده شده است (راست، چپ، وسط).
Location	ساختار Point	موقعیت اشاره‌گر ماوس روی کنترل را مشخص می‌کند.
X	int	مختصات افقی (x) اشاره‌گر ماوس هنگام رخداد رویداد را برمی‌گرداند.
Y	int	مختصات عمودی (y) اشاره‌گر ماوس هنگام رخداد رویداد را برمی‌گرداند.
Clicks	int	تعداد دفعاتی که دکمه فشرده و رها شده است (مثلاً ۲ برای دابل‌کلیک).

۱۹۳ صفحه ۱۶۵ پژوهش

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

ـ کاربرد دستور روبه‌رو را بررسی کنید: Button btn = sender as Button;

با عملگر += می‌توان یک متد را به رویداد کنترل متصل کرد (معمولاً در سازنده‌ی فرم یا رویداد Load):

```
button1.Click += new EventHandler(MyClickMethod);   // یا ساده‌تر: button1.Click +=
MyClickMethod;
void MyClickMethod(object sender, EventArgs e) { MessageBox.Show("Clicked"); }
```

دستور Button btn = sender as Button; آرگومان sender (از نوع object) را به نوع Button تبدیل می‌کند تا بتوان به ویژگی‌های دکمه‌ای که رویداد برای آن رخ داده دسترسی داشت؛ تفاوت آن با (Button)sender این است که اگر تبدیل ممکن نباشد خطا نمی‌دهد و مقدار null برمی‌گرداند.

ایران توشه
توشه ای برای موفقیت

۱۹۴ صفحه ۱۶۵ فعالیت منزل

در برنامه‌ی کارگاه ۴ واحد یادگیری ۶ کد متد رویداد کلیک همه‌ی برچسب‌ها مشابه است. بنابراین می‌توانیم به‌جای ۹ متد رویداد کلیک برای برچسب‌ها، یک متد رویداد کلیک برای همه‌ی برچسب‌ها بنویسیم. ... کد زیر را در متد رویداد کلیک برچسب label1 وارد می‌کنیم.

در پنجره‌ی Properties برای رویداد Click برچسب‌های label2 تا label9 نام متد label1_Click را از فهرست کشویی انتخاب می‌کنیم؛ سپس:

```
private void label1_Click(object sender, EventArgs e)
{
    Label lbl = (Label)sender;      // برچسبی که روی آن کلیک شده
    lbl.Visible = false;            // همان برچسب پنهان می‌شود
}
```

هر بار که روی هر یک از ۹ برچسب کلیک شود، همین متد اجرا شده و فقط همان برچسب پنهان می‌شود؛ زیرا sender کنترلی است که رویداد برای آن رخ داده است.

۱۹۵ صفحه ۱۶۵ فیلم

پس از مشاهده‌ی فیلم، «بازی سرعت کلیک» را نوشته و اجرا کنید. (فیلم شماره‌ی ۱۱۰۶)

بازی سرعت کلیک: یک دکمه در مکان تصادفی فرم ظاهر می‌شود، کاربر باید با کلیک روی آن امتیاز بگیرد و زمان‌سنج بازی را پس از مدت معین پایان می‌دهد. (فیلم آموزشی در دسترس نبود؛ برنامه‌ی زیر طراحی شد.)

```
int score = 0, time = 20;
Random rnd = new Random();
private void btnStart_Click(object sender, EventArgs e)   // دکمه‌ی شروع
{
    score = 0; time = 20;
    lblScore.Text = "0"; lblTime.Text = "20";
    timer1.Interval = 1000; timer1.Enabled = true;
    btnTarget.Visible = true;
}
private void btnTarget_Click(object sender, EventArgs e)  // دکمه‌ی هدف
{
    score++;
    lblScore.Text = score.ToString();
    btnTarget.Left = rnd.Next(0, this.ClientSize.Width - btnTarget.Width);
    btnTarget.Top = rnd.Next(50, this.ClientSize.Height - btnTarget.Height);
}
private void timer1_Tick(object sender, EventArgs e)
{
    time--;
    lblTime.Text = time.ToString();
    if (time == 0)
    {
        timer1.Enabled = false;
        btnTarget.Visible = false;
        MessageBox.Show("پایان بازی! امتیاز شما: " + score);
    }
}
```

این جواب ممکن است دقیق نباشد.

ایران توشه
توشه ای برای موفقیت

فیلم را مشاهده کنید و پروژه‌ی مشابه ایجاد کرده، کدهای آن را بنویسید. (فیلم شماره‌ی ۱۱۰۷)

«بازی اعصاب‌سنج» را طراحی کنید. با سلیقه‌ی خودتان مسیرهای مشکل‌تری برای بازی ایجاد کنید. کادر تصویر لامپ را به پروژه اضافه کنید تا به هنگام خطای کاربر به‌صورت چشمک‌زن ظاهر شود. با خطای کاربر در بازی شکل اشاره‌گر ماوس تغییر کند.

ـ آیا می‌توان در بازی اعصاب‌سنج از رویدادهای MouseMove_Form یا MouseHover_Form برای بررسی خطای کاربر استفاده کرد؟

ـ اگر به‌جای رویداد MouseClick از رویداد Click استفاده کنیم، چه مشکلی پیش می‌آید؟ چه تفاوتی بین رویداد Click و MouseClick وجود دارد؟

طراحی: مسیر بازی را با چند برچسب lblWall1 تا lblWallN با رنگ زمینه‌ی سیاه (دیوارهای مسیر) روی فرم می‌کشیم؛ هدف (خط پایان) برچسب lblFinish و کادر تصویر لامپ picLamp با Visible = false است. اگر ماوس به یکی از دیوارها وارد شود (MouseEnter)، کاربر خطا کرده است.

```
private void Wall_MouseEnter(object sender, EventArgs e)   // برای همه‌ی دیوارها یک متد
{
    this.Cursor = Cursors.No;           // تغییر شکل اشاره‌گر
    picLamp.Visible = true;             // لامپ روشن می‌شود
    timerBlink.Enabled = true;          // Interval = 300 برای چشمک‌زدن
}
private void timerBlink_Tick(object sender, EventArgs e)
{
    picLamp.Visible = !picLamp.Visible;
}
private void lblFinish_MouseEnter(object sender, EventArgs e)
{
    MessageBox.Show("آفرین! به پایان رسیدید");
}
private void lblStart_MouseEnter(object sender, EventArgs e)
{
    this.Cursor = Cursors.Default;      // شروع دوباره
    timerBlink.Enabled = false;
    picLamp.Visible = false;
}
```

استفاده از MouseMove یا MouseHover فرم مناسب نیست: MouseMove روی کل فرم با هر حرکت رخ می‌دهد و تشخیص برخورد با دیوار ممکن نمی‌کند (و تشخیص باید برای هر دیوار جداگانه باشد)؛ MouseHover نیز فقط پس از ساکن ماندن ماوس رخ می‌دهد و برخورد لحظه‌ای را نمی‌بیند. MouseEnter روی هر دیوار بهترین گزینه است.

رویداد Click می‌تواند با صفحه‌کلید (مثلاً فشردن Enter یا Space روی دکمه‌ی دارای فوکوس) هم رخ دهد و آرگومان آن EventArgs است و اطلاعات ماوس (مثل Button و Location) را ندارد؛ ولی MouseClick فقط با کلیک ماوس رخ می‌دهد و آرگومان MouseEventArgs دارد که دکمه و مختصات ماوس را می‌دهد.

این جواب ممکن است دقیق نباشد.

۱۹۷ صفحه ۱۶۷ کنجکاوی

ـ چه تفاوتی بین منوهای بالا از لحاظ شکل و قالب و گروه‌بندی وجود دارد؟

ـ در کدام شکل، منو با راست‌کلیک فعال می‌شود؟

منوی نواری (Menu Bar) در بالای پنجره به‌صورت افقی چیده می‌شود و گزینه‌های اصلی (مثل پرونده، ویرایش، نما) هر یک زیرمنوی عمودی دارند که گزینه‌ها با خط جداکننده گروه‌بندی شده‌اند؛ منوی میان‌بر (Context Menu) در جای ماوس باز می‌شود و فقط شامل عملیات مرتبط با همان محل است و گزینه‌های آن به‌صورت یک فهرست عمودی بدون نوار اصلی هستند.

منوی میان‌بر (Context Menu) با راست‌کلیک فعال می‌شود؛ مثلاً منوی باز شده با راست‌کلیک روی میزکار یا یک پرونده.

۱۹۸ صفحه ۱۶۸ فعالیت گروهی

برای انجام چه عملیاتی در ویندوز از منوها استفاده می‌کنید؟ آیا این عملیات کلید میان‌بر دارند؟ آیا این منوها با کلید Alt فعال می‌شوند؟ یک پنجره باز کنید و کلید Alt را فشار دهید.

با راست‌کلیک در محیط‌های مختلف ویندوز، منوی یکسانی باز می‌شود؟

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

منوها برای باز کردن، ذخیره، چاپ، کپی، برش و چسباندن، جست‌وجو، حذف و تنظیمات استفاده می‌شوند. بیشتر عملیات کلید میان‌بر دارند. با فشردن Alt در بسیاری از پنجره‌ها (مانند Notepad یا File Explorer) منوی نواری فعال شده، حروف دسترسی (زیرخط‌دار) نمایان می‌شوند. منوی راست‌کلیک در محیط‌های مختلف ویندوز یکسان نیست؛ گزینه‌ها بر اساس شیء (میزکار، پوشه، پرونده، متن و ...) تغییر می‌کنند.

عملیات و کلید میان‌بر

ایران توشه
توشه‌ای برای موفقیت

نوع عملیات با منو	کلید میان‌بر
تهیه‌ی نسخه‌ی مشابه پرونده یا پوشه (Copy)	Ctrl+C
چسباندن (Paste)	Ctrl+V
برش (Cut)	Ctrl+X
چاپ (Print)	Ctrl+P
ذخیره (Save)	Ctrl+S
بازکردن پرونده (Open)	Ctrl+O
انتخاب همه (Select All)	Ctrl+A
واگرد (Undo)	Ctrl+Z

۱۹۹ صفحه ۱۶۹ کارگاه ۶

۲ ـ کنترل MenuStrip کجا قرار می‌گیرد؟

۳ ـ گزینه‌های دیگر منو را کجا می‌نویسید؟

۴ ـ با راست‌کلیک روی گزینه‌ها، چه گزینه‌های دیگری مشاهده می‌کنید؟

۵ ـ روی گزینه‌ی «ذخیره» راست‌کلیک کنید و از گزینه‌ی MenuItem، Insert را انتخاب کنید. چه تغییری در منو رخ می‌دهد؟ اگر بخواهید گزینه‌ای را حذف کنید، چگونه اقدام می‌کنید؟

۶ ـ مقدار ویژگی RightToLeft را Yes کنید. چه تغییری در منو رخ می‌دهد؟

کنترل MenuStrip در بالای فرم قرار می‌گیرد (به‌صورت نواری) و علاوه بر آن به‌عنوان یک component در سینی اجزای پایین پنجره‌ی طراحی نیز دیده می‌شود.

گزینه‌های بعدی منوی اصلی را در کادر Type Here کنار گزینه‌ی «پرونده» و گزینه‌های زیرمنو را در کادر Type Here زیر هر گزینه می‌نویسیم.

با راست‌کلیک روی گزینه‌ها گزینه‌هایی مانند Insert (با زیرگزینه‌های MenuItem، ComboBox، TextBox و Separator)، Delete، Set Image، Enabled، Checked، ShowShortcutKeys، Properties و ... دیده می‌شود.

با Insert ← MenuItem یک گزینه‌ی خالی جدید قبل از گزینه‌ی «ذخیره» وارد می‌شود تا متن آن را بنویسیم؛ برای حذف یک گزینه روی آن راست‌کلیک کرده و Delete را انتخاب می‌کنیم (یا آن را انتخاب و کلید Delete را می‌زنیم).

با RightToLeft = Yes چیدمان منو راست‌به‌چپ می‌شود: گزینه‌ی «پرونده» در سمت راست قرار می‌گیرد و متن گزینه‌ها راست‌چین می‌شود (مناسب منوهای فارسی).

۲۰۰ صفحه ۱۷۰ کارگاه ۷

۲ ـ ویژگی RightToLeft را برابر Yes قرار دهید. شکل منو چه تغییری می‌کند؟

۴ ـ ویژگی Enabled گزینه‌ی «چسباندن» را false کنید. شکل این گزینه چه تغییری می‌کند؟

۵ ـ ویژگی Visible گزینه‌ی «ذخیره» را false کنید. شکل این گزینه چه تغییری می‌کند؟ آیا پس از اجرا گزینه تغییر می‌کند؟

۶ ـ ویژگی Checked گزینه‌ی «قرمز» از «رنگ زمینه» را true کنید. شکل این گزینه چه تغییری می‌کند؟

۷ ـ از چه قالب‌های تصویری می‌توان برای تصویر گزینه‌ی منو استفاده کرد؟

۸ ـ چه ویژگی‌هایی باید تغییر کنند؟ (رنگ زمینه و متن گزینه‌های منو)

۱۰ ـ متن گزینه‌ی «پرونده» را به «&پرونده» تغییر دهید. شکل این گزینه چه تغییری می‌کند؟ برنامه را اجرا کنید. هم‌زمان کلید ترکیبی Alt و پ را بگیرید. چه مشاهده می‌کنید؟ کلید Alt را بگیرید. چه مشاهده می‌کنید؟

- RightToLeft = Yes: گزینه‌های منوی اصلی از راست شروع می‌شوند و زیرمنوها راست‌چین می‌شوند.
- Enabled = false: گزینه‌ی «چسباندن» کم‌رنگ (خاکستری) می‌شود و قابل انتخاب نیست.
- Visible = false: گزینه در زمان طراحی کم‌رنگ دیده می‌شود ولی پس از اجرا کاملاً حذف و ناپدید است و دیده نمی‌شود.
- Checked = true: علامت تیک (√) کنار گزینه‌ی «قرمز» ظاهر می‌شود.
- قالب‌های تصویری مناسب: png، jpg، bmp، gif، ico (قالب‌های پشتیبانی‌شده برای ویژگی Image).
- برای رنگ‌ها: BackColor و ForeColor گزینه‌های منو (و در صورت نیاز Font).
- ویژگی ShortcutKeys گزینه‌ی «خروج از برنامه» را Ctrl+E قرار می‌دهیم.
- با نوشتن «&پرونده» حرف «پ» زیرخط‌دار می‌شود (در حالت فعال شدن منو با Alt)؛ با گرفتن Alt + پ منوی «پرونده» باز می‌شود و با فقط گرفتن Alt حروف دسترسی زیرخط‌دار نمایان می‌شوند.

برای نام‌گذاری گزینه‌ها از پیشوند mnu استفاده می‌کنیم؛ مثلاً mnuNew برای «جدید» و mnuExit برای «خروج».

۲۰۱ صفحه ۱۷۰ فعالیت منزل

منوی نرم‌افزار Notepad ویندوز را در یک پروژه طراحی کنید.

کنترل MenuStrip را روی فرم گذاشته، ساختار زیر را با Insert → MenuItem و Separator می‌سازیم (نام‌گذاری با پیشوند mnu):

منوی Notepad

منوی اصلی	گزینه‌ها (کلید میان‌بر)
پرونده (File)	جدید (Ctrl+N)، باز کردن (Ctrl+O)، ذخیره (Ctrl+S)، ذخیره به‌عنوان، — جداکننده —، چاپ (Ctrl+P)، خروج
ویرایش (Edit)	واگرد (Ctrl+Z)، — جداکننده —، برش (Ctrl+X)، کپی (Ctrl+C)، چسباندن (Ctrl+V)، حذف (Del)، جست‌وجو (Ctrl+F)، انتخاب همه (Ctrl+A)، زمان/تاریخ (F5)
قالب (Format)	شکستن خودکار خط (Word Wrap)، قلم (Font...)
نما (View)	نوار وضعیت (Status Bar)
راهنما (Help)	مشاهده‌ی راهنما، درباره‌ی Notepad

۲۰۲ صفحه ۱۷۱ فعالیت کارگاهی

جدول ۴ شامل ویژگی‌های متداول گزینه‌ی منو است. با توجه به کارگاه ۷ این جدول را تکمیل کنید.

جدول ۴ـ ویژگی‌های کنترل منو

ایران توشه
توشه‌ای برای موفقیت

ویژگی	شرح
Name	نام منو یا گزینه‌های منو
Text	متن (عنوان) گزینه‌های منو
Checked	نمایش یا عدم نمایش علامت تیک کنار گزینه
Enabled	فعال یا غیرفعال بودن گزینه (در حالت غیرفعال کم‌رنگ می‌شود)
Visible	نمایش یا پنهان بودن گزینه
Image	تصویر کنار گزینه
RightToLeft	جهت نمایش (راست‌به‌چپ یا چپ‌به‌راست) متن و چیدمان گزینه‌ها
ShortcutKeys	کلید یا ترکیب کلید میان‌بر گزینه (مثل Ctrl+E)
ToolTipText	متن راهنما که با نگه داشتن ماوس روی گزینه نمایش داده می‌شود

۲۰۳ صفحه ۱۷۱ کنجکاوی

آیا می‌توان جهت نمایش متن گزینه‌های منو را تغییر داد؟

بله؛ با ویژگی RightToLeft منو (راست‌به‌چپ یا چپ‌به‌راست) می‌توان جهت چیدمان و نمایش گزینه‌ها را تغییر داد، و با ویژگی TextDirection گزینه‌ها (افقی، عمودی ...) جهت متن گزینه قابل تنظیم است.

۲۰۴ صفحه ۱۷۲ پژوهش

بررسی کنید که چه رویدادهای دیگری برای منو استفاده می‌شود و کاربرد آنها چیست؟

ایران توشه
توشه ای برای موفقیت

- DropDownOpening: هنگام باز شدن زیرمنو رخ می‌دهد؛ مناسب برای فعال/غیرفعال کردن گزینه‌ها قبل از نمایش (مثلاً فعال بودن «چسباندن»).
- DropDownClosed: هنگام بسته شدن زیرمنو رخ می‌دهد.
- MouseEnter و MouseHover: ورود ماوس به گزینه (مثلاً نمایش راهنما در نوار وضعیت).
- CheckedChanged: هنگام تغییر وضعیت علامت گزینه (تیک‌دار شدن).
- CheckStateChanged و Paint و TextChanged: به‌ترتیب هنگام تغییر حالت علامت، رسم گزینه و تغییر متن.

۲۰۵ صفحه ۱۷۲ کارگاه ۸

عملکرد گزینه‌ی «پاک» چیست؟

پس از اجرای برنامه، با کلیک روی گزینه‌ی «سفید» چه رخ می‌دهد؟ برای گزینه‌ی «زرد» از زیرمنوی «رنگ زمینه»، همانند مرحله‌ی قبل، کدنویسی کنید.

۶ ـ برای گزینه‌های «قرمز» و «آبی» از زیرمنوی «رنگ متن» نیز کد بنویسید.

۷ ـ متد رویداد کلیک گزینه‌ی «فونت» را بنویسید. کد تغییر قلم کادرمتن txtPaper را با استفاده از کادر محاوره‌ای fontDialog1 بنویسید.

۸ ـ برای گزینه‌ی «خروج از برنامه» کد مناسب بنویسید.

گزینه‌ی «پاک» با ()txtPaper.Clear تمام متن کادر متن را پاک می‌کند. با کلیک روی «سفید» رنگ زمینه‌ی کادر متن سفید شده، کنار «سفید» علامت تیک ظاهر می‌شود و تیک گزینه‌ی «زرد» برداشته می‌شود.

ایران توشه
توشه ای برای موفقیت

```
private void mnuClear_Click(object sender, EventArgs e)
{
    txtPaper.Clear();
}
private void mnuWhite_Click(object sender, EventArgs e)
{
    txtPaper.BackColor = Color.White;
    mnuWhite.Checked = true;
    mnuYellow.Checked = false;
}
private void mnuYellow_Click(object sender, EventArgs e)
{
    txtPaper.BackColor = Color.Yellow;
    mnuYellow.Checked = true;
    mnuWhite.Checked = false;
}
private void mnuBlack_Click(object sender, EventArgs e)
{
    txtPaper.ForeColor = Color.Black;
    mnuBlack.Checked = true; mnuRed.Checked = false; mnuBlue.Checked = false;
}
private void mnuRed_Click(object sender, EventArgs e)
{
    txtPaper.ForeColor = Color.Red;
    mnuRed.Checked = true; mnuBlack.Checked = false; mnuBlue.Checked = false;
}
private void mnuBlue_Click(object sender, EventArgs e)
{
    txtPaper.ForeColor = Color.Blue;
    mnuBlue.Checked = true; mnuBlack.Checked = false; mnuRed.Checked = false;
}
private void mnuFont_Click(object sender, EventArgs e)
{
    if (fontDialog1.ShowDialog() == DialogResult.OK)
        txtPaper.Font = fontDialog1.Font;
}
private void mnuExit_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

ایران توشه
توشه ای برای موفقیت

۲۰۶ صفحه ۱۷۳ فعالیت کارگاهی

بعد از افزودن کنترل MenuStrip به فرم، روی آن راست‌کلیک کنید و گزینه‌ی Insert Standard Items را انتخاب کنید. در فرم برنامه چه مشاهده می‌کنید؟ منوی ایجادشده را به‌دلخواه ویرایش و سفارشی کنید. آیا پس از اجرای برنامه و کلیک روی گزینه‌ها، عملیاتی انجام می‌شود؟

یک منوی استاندارد شامل منوهای File (New، Open، Save، Save As، Print، Print Preview، Exit)، Edit (Undo، Redo، Cut، Copy، Paste، Select All)، Tools (Customize، Options) و Help (Contents، Index، Search، About) به‌صورت خودکار در بالای فرم ساخته می‌شود. این گزینه‌ها فقط ظاهر منو را می‌سازند؛ پس از اجرا با کلیک روی آنها هیچ عملیاتی انجام نمی‌شود، زیرا برای هر گزینه باید متد رویداد Click نوشته شود.

۲۰۷ صفحه ۱۷۴ کارگاه ۹

برای تغییر Theme ویندوز چگونه عمل می‌کنید؟ برای مشاهده‌ی ویژگی‌های سیستم (System Properties) چگونه عمل می‌کنید؟

۳ ـ ... برنامه را اجرا کنید و روی کنترل کادر متن راست‌کلیک کنید. چه چیزی مشاهده می‌کنید؟ برای اینکه با راست‌کلیک روی فرم همین منو نمایش داده شود، چگونه عمل می‌کنید؟

۴ ـ برای رویداد کلیک گزینه‌ها، کد بنویسید.

چه تفاوتی بین کنترل MenuStrip و ContextMenuStrip وجود دارد؟

برای تغییر Theme: راست‌کلیک روی میزکار ← Personalize (شخصی‌سازی) ← Themes؛ برای ویژگی‌های سیستم: راست‌کلیک روی This PC (رایانه) ← Properties. هر دو با راست‌کلیک و استفاده از منوی میان‌بر انجام می‌شوند.

پس از قرار دادن ContextMenuStrip و تخصیص آن به ویژگی ContextMenuStrip کادر متن، با راست‌کلیک روی کادر متن همان منوی نوشته (رنگ زمینه، رنگ متن، قلم، پاک، خروج) باز می‌شود. برای نمایش همین منو با راست‌کلیک روی فرم نیز کافی است ویژگی ContextMenuStrip فرم را روی contextMenuStrip1 قرار دهیم.

برای رویداد کلیک گزینه‌ها، همان کدهای کارگاه ۸ نوشته می‌شود (می‌توان رویداد Click گزینه‌های منوی زمینه را همان متدهای قبلی (مثلاً mnuClear_Click) تعیین کرد):

```
private void ctxClear_Click(object sender, EventArgs e)
{
    txtPaper.Clear();
}
private void ctxRed_Click(object sender, EventArgs e)
{
    txtPaper.ForeColor = Color.Red;
}
private void ctxFont_Click(object sender, EventArgs e)
{
    if (fontDialog1.ShowDialog() == DialogResult.OK)
        txtPaper.Font = fontDialog1.Font;
}
```

تفاوت: MenuStrip منوی نواری ثابت در بالای فرم است و همیشه دیده می‌شود، ولی ContextMenuStrip منوی زمینه (میان‌بر) است که به یک کنترل یا فرم نسبت داده می‌شود و فقط با راست‌کلیک روی آن، در محل اشاره‌گر ظاهر می‌شود.

۲۰۸ صفحه ۱۷۵ کارگاه ۱۰

۳ ـ متد رویداد کلیک گزینه‌ی «شهدای دفاع مقدس» را بنویسید. قطعه کدی به برنامه اضافه کنید تا با کلیک روی هر تصویر اطلاعاتی مربوط به آن در ToolTip نمایش داده شود.

عملکرد علامت @ قبل از مسیر پرونده چیست؟

۴ ـ متد رویداد کلیک گزینه‌ی «سرود جمهوری اسلامی» را بنویسید.

ایران توشه
توشه ای برای موفقیت

۵ ـ برنامه‌ی نمایش تصاویر شهدای مدافع حرم را همانند شکل تغییر دهید. ۶ ـ با حرکت ماوس روی تصویر شهید، توضیحی از ویژگی‌های شهید در پایین تصویر نمایش داده شود. ۷ ـ قابلیتی به برنامه اضافه کنید تا با کلیک روی هر تصویر، تصویر در اندازه‌ی بزرگ‌تر نمایش داده شود.

```
private void mnuShohadaDefa_Click(object sender, EventArgs e)
{
    pic1.ImageLocation = @"images\hemmat.jpg";
    pic2.ImageLocation = @"images\aviny.jpg";
    pic3.ImageLocation = @"images\mahdavi.jpg";
    pic4.ImageLocation = @"images\hadi.jpg";
    pic5.ImageLocation = @"images\doran.jpg";
    pic6.ImageLocation = @"images\babaee.jpg";
    pic7.ImageLocation = @"images\chamran.jpg";
    pic8.ImageLocation = @"images\bronsi.jpg";
    pnlPictures.Visible = true;
    lblTitle.Text = "شهدای دفاع مقدس";
    lblDescription.Text = "آمار شهدای دفاع مقدس ۱۸۸ هزار و ۱۵ نفر است";
}
// ToolTip برای هر تصویر
private void pic1_Click(object sender, EventArgs e)
{
    ToolTip tip = new ToolTip();
    tip.Show("شهید همت؛ فرمانده لشکر ۲۷ محمد رسول‌الله", pic1, 3000);
}

private void mnuSorood1_Click(object sender, EventArgs e)
{
    System.Media.SoundPlayer player;
    player = new System.Media.SoundPlayer(@"sounds\Sorod.wav");
    player.Play();
    mnuSorood1.Checked = true;
    mnuSorood2.Checked = false;
    // ... mnuSorood3 تا mnuSorood6 نیز false
}
```

علامت @ قبل از رشته، آن را «رشته‌ی تحت‌اللفظی» (verbatim) می‌کند: نویسه‌ی \ در مسیر پرونده (images\hemmat.jpg) به‌عنوان نویسه‌ی فرار (escape) تفسیر نمی‌شود و نیازی به نوشتن \\ نیست. کلاس SoundPlayer فقط فایل صوتی با قالب wav را پخش می‌کند و متد Play آن صدا را پخش می‌کند.

مراحل ۵ تا ۷

```
// ۶: نمایش توضیح با حرکت ماوس روی تصویر
private void pic1_MouseEnter(object sender, EventArgs e)
{
    lblDescription.Text = "شهید حسین همدانی؛ از فرماندهان مدافع حرم";
}
// ۷: بزرگ‌نمایی با کلیک روی تصویر (یک متد برای همه‌ی تصویرها)
private void pic_Click(object sender, EventArgs e)
{
    PictureBox p = (PictureBox)sender;
    picBig.ImageLocation = p.ImageLocation;
    picBig.Visible = true;
}
```

این جواب ممکن است دقیق نباشد.

۲۰۹ صفحه ۱۸۰ **فعالیت کارگاهی**

در نرم‌افزار Word برای تایپ واژگان و قالب‌بندی آنها چگونه از صفحه‌کلید استفاده می‌کنید؟

اگر رایانه دستگاه ماوس نداشته باشد چگونه در سیستم‌عامل ویندوز یک پنجره را می‌بندید؟

چرا در بعضی از صفحه‌کلیدها، رنگ برخی از کلیدها متفاوت است؟

نرم‌افزار Notepad را اجرا کنید. با فشردن کلیدهای زیر چه چیزی نمایش داده می‌شود؟ (کلید جهت‌دار بالا، A، Ctrl، BackSpace)

با فشردن کلیدهای زیر چه نوع نویسه‌ای ایجاد می‌شود؟ (0، A، Enter، Space، Home، Alt، #)

- حروف و اعداد با کلیدهای نویسه‌ای تایپ می‌شوند و قالب‌بندی با کلیدهای میان‌بر انجام می‌شود (مثلاً Ctrl+B برای پررنگ، Ctrl+I برای مورب، Ctrl+U برای زیرخط) و کلید Shift برای حروف بزرگ.
- با ترکیب کلیدهای Alt+F4 پنجره‌ی فعال بسته می‌شود.
- کلیدهای رنگی (مثل کلیدهای کنترلی، تابعی، یا کلیدهای عددی) برای تفکیک گروه‌های کلیدها و پیدا کردن سریع‌تر آنها هستند.

ایران توشه
توشه ای برای موفقیت

نتیجه‌ی فشردن کلید در Notepad

کلید	نتیجه
کلید جهت‌دار بالا (UpArrow)	مکان‌نما یک سطر به بالا می‌رود (نویسه‌ای تولید نمی‌شود)
A	نویسه‌ی a (یا A با Shift یا Caps Lock) نوشته می‌شود
Ctrl	به‌تنهایی هیچ چیز نمایش داده نمی‌شود (کلید ترکیبی است)
BackSpace	نویسه‌ی قبل از مکان‌نما پاک می‌شود

نوع نویسه

کلید	نوع نویسه
Alt	غیر چاپ‌شدنی
Home	غیر چاپ‌شدنی
Space	چاپ‌شدنی (فاصله)
Enter	غیر چاپ‌شدنی (رفتن به سطر بعد)
A	چاپ‌شدنی
0	چاپ‌شدنی
#	چاپ‌شدنی

۲۱۰ صفحه ۱۸۱ کارگاه ۱

شرح رویدادهای صفحه‌کلید در جدول ۵ آمده است. جدول را تکمیل کنید.

۲ ـ ... برنامه را اجرا کنید. کلیدی را به دلخواه فشار دهید. چه اتفاقی می‌افتد؟

۳ ـ در متد رویداد KeyUp فرم کدی بنویسید که رنگ زمینه‌ی فرم را آبی کند. برنامه را اجرا کنید. کلیدی را به دلخواه فشار دهید. چه اتفاقی می‌افتد؟

جدول ۵ـ رویدادهای صفحه‌کلید

نام رویداد	شرح رویداد
KeyDown	زمانی رخ می‌دهد که کلیدی فشرده شود (اولین بار).
KeyPress	زمانی رخ می‌دهد که کنترل فوکوس داشته باشد و کاربر کلیدی را فشرده و رها کند (کلید نویسه‌ای).
KeyUp	زمانی رخ می‌دهد که کلیدی رها شود.

ایران توشه
توشه‌ای برای موفقیت

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    this.BackColor = Color.Red;
}
private void Form1_KeyUp(object sender, KeyEventArgs e)
{
    this.BackColor = Color.Blue;
}
```

با فشردن هر کلید رنگ فرم قرمز می‌شود و با رها کردن کلید رنگ آن آبی می‌شود.

۲۱۱ صفحه ۱۸۲ کارگاه ۲

۴ ـ برنامه را اجرا کرده، کلیدهای زیر را وارد کنید و نتیجه را در جدول زیر بنویسید (Alt، Home، Space، Enter، A، 0، #). نتیجه‌ی حاصل از ورود نویسه‌های چاپ‌شدنی و غیرچاپ‌شدنی چیست؟

با کمک هم‌گروهی خود، متن انگلیسی زیر را برای آشنایی بهتر با رویدادهای صفحه‌کلید، ترجمه کنید. The KeyPress event is not raised by non-character keys other than space and backspace; however, the non-character keys do raise the KeyDown and KeyUp events. Character keys include letters, numbers, punctuation, the Space bar, and the following non-printing characters: Tab, Enter, Return, Delete (or Backspace), Clear and Esc (Escape).

رویدادهای رخ‌داده

کلید	نتیجه (برچسب)
Alt	KeyDown-KeyUp- (KeyPress رخ نمی‌دهد)
Home	KeyDown-KeyUp-
Space	KeyDown-KeyPress-KeyUp-
Enter	KeyDown-KeyPress-KeyUp-
A	KeyDown-KeyPress-KeyUp-
0	KeyDown-KeyPress-KeyUp-
#	KeyDown-KeyPress-KeyUp- (با Shift؛ ابتدا KeyDown مربوط به Shift)

نتیجه: برای نویسه‌های چاپ‌شدنی (و Space، Enter، BackSpace، Esc، Tab) هر سه رویداد (KeyDown، KeyPress، KeyUp) به این ترتیب رخ می‌دهد، ولی برای نویسه‌های غیرچاپ‌شدنی (مانند Alt و Home) فقط KeyDown و KeyUp رخ می‌دهد.

ترجمه‌ی متن انگلیسی

رویداد KeyPress به‌وسیله‌ی کلیدهای غیرنویسه‌ای، جز کلیدهای فاصله (Space) و پاک‌کردن به عقب (Backspace)، ایجاد نمی‌شود؛ با این حال، کلیدهای غیرنویسه‌ای رویدادهای KeyDown و KeyUp را ایجاد می‌کنند. کلیدهای نویسه‌ای شامل حروف، اعداد، علائم نگارشی، کلید فاصله و نویسه‌های غیرچاپی زیر هستند: Tab، Enter، Return، Delete (یا Backspace)، Clear و Esc (Escape).

۲۱۲ صفحه ۱۸۴ کارگاه ۳

عملکرد متد رویداد Form1_KeyDown چیست؟

عملکرد timer1 در این برنامه چیست؟ برنامه را اجرا کنید. یک کلید دلخواه را نگه دارید. چه اتفاقی می‌افتد؟ کلید را رها کنید. چه اتفاقی می‌افتد؟

۸ ـ برنامه را طوری تغییر دهید که توپ پس از رها کردن کلید در جایگاه اولیه‌اش متوقف شود.

آیا نوع آرگومان e در رویداد KeyDown و KeyPress یکسان است و ویژگی‌های یکسانی دارد؟

متد Form1_KeyDown با فشردن هر کلید، متغیر isUp را true می‌کند (جهت حرکت توپ به بالا) و زمان‌سنج را فعال می‌کند. timer1 هر ۱۰ میلی‌ثانیه Tick را اجرا می‌کند و بسته به isUp توپ را ۵ پیکسل بالا یا پایین می‌برد. با نگه داشتن کلید توپ به بالا می‌رود (پرتاب) و با رها کردن کلید (KeyUp: isUp = false) توپ شروع به پایین آمدن (سقوط) می‌کند.

```
bool isUp;
int startTop;                                   // مکان اولیه‌ی توپ (در Form_Load:
startTop = picBall.Top)
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    isUp = true;
    timer1.Enabled = true;
}
private void Form1_KeyUp(object sender, KeyEventArgs e)
{
    isUp = false;
}
private void timer1_Tick(object sender, EventArgs e)
{
    if (isUp)
        picBall.Top -= 5;
    else
    {
        picBall.Top += 5;
        if (picBall.Top >= startTop)        // رسیدن به جایگاه اولیه
        {
            picBall.Top = startTop;
            timer1.Enabled = false;
        }
    }
}
```

خیر؛ آرگومان e در KeyDown و KeyUp از نوع KeyEventArgs (دارای KeyCode، KeyValue، KeyData، Alt، Shift، Control، SuppressKeyPress) و در KeyPress از نوع KeyPressEventArgs (دارای KeyChar و Handled) است.

۲۱۳ صفحه ۱۸۵ کارگاه ۴

کد اسکی (ASCII code) چیست؟

۴ ـ برنامه را اجرا کرده، کلیدهای زیر را فشار دهید و نتایج را در جدول بنویسید: B، b، + و –، + و – بخش عددی صفحه‌کلید، Shift.

ایران توشه
توشه ای برای موفقیت

آیا در این برنامه، نتیجه برای حروف کوچک و بزرگ یکسان است؟ آیا رویداد KeyDown و KeyUp برای تشخیص حروف الفبای کوچک و بزرگ مناسب است؟ کدام ویژگی آرگومان e کد اسکی نویسه را مشخص می‌کند؟

کد اسکی (ASCII: American Standard Code for Information Interchange) یک جدول استاندارد است که به هر نویسه‌ی انگلیسی، عدد و علامت یک کد عددی (۰ تا ۱۲۷) اختصاص می‌دهد؛ مثلاً A=۶۵، B=۶۶، a=۹۷، b=۹۸ و 0=۴۸.

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    lblCharacter.Text = e.KeyCode.ToString();
    lblCode.Text = e.KeyValue.ToString();
}
```

جدول مرحله‌ی ۴ (KeyDown و KeyCode/KeyValue)

نام کلید	نویسه (KeyCode)	کد اسکی (KeyValue)
B	B	۶۶
b	B	۶۶
+ و - (ردیف بالای حروف)	Oemplus و OemMinus	۱۸۷ و ۱۸۹
+ و - بخش عددی صفحه‌کلید	Add و Subtract	۱۰۷ و ۱۰۹
Shift	ShiftKey	۱۶

بله؛ برای حروف کوچک و بزرگ نتیجه یکسان است (B و b هر دو KeyCode = B و KeyValue = 66)؛ زیرا KeyUp/KeyDown نویسه را نمی‌شناسند و فقط کلید فیزیکی را می‌شناسند. بنابراین این دو رویداد برای تشخیص حروف کوچک و بزرگ مناسب نیستند (مگر با بررسی e.Shift). ویژگی KeyValue کد کلید را برمی‌گرداند.

ایران توشه
توشه ای برای موفقیت

۲۱۴ صفحه ۱۸۶ کارگاه ۴

۵ ـ ویژگی KeyData را برای مرحله‌ی ۳ آزمایش کنید ... جدول مرحله‌ی ۴ و ۵ را با هم مقایسه کنید.

۶ ـ کد اسکی نویسه‌ها را با استفاده از رویداد KeyPress نمایش دهید.

۷ ـ جدول مرحله‌ی ۴ را با توجه به برنامه‌ی جدید، تکمیل کنید. آیا رویداد KeyPress برای تشخیص حروف الفبای کوچک و بزرگ مناسب است؟ آیا کد کلیدهای Shift و جهت‌دار نمایش داده می‌شود؟

مرحله‌ی ۵: نمایش KeyData

نام کلید	KeyData
B (بدون Shift)	B
B / b با Shift	B, Shift
+ (ردیف بالا، با Shift)	Oemplus, Shift
- (ردیف بالا)	OemMinus
+ و - بخش عددی	Add و Subtract
Shift	ShiftKey, Shift

تفاوت: KeyCode فقط نام کلید را می‌دهد ولی KeyData کلید به‌همراه کلیدهای ترکیبی (Shift، Control، Alt) را نشان می‌دهد؛ مثلاً Ctrl+A برای KeyData مقدار A, Control است.

```
private void Form1_KeyPress(object sender, KeyPressEventArgs e)
{
    lblCharacter.Text = e.KeyChar.ToString();
    int code = (int)e.KeyChar;
    lblCode.Text = code.ToString();
}
```

جدول مرحله‌ی ۷ (KeyPress)

نام کلید	نویسه	کد اسکی
B	B	۶۶
b	b	۹۸
+ (هر دو)	+	۴۳
- (هر دو)	-	۴۵
Shift	رویدادی رخ نمی‌دهد	—

بله، KeyPress برای تشخیص حروف کوچک و بزرگ مناسب است (کد اسکی ۶۶ و ۹۸ متفاوت است)؛ ولی کد کلیدهای Shift و جهت‌دار نمایش داده نمی‌شود، زیرا این کلیدها غیرنویسه‌ای هستند و KeyPress برای آنها رخ نمی‌دهد.

۲۱۵ صفحه ۱۸۷ کنجکاوی

برای شناسایی کلید فشرده‌شده در رویداد KeyPress از چه ویژگی استفاده کردیم؟ نوع داده‌ی آن چیست؟

ایران توشه
توشه ای برای موفقیت

مشابه جدول ۶ ویژگی‌های آرگومان e در رویداد KeyPress را بنویسید.

از ویژگی e.KeyChar از نوع char استفاده کردیم (و برای کد اسکی آن را به int تبدیل کردیم).

ویژگی‌های آرگومان e در رویداد KeyPress (کلاس KeyPressEventArgs)

نام ویژگی	نوع داده	شرح
KeyChar	char	نویسه‌ی مربوط به کلید فشرده‌شده را برمی‌گرداند یا تنظیم می‌کند.
Handled	bool	اگر true شود، رویداد پردازش‌شده تلقی می‌شود و نویسه وارد کنترل نمی‌شود (جلوگیری از ورود نویسه).

۲۱۶ صفحه ۱۸۷ **فعالیت منزل**

برنامه‌ای بنویسید که با فشردن کلیدهای فارسی، کد حروف فارسی نمایش داده شود.

با زبان فارسی صفحه‌کلید، رویداد KeyPress نویسه‌ی یونیکد حرف را در e.KeyChar می‌دهد و تبدیل آن به int کد یونیکد (مثلاً «ا» ← ۱۵۷۵ و «ب» ← ۱۵۷۶) است:

```
private void Form1_KeyPress(object sender, KeyPressEventArgs e)
{
    lblCharacter.Text = e.KeyChar.ToString();
    lblCode.Text = ((int)e.KeyChar).ToString();   // کد یونیکد حرف فارسی
}
```

۲۱۷ صفحه ۱۸۹ **کارگاه ۵**

ـ متغیر ratio چه مقداری می‌گیرد و علت استفاده از آن چیست؟

ـ اعداد ۱۸۷ و ۱۸۹ کد اسکی چه کلیدهایی هستند؟

ـ عملکرد متد رویداد KeyDown فرم چیست؟

۴ ـ برنامه را طوری تغییر دهید که با کلیدهای + و - در قسمت عددی صفحه‌کلید هم، تصویر بزرگ و کوچک شود.

۵ ـ برنامه را طوری تغییر دهید که با تغییر اندازه‌ی تصویر، کادرتصویر در وسط فرم قرار گیرد.

ایران توشه
توشه‌ای برای موفقیت

۶ ـ برنامه را طوری تغییر دهید که با تغییر اندازه‌ی تصویر، ارتفاع تصویر حداکثر ۳۰۰ و حداقل ۵۰ پیکسل شود.

متغیر ratio نسبت عرض به ارتفاع کادر تصویر (Width / Height) را می‌گیرد؛ با ضرب آن در ارتفاع جدید، عرض متناسب محاسبه می‌شود تا تصویر هنگام بزرگ و کوچک شدن تغییر شکل ندهد. اعداد ۱۸۷ و ۱۸۹ کد کلیدهای + (Oemplus) و – (OemMinus) ردیف بالای حروف هستند. متد رویداد با فشردن Ctrl و + ارتفاع را ۵ پیکسل زیاد، با Ctrl و – ۵ پیکسل کم می‌کند و عرض را متناسب با آن تنظیم می‌کند.

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    double ratio = (double)picSample.Width / picSample.Height;
    if (e.Control)
    {
        // + (109 ,107) و بخش عددی (189 ,187) و - ردیف بالا
        if (e.KeyValue == 187 || e.KeyValue == 107)
        {
            if (picSample.Height < 300) picSample.Height += 5;
        }
        else if (e.KeyValue == 189 || e.KeyValue == 109)
        {
            if (picSample.Height > 50) picSample.Height -= 5;
        }
        picSample.Width = (int)(ratio * picSample.Height);
        // قرار دادن کادرتصویر در وسط فرم
        picSample.Left = (this.ClientSize.Width - picSample.Width) / 2;
        picSample.Top = (this.ClientSize.Height - picSample.Height) / 2;
    }
}
```

۲۱۸ صفحه ۱۹۰ فعالیت منزل

برنامه‌ی کارگاه ۵ را با استفاده از ویژگی KeyData بنویسید.

ایران توشه
توشه ای برای موفقیت

KeyData مقدار شمارشی Keys است که کلید و کلیدهای ترکیبی را با هم دارد؛ ترکیب Ctrl و + برابر Keys.Control | Keys.Oemplus است:

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    double ratio = (double)picSample.Width / picSample.Height;
    if (e.KeyData == (Keys.Control | Keys.Oemplus))
        picSample.Height += 5;
    else if (e.KeyData == (Keys.Control | Keys.OemMinus))
        picSample.Height -= 5;
    picSample.Width = (int)(ratio * picSample.Height);
}
```

۲۱۹ صفحه ۱۹۰ کارگاه ۶

۴ ـ برنامه را دوباره اجرا کنید. اگر کلیدی فشار دهید، آیا کد اسکی و نویسه نمایش داده می‌شوند؟ آیا رویداد KeyDown فرم رخ می‌دهد؟

ـ اگر کلید حرفی را تایپ کنید، آیا متنی در کادرمتن نوشته می‌شود؟ ـ کلید Tab را فشار دهید. اگر کلید حرفی تایپ کنید متنی در کادرمتن نوشته می‌شود؟ ـ اگر کلیدی فشار دهید، آیا کد اسکی و نویسه نمایش داده می‌شوند؟ آیا رویداد KeyDown فرم رخ می‌دهد؟

چند کنترل در جعبه‌ابزار بیابید که به رویدادهای صفحه‌کلید واکنش نشان می‌دهند.

ـ آیا در کارگاه ۴ باید مقدار ویژگی KeyPreview، true شود؟ چرا؟

ـ برای تغییر فوکوس در برنامه از چه کلیدی استفاده می‌کنید؟ در هنگام برنامه‌نویسی از چه کدی برای تغییر فوکوس استفاده می‌کنید؟

اگر فوکوس روی کادر متن باشد (KeyPreview = false)، رویدادهای صفحه‌کلید فرم رخ نمی‌دهند: کد اسکی و نویسه نمایش داده نمی‌شوند؛ و حرف تایپ‌شده در کادر متن نوشته می‌شود. پس از فشردن Tab فوکوس به دکمه می‌رود؛ تایپ حرف، متنی در کادر متن نمی‌نویسد. با KeyPreview = true فرم قبل از کنترل به رویدادهای صفحه‌کلید واکنش نشان می‌دهد و در هر دو حالت (فوکوس روی کادر متن یا دکمه) کد اسکی و نویسه نمایش داده می‌شوند و رویداد KeyDown فرم رخ می‌دهد.

کنترل‌هایی مانند TextBox، Button، ListBox، ComboBox، CheckBox، RadioButton و DataGridView رویدادهای صفحه‌کلید دارند.

در کارگاه ۴ فقط فرم دارای کنترلی نبود که فوکوس را بگیرد؛ پس نیازی نیست، ولی اگر در فرم کنترل‌هایی مانند کادر متن یا دکمه وجود داشته باشد باید KeyPreview = true شود. با کلید Tab فوکوس تغییر می‌کند و در برنامه‌نویسی با متد Focus() (مثلاً txtLastName.Focus();).

ایران توشه
توشه ای برای موفقیت

۲۲۰ صفحه ۱۹۱ فعالیت منزل

پس از مشاهده‌ی فیلم، «برنامه‌ی یافتن نویسه در صفحه‌کلید» را بنویسید. (فیلم شماره‌ی ۱۱۰۸)

برنامه نویسه‌ی تایپ‌شده را دریافت کرده و نمایش می‌دهد که روی کدام کلید است (و با رنگ کردن برچسب کلید مربوط، آن را «پیدا می‌کند»). فیلم در دسترس نبود؛ طراحی ساده: برای هر کلید یک برچسب هم‌نام (مثلاً lblA) وجود دارد.

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    foreach (Control c in this.Controls)
        if (c is Label) c.BackColor = Color.White;
    Control[] found = this.Controls.Find("lbl" + e.KeyCode.ToString(), true);
    if (found.Length > 0)
        found[0].BackColor = Color.Yellow;     // کلید فشرده‌شده
}
```

این جواب ممکن است دقیق نباشد.

۲۲۱ صفحه ۱۹۲ کارگاه ۷

۴ ـ کد را تغییر دهید تا سفینه علاوه بر کلیدهای جهت‌دار با فشار کلیدهای A ـ W ـ S ـ D هم حرکت کند.

برنامه را اجرا کنید.

آیا ویژگی KeyPreview فرم، باید true شود؟ چرا؟

ـ در کارگاه ۷ برنامه را تغییر دهید تا با فشردن کلیدهای جهت‌دار، سفینه به‌صورت خودکار در همان جهت حرکت کند و با برخورد به دیواره‌ی فرم در جهت خلاف برگردد. از زمان‌سنج استفاده کنید.

ـ برنامه‌ای بنویسید که در آن پنجره‌ی برنامه، با فشردن کلید ESC بسته شود.

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    switch (e.KeyCode)
    {
        case Keys.Left:
        case Keys.A:
            picSpacecraft.Left -= 5; break;
        case Keys.Right:
        case Keys.D:
            picSpacecraft.Left += 5; break;
        case Keys.Up:
        case Keys.W:
            picSpacecraft.Top -= 5; break;
        case Keys.Down:
        case Keys.S:
            picSpacecraft.Top += 5; break;
        case Keys.Escape:
            this.Close(); break;       // بستن پنجره با ESC
    }
}
```

اگر در فرم کنترلی (مثل دکمه) فوکوس را بگیرد، رویداد KeyDown فرم رخ نمی‌دهد؛ بنابراین در صورت وجود چنین کنترل‌هایی باید KeyPreview = true شود. در این برنامه که فقط کادر تصویر دارد (فرم فوکوس را دارد) لازم نیست، ولی گذاشتن آن بی‌ضرر است و توصیه می‌شود.

حرکت خودکار با زمان‌سنج و برگشت با برخورد به دیواره

ایران توشه
توشه ای برای موفقیت

```
int dx = 0, dy = 0;                        // جهت حرکت (timer1.Interval = 20, Enabled =
true)
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Left)  { dx = -5; dy = 0; }
    if (e.KeyCode == Keys.Right) { dx = 5;  dy = 0; }
    if (e.KeyCode == Keys.Up)    { dx = 0;  dy = -5; }
    if (e.KeyCode == Keys.Down)  { dx = 0;  dy = 5; }
}
private void timer1_Tick(object sender, EventArgs e)
{
    picSpacecraft.Left += dx;
    picSpacecraft.Top += dy;
    if (picSpacecraft.Left <= 0 || picSpacecraft.Right >= this.ClientSize.Width) dx = -
dx;
    if (picSpacecraft.Top <= 0 || picSpacecraft.Bottom >= this.ClientSize.Height) dy = -
dy;
}
```

۲۲۲ صفحه ۱۹۳ کارگاه ۸

فوکوس روی کدام کنترل قرار دارد؟

عملکرد متد رویداد KeyDown_txtFirstName چیست؟

۴ ـ متد رویداد KeyDown را برای کنترل‌های کادرمتن دیگر هم بنویسید.

آخرین کنترلی که فوکوس دریافت می‌کند، کدام است؟

هنگام اجرا فوکوس روی اولین کنترل به ترتیب TabIndex یعنی txtFirstName است. متد txtFirstName_KeyDown هنگام فشردن Enter در کادر نام، فوکوس را با متد ()Focus به کادر نام خانوادگی منتقل می‌کند.

ایران توشه
توشه ای برای موفقیت

```
private void txtFirstName_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
        txtLastName.Focus();
}
private void txtLastName_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
        txtFatherName.Focus();
}
private void txtFatherName_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
        txtMelliCode.Focus();
}
private void txtMelliCode_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
        btnOk.Focus();            // دکمه‌ی اول
}
```

آخرین کنترل، دکمه‌ای است که کادر کد ملی فوکوس را به آن می‌دهد (در اینجا دکمه‌ی اول، مثلاً btnOk).

۲۲۳ صفحه ۱۹۴ پژوهش

اگر بخواهیم با فوکوس روی هر کادرمتن رنگ زمینه‌ی آن تغییر کند و با از دست دادن فوکوس رنگ آن به حالت اول درآید، از چه رویدادهایی استفاده می‌کنیم؟

از رویدادهای Enter (یا GotFocus) برای دریافت فوکوس و Leave (یا LostFocus) برای از دست دادن فوکوس:

```
private void txtFirstName_Enter(object sender, EventArgs e)
{
    txtFirstName.BackColor = Color.LightYellow;
}
private void txtFirstName_Leave(object sender, EventArgs e)
{
    txtFirstName.BackColor = Color.White;
}
```

۲۲۴ صفحه ۱۹۵ کارگاه ۹

۳ ـ ... چه کلیدهایی در کادرمتن شماره‌ی دانش‌آموزی تایپ می‌شوند؟ عملکرد ویژگی SuppressKeyPress چیست؟

ایران توشه
توشه ای برای موفقیت

۴ ـ لزوم تبدیل e.KeyCode به char چیست؟ چه کلیدهایی در کادرمتن شماره‌ی دانش‌آموزی تایپ می‌شوند؟ آیا می‌توان داده‌ی درون کادرمتن را با کلید Delete یا Backspace حذف کرد؟ آیا می‌توان با کلیدهای جهت‌دار، داده را پیمایش کرد؟

۵ ـ کد مرحله‌ی ۴ را به‌گونه‌ای ویرایش کنید که بتوان عدد ورودی را حذف و پیمایش کرد.

با e.SuppressKeyPress = true هیچ کلیدی در کادر متن تایپ نمی‌شود؛ این ویژگی با true شدن از ورود داده از طریق صفحه‌کلید جلوگیری می‌کند.

e.KeyCode از نوع Keys است ولی متد char.IsDigit نویسه‌ی char می‌گیرد؛ پس باید (char)e.KeyCode نوشته شود. در این حالت فقط رقم‌های ردیف بالای صفحه‌کلید (کد ۴۸ تا ۵۷) تایپ می‌شوند؛ کلیدهای Backspace، Delete و جهت‌دار نیز مسدود می‌شوند (غیرقابل حذف و پیمایش) و ارقام بخش عددی هم پذیرفته نمی‌شوند.

```
private void txtStudentNO_KeyDown(object sender, KeyEventArgs e)
{
    // فقط رقم، به‌همراه Backspace، Delete و کلیدهای جهت‌دار
    if (!char.IsDigit((char)e.KeyCode)
        && e.KeyCode != Keys.Back && e.KeyCode != Keys.Delete
        && e.KeyCode != Keys.Left && e.KeyCode != Keys.Right
        && !(e.KeyCode >= Keys.NumPad0 && e.KeyCode <= Keys.NumPad9))
        e.SuppressKeyPress = true;
}
```

۲۲۵ صفحه ۱۹۶ کارگاه ۹

۶ ـ ... چه نویسه‌هایی در کادرمتن تایپ می‌شود؟

۷ ـ ... برنامه را اجرا کرده، نام AliReza را در کادرمتن وارد کنید. به حروف بزرگ و کوچک توجه کنید. عملکرد ویژگی Handled چیست؟

۸ ـ برنامه را طوری تغییر دهید که بتوان حروف تایپ‌شده را پاک و پیمایش کرد.

۹ ـ ... برنامه را اجرا کرده، نام خانوادگی Ahmadi را در کادرمتن وارد کنید. چه متنی در کادرمتن نوشته می‌شود؟ عملکرد متد رویداد KeyPress_txtLastName چیست؟

در متد KeyDown نام (txtFirstName) فقط حروف الفبا (char.IsLetter) و کلید Backspace پذیرفته می‌شود؛ ارقام و علائم تایپ نمی‌شوند.

در متد KeyPress فقط حروف کوچک (char.IsLower) پذیرفته می‌شود؛ ویژگی Handled اگر true شود نویسه پردازش‌شده تلقی شده و وارد کادر متن نمی‌شود. با ورود AliReza نویسه‌های بزرگ A و R نادیده گرفته می‌شوند و فقط lieza نوشته می‌شود.

```
private void txtFirstName_KeyPress(object sender, KeyPressEventArgs e)
{
    // حروف کوچک، Backspace (کد 8) و بقیه‌ی کلیدهای کنترلی (جهت‌دار) KeyPress ندارند)
    if (!(char.IsLower(e.KeyChar) || e.KeyChar == (char)Keys.Back))
        e.Handled = true;
}
private void txtLastName_KeyPress(object sender, KeyPressEventArgs e)
{
    if ((e.KeyChar >= 'a') && (e.KeyChar <= 'z'))
    {
        int Code = (int)e.KeyChar - 32;
        e.KeyChar = (char)Code;        // تبدیل حرف کوچک به بزرگ
    }
}
```

با ورود Ahmadi در کادر نام خانوادگی، متن AHMADI نوشته می‌شود؛ متد KeyPress_txtLastName هر حرف کوچک انگلیسی را با کم کردن ۳۲ از کد اسکی آن به حرف بزرگ تبدیل می‌کند (اختلاف کد حروف کوچک و بزرگ ۳۲ است).

۲۲۶ صفحه ۱۹۷ فعالیت کارگاهی

نتیجه‌ی خروجی مربوط به مثال‌های جدول ۷ را بنویسید.

جدول ۷ـ متدهای نوع داده‌ی char

نام متد	نوع خروجی	شرح	مثال ۱	مثال ۲
IsDigit	bool	آیا نویسه‌ی ورودی رقم است.	char.IsDigit('A') ← false	char.IsDigit('5') ← true
IsLetter	bool	آیا نویسه‌ی ورودی حرف است.	char.IsLetter('A') ← true	char.IsLetter('@') ← false
IsUpper	bool	آیا نویسه‌ی ورودی حرف بزرگ است.	char.IsUpper('d') ← false	char.IsUpper('M') ← true
IsLower	bool	آیا نویسه‌ی ورودی حرف کوچک است.	char.IsLower('d') ← true	char.IsLower('M') ← false

۲۲۷ صفحه ۱۹۸ کنجکاوی

متغیر از نوع رشته‌ای شامل چه مقادیری می‌شود؟

ثابت رشته‌ای چیست؟

ایران توشه
توشه ای برای موفقیت

نوع داده‌ی name[0] چیست؟ آیا می‌توان این نویسه را مقداردهی کرد؟

چه عملیاتی در واژه‌پردازها روی رشته‌ها انجام می‌شود؟ نرم‌افزارهایی را نام ببرید که داده‌های رشته‌ای در آن استفاده می‌شود.

به وبگاه سامانه‌ی دانش‌آموزی مراجعه کرده‌اید؟ چه داده‌هایی در سامانه، در مورد شما ثبت‌شده است؟

هنگامی که می‌خواهید یک رایانامه (Email) ایجاد کنید چه داده‌هایی باید در سرویس‌دهنده وارد کنید؟ کدام یک از این داده‌ها، رشته‌ای است؟

- متغیر رشته‌ای می‌تواند دنباله‌ای از نویسه‌ها (حروف، ارقام، علائم و فاصله) را نگه دارد؛ مثلاً "AliReza" یا "123" یا رشته‌ی خالی "".
- ثابت رشته‌ای مقدار رشته‌ای است که مستقیم در برنامه نوشته می‌شود و بین دو علامت نقل‌قول قرار دارد؛ مانند "Hello".
- نوع داده‌ی name[0] برابر char است (یک نویسه). نمی‌توان آن را مقداردهی کرد: رشته در سی‌شارپ تغییرناپذیر (immutable) و name[0] فقط‌خواندنی است؛ دستور name[0] = 'B'; خطای کامپایل می‌دهد.
- عملیات در واژه‌پردازها روی رشته‌ها: جست‌وجو و جایگزینی، حذف، درج، تبدیل حروف به بزرگ/کوچک، شمارش نویسه‌ها و واژه‌ها، مرتب‌سازی، مقایسه. نرم‌افزارها: Notepad، Word، مرورگرها، موتورهای جست‌وجو، پیام‌رسان‌ها، پایگاه‌های داده.
- داده‌های سامانه‌ی دانش‌آموزی: نام و نام خانوادگی، نام پدر، کد ملی، شماره‌ی دانش‌آموزی، نشانی، نمرات و پایه؛ داده‌های رایانامه: نام، نام خانوادگی، نام کاربری، گذرواژه، شماره‌ی تلفن؛ بیشتر آنها (نام، نام کاربری، گذرواژه، نشانی و حتی کد ملی) رشته‌ای هستند.

ایران توشه
توشه‌ای برای موفقیت

۲۲۸ صفحه ۱۹۹ کارگاه ۱۰

۳ ـ ... پس از کلیک روی دکمه چه مقداری در کادرمتن قرار می‌گیرد؟ عملکرد ویژگی Length چیست؟

۴ ـ ... چه مقداری در کادرمتن قرار می‌گیرد؟ عملکرد متد ToUpper چیست؟ عملکرد متد ToLower را بررسی کنید.

۵ ـ ... عملکرد متد IndexOf چیست؟ اگر به‌جای ۰ عدد ۵ قرار دهید، خروجی چه می‌شود؟ اگر در مرحله‌ی ۵ به‌جای Reza، واژه‌ی Ahmad را بنویسید، خروجی چه می‌شود؟

عملکرد متد LastIndexOf چیست؟

```
string name = "AliReza";
txtResult.Text = name.Length.ToString();    // 7
txtResult.Text = name.ToUpper();            // ALIREZA
txtResult.Text = name.ToLower();            // alireza

string name2 = "AliReza Rezaee";
txtResult.Text = name2.IndexOf("Reza", 0).ToString();   // 3
txtResult.Text = name2.IndexOf("Reza", 5).ToString();   // 8
txtResult.Text = name2.IndexOf("Ahmad", 0).ToString();  // -1
txtResult.Text = name2.LastIndexOf("Reza").ToString();  // 8
```

- Length: تعداد نویسه‌های رشته را برمی‌گرداند؛ برای AliReza مقدار 7.
- ToUpper(): همه‌ی حروف را بزرگ می‌کند (ALIREZA) و ToLower() همه‌ی حروف را کوچک می‌کند (alireza).
- IndexOf("Reza", 0): اندیس اولین رخداد واژه‌ی Reza را از اندیس ۰ به بعد برمی‌گرداند: 3. با شروع از ۵، رخداد اول نادیده گرفته می‌شود و رخداد دوم (Rezaee) در اندیس 8 پیدا می‌شود. اگر واژه پیدا نشود (مثل Ahmad) نتیجه 1- است.
- LastIndexOf: اندیس آخرین رخداد واژه را (جست‌وجو از انتها) برمی‌گرداند: برای Reza در AliReza Rezaee برابر 8.

۲۲۹ صفحه ۲۰۰ کارگاه ۱۱

اگر کاربر در کادرمتن نام کاربری ADMIN را وارد کند، چه پیامی دریافت می‌کند؟

۴ ـ برنامه را بهگونه‌ای ویرایش کنید که ورود حروف کوچک یا بزرگ در نام کاربری یا گذرواژه، تأثیری در اعتبارسنجی کاربر نداشته باشد.

برنامه را اجرا کنید. نام کاربری را با حروف کوچک و بزرگ وارد کنید. (CompareTo)

۵ ـ فاصله‌ی ابتدا و انتهای نام کاربری را حذف کنید.

با ورود ADMIN پیام «نام کاربری یا گذرواژه اشتباه است» نمایش داده می‌شود؛ زیرا مقایسه‌ی == بین رشته‌ها به حروف بزرگ و کوچک حساس است.

ایران توشه
توشه‌ای برای موفقیت

```
private void btnLogin_Click(object sender, EventArgs e)
{
    // بدون توجه به حروف بزرگ و کوچک، و حذف فاصله‌ی ابتدا و انتها
    string user = txtUserName.Text.Trim().ToLower();
    string pass = txtPassword.Text.Trim().ToLower();
    if (user == "admin" && pass == "123")
        MessageBox.Show("ورود شما با موفقیت انجام شد");
    else
    {
        string s = "نام کاربری یا گذرواژه اشتباه است";
        MessageBox.Show(s);
    }
}

// با CompareTo (مقدار 0 یعنی دو رشته برابرند):
if (txtUserName.Text.Trim().ToLower().CompareTo("admin") == 0 &&
    txtPassword.Text.Trim().ToLower().CompareTo("123") == 0)
    MessageBox.Show("ورود شما با موفقیت انجام شد");
```

متد CompareTo اگر رشته‌ها برابر باشند ۰، اگر رشته‌ی اول در ترتیب الفبایی بعد از دومی باشد مقدار مثبت (۱) و در غیر این صورت منفی (۱-) برمی‌گرداند؛ به حروف بزرگ و کوچک حساس است (پس با ToLower هم‌ترکیب شده است). متد Trim() فاصله‌های ابتدا و انتهای رشته را حذف می‌کند تا اعتبارسنجی با فاصله‌ی اضافی خراب نشود.

۲۳۰ صفحه ۲۰۱ فعالیت کارگاهی

ستون‌های «عملکرد» و «نوع داده‌ی خروجی» را تکمیل کنید. اگر "sample = "AliReza باشد، حاصل دستورات در ستون «مثال» را بنویسید.

جدول ۸ـ شرح متدهای رشته‌ای ("sample = "AliReza

ایران توشه
توشه ای برای موفقیت

نام متد	عملکرد	نوع خروجی	مثال	حاصل
ToUpper	تبدیل همه‌ی حروف به بزرگ	string	sample.ToUpper()	ALIREZA
ToLower	تبدیل همه‌ی حروف به کوچک	string	sample.ToLower()	alireza
IndexOf	اندیس اولین رخداد یک رشته از مکان مشخص (۱- اگر نباشد)	int	sample.IndexOf("Reza",0)	3
Replace	جایگزینی همه‌ی رخدادهای یک رشته با رشته‌ی دیگر	string	sample.Replace("Ali","Mohammad")	MohammadReza
Remove	حذف تعداد معینی نویسه از یک مکان	string	sample.Remove(2,3)	Alza
Insert	درج یک رشته در مکان مشخص	string	sample.Insert(4,"Mohammad")	AliRMohammadeza
Format	قالب‌بندی و ساخت رشته با جانگهدارها	string	string.Format("{0} is a student", sample)	AliReza is a student
CompareTo		int	sample.CompareTo("Ali")	

ایران توشه
توشه ای برای موفقیت

	مقایسه‌ی دو رشته؛ ۰ برابر، مثبت یا منفی بر اساس ترتیب			۱ (عدد مثبت؛ AliReza بزرگ‌تر از Ali)
Trim	حذف فاصله‌های ابتدا و انتهای رشته	string	" Ali ".Trim()	Ali

۲۳۱ صفحه ۲۰۱ فیلم

پس از مشاهده‌ی فیلم «برنامه‌ی جست‌وجو و جایگزینی واژه» را بنویسید. (فیلم شماره‌ی ۱۱۰۹) کارگاه ۱۲: به‌کارگیری متدهای IndexOf و Replace

برنامه: یک کادر متن چندخطی (txtText)، کادر «واژه‌ی مورد جست‌وجو» (txtFind)، کادر «واژه‌ی جایگزین» (txtReplace) و دکمه‌های btnFind و btnReplace (فیلم در دسترس نبود؛ طراحی ساده):

```
private void btnFind_Click(object sender, EventArgs e)
{
    int index = txtText.Text.IndexOf(txtFind.Text);
    if (index >= 0)
    {
        txtText.Focus();
        txtText.Select(index, txtFind.Text.Length);   // انتخاب واژه‌ی پیداشده
    }
    else
        MessageBox.Show("واژه پیدا نشد");
}
private void btnReplace_Click(object sender, EventArgs e)
{
    if (txtText.Text.IndexOf(txtFind.Text) >= 0)
        txtText.Text = txtText.Text.Replace(txtFind.Text, txtReplace.Text);
    else
        MessageBox.Show("واژه پیدا نشد");
}
```

این جواب ممکن است دقیق نباشد.

ایران توشه
توشه ای برای موفقیت

به پروژه‌ی کتابخانه منو اضافه کنید: با انتخاب منوی «خروج» کادر پیام تأیید خروج نمایش داده شده، در صورت تأیید کاربر برنامه تمام شود؛ با کلیک روی منوی «راهنما» فرم «درباره‌ی ما» ظاهر شود؛ گزینه‌ی «ورود» منوی «مدیر» برای دریافت نام کاربری و گذرواژه‌ی مدیر است؛ منوی «ویرایش» برای انتخاب رنگ و قلم کادرهای متن در تمام سربرگ‌هاست. موارد زیر را برای تکمیل پروژه اضافه کنید: دسترسی به سربرگ‌های فعال با صفحه‌کلید؛ ComboBox برای محدود کردن سال چاپ کتاب؛ فقط نویسه‌های حرفی در نام مؤلف؛ حذف فضای خالی در کادر جست‌وجو؛ غیرفعال بودن دکمه‌ی Search تا زمانی که عبارتی وارد نشده است.

```
// منوی خروج
private void mnuExit_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("خروج" ,"آیا مایل به خروج هستید؟", MessageBoxButtons.YesNo,
        MessageBoxIcon.Question) == DialogResult.Yes)
        Application.Exit();
}
// منوی راهنما ← درباره‌ی ما
private void mnuAbout_Click(object sender, EventArgs e)
{
    AboutForm frm = new AboutForm();
    frm.ShowDialog();
}
// منوی مدیر ← ورود
private void mnuAdminLogin_Click(object sender, EventArgs e)
{
    LoginForm frm = new LoginForm();
    frm.ShowDialog();
    if (frm.succeeded)
        tabMember.Enabled = tabBook.Enabled = true;
}
// منوی ویرایش ← رنگ و قلم کادرهای متن (همه‌ی سربرگ‌ها)
private void mnuFont_Click(object sender, EventArgs e)
{
    if (fontDialog1.ShowDialog() == DialogResult.OK)
        foreach (TabPage tp in tabControl1.TabPages)
            foreach (Control c in tp.Controls)
                if (c is TextBox) c.Font = fontDialog1.Font;
}
private void mnuColor_Click(object sender, EventArgs e)
{
    if (colorDialog1.ShowDialog() == DialogResult.OK)
        foreach (TabPage tp in tabControl1.TabPages)
            foreach (Control c in tp.Controls)
                if (c is TextBox) c.BackColor = colorDialog1.Color;
}
```

ایران توشه
توشه اول دانش‌آموزی

```
// تکمیل پروژه
// ۱) دسترسی به سربرگ‌ها با صفحه‌کلید: Ctrl+1 / Ctrl+2 / Ctrl+3
private void MainForm_KeyDown(object sender, KeyEventArgs e)   // KeyPreview = true
{
    if (e.Control && e.KeyCode == Keys.D1 && tabMember.Enabled) tabControl1.SelectedTab
= tabMember;
    if (e.Control && e.KeyCode == Keys.D2 && tabBook.Enabled)   tabControl1.SelectedTab
= tabBook;
}
// ۲) سال چاپ با ComboBox (DropDownStyle = DropDownList؛ سال‌ها در Items)
// ۳) نام مؤلف فقط حروف
private void txtAuthor_KeyPress(object sender, KeyPressEventArgs e)
{
    if (!(char.IsLetter(e.KeyChar) || e.KeyChar == ' ' || e.KeyChar == (char)Keys.Back))
        e.Handled = true;
}
// ۴ و ۵) جست‌وجو
private void txtSearch_TextChanged(object sender, EventArgs e)
{
    btnSearch.Enabled = txtSearch.Text.Trim().Length > 0;   // فعال فقط وقتی عبارتی وارد
شده
}
private void btnSearch_Click(object sender, EventArgs e)
{
    string key = txtSearch.Text.Trim();                      // حذف فاصله‌ی ابتدا و
انتها
    // جست‌وجو در آرایه‌ی کتاب‌ها با key ...
}
```

طرح دقیق فرم کتابخانه در دسترس نبود؛ نام کنترل‌ها پیشنهادی هستند.

ایران توشه
توشه ای برای موفقیت

گام به گام توسعه برنامه‌سازی و پایگاه داده - پودمان پنجم

مدیریت پایگاه داده

فهرست

۲۳۳ صفحه ۲۰۷ · فعالیت گروهی — پرونده و انواع آن ۱۷۳

۲۳۴ صفحه ۲۰۹ · کارگاه ۱ — ایجاد پرونده و نوشتن در آن (WriteAllText و AppendAllText) ۱۷۳

۲۳۵ صفحه ۲۱۰ · کارگاه ۲ — پروژه Notepad: ذخیره (Save As / Save / New) ۱۷۴

۲۳۶ صفحه ۲۱۳ · کارگاه ۳ — خواندن پرونده‌ی متنی با ReadAllText ۱۷۷

۲۳۷ صفحه ۲۱۴ · کارگاه ۴ — خواندن پرونده‌ی متنی با ReadAllLines ۱۷۸

۲۳۸ صفحه ۲۱۶ · کارگاه ۵ — مدیریت خطا با دستور try-catch ۱۷۹

۲۳۹ صفحه ۲۱۷ · فعالیت کارگاهی — جدول ۱: متدهای کلاس Directory ۱۸۰

۲۴۰ صفحه ۲۱۸ · فعالیت کارگاهی — جدول ۲: متدهای کلاس Path ۱۸۰

۲۴۱ صفحه ۲۱۹ · کارگاه ۶ — کاربرد کلاس‌های Directory و Path ۱۸۱

۲۴۲ صفحه ۲۲۱ · فعالیت گروهی — کاربرد پایگاه داده (دفترچه مخاطبان) ۱۸۲

۲۴۳ صفحه ۲۲۲ · فعالیت کارگاهی — ایجاد پایگاه داده (فروشگاه فرهنگیان) ۱۸۳

۲۴۴ صفحه ۲۲۳ · فعالیت کارگاهی — طراحی پایگاه داده‌ی بازی و ساخت MyShop ۱۸۴

ایران توشه
توشه ای برای موفقیت

۲۴۵ صفحه ۲۲۴ · کارگاه ۷ — ارتباط با پایگاه داده از طریق wizard (واژه‌نامه) ۱۸۵
۲۴۶ صفحه ۲۲۶ · کارگاه ۸ — سفارشی کردن DataGridView و جدول ۳ ۱۸۶
۲۴۷ صفحه ۲۲۷ · کارگاه ۹ — مقیدسازی کنترل‌ها (Binding) ۱۸۸
۲۴۸ صفحه ۲۲۹ · کارگاه ۱۰ — بارگذاری داده و پیمایش رکوردها ۱۸۹
۲۴۹ صفحه ۲۳۰ · کارگاه ۱۱ — دسترسی به خانه‌های DataGridView ۱۹۰
۲۵۰ صفحه ۲۳۲ · کارگاه ۱۲ — جستجو در جدول (SplitContainer و پرسوجوی WHERE) ۱۹۱
۲۵۱ صفحه ۲۳۵ · کارگاه ۱۳ — ورود کاربر (Login) به برنامه‌ی فروشگاه ۱۹۲
۲۵۲ صفحه ۲۳۷ · کارگاه ۱۴ — درج یک رکورد در جدول (INSERT) ۱۹۴
۲۵۳ صفحه ۲۳۹ · کارگاه ۱۵ — حذف رکورد از جدول (DELETE) ۱۹۶
۲۵۴ صفحه ۲۴۰ · کارگاه ۱۶ — ویرایش رکورد در جدول (UPDATE) ۱۹۷
۲۵۵ صفحه ۲۴۱ · کارگاه ۱۷ — درج و حذف سطر در DataGridView (برگه خرید) ۱۹۸
۲۵۶ صفحه ۲۴۵ · کارگاه ۱۸ — مدیریت خطا در برنامه‌ی پایگاه داده ۲۰۰
۲۵۷ صفحه ۲۴۶ · کارگاه ۱۹ — توسعه‌ی پروژه‌ی فروشگاه (TryParse، برگه خرید، کاربران) ۲۰۲
۲۵۸ صفحه ۲۴۷ · پروژه فاز ۵ — اتصال پروژه‌ی کتابخانه به پایگاه داده ۲۰۳

ایران توشه
توشه‌ای برای موفقیت

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

۲۳۳ صفحه ۲۰۷ فعالیت گروهی

در جدول زیر با کمک هم‌گروهی خود مشخص کنید، کدام پرونده‌ها از نوع دودویی و کدام پرونده‌ها از نوع متنی هستند. سعی کنید این نوع از پرونده‌ها را با برنامه Notepad باز کنید. چه تفاوتی مشاهده می‌کنید؟ (Adobe pdf، html Document، Microsoft Word Document، MP3 Audio File، C# source code)

نوع پرونده‌ها

نوع پرونده	پسوند	دودویی / متنی
Adobe pdf	.pdf	دودویی
html Document	.html	متنی
Microsoft Word Document	.docx	دودویی
MP3 Audio File	.mp3	دودویی
C# source code	.cs	متنی

پرونده‌های متنی (html و cs) در Notepad به‌صورت نویسه‌های خوانا (کد و متن) نمایش داده می‌شوند و می‌توان آنها را ویرایش کرد. پرونده‌های دودویی (pdf، docx، mp3) در Notepad فقط مجموعه‌ای از نویسه‌های نامفهوم و علامت‌های عجیب نشان می‌دهند، زیرا داده‌ها به همان شکل حافظه (بایت) ذخیره شده‌اند و باید با برنامه‌ی مخصوص خودشان باز شوند.

ایران‌توشه
توشه ای برای موفقیت

۲۳۴ صفحه ۲۰۹ کارگاه ۱

۴ ـ پرونده‌ی Info.txt را باز کنید. پوشه‌ی Debug پروژه را باز کنید. آیا پرونده‌ی متنی Info.txt در آن پوشه قرار دارد؟ محتوای این پرونده‌ی متنی چیست؟

۵ ـ بعد از اجرای مجدد برنامه، محتوای پرونده‌ی متنی Info.txt را بررسی کنید. نام «حسن» را وارد کرده، دکمه «ثبت» را کلیک کنید. آیا نام «علی» در پرونده‌ی Info.txt وجود دارد؟

۶ ـ دستور نوشتن در پرونده را به‌صورت AppendAllText تغییر دهید. نام‌های «حسن» و «علی» را وارد کنید. آیا هر دو نام در پرونده ذخیره شده است؟ متدهای WriteAllText و AppendAllText هر دو رشته‌ی data را در پرونده می‌نویسند. این دو متد چه تفاوتی با هم دارند؟

۸ ـ کدهای برنامه را به‌صورت زیر تغییر دهید. پرونده‌ی Info.txt را حذف کنید. چند نام دلخواه وارد و ثبت کنید. چه داده‌هایی در پرونده ذخیره می‌شوند؟

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

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

۴ ـ بله. چون نام پرونده بدون مسیر نوشته شده، پرونده در مسیر جاری برنامه، یعنی پوشه‌ی bin\Debug پروژه ساخته می‌شود. محتوای آن همان نام واردشده است: علی.

۵ ـ خیر. WriteAllText اگر پرونده از قبل وجود داشته باشد، محتوای قبلی را پاک می‌کند و متن جدید را جایگزین می‌کند؛ بنابراین فقط «حسن» در پرونده می‌ماند.

۶ ـ بله، هر دو نام ذخیره می‌شود. تفاوت: WriteAllText محتوای قبلی پرونده را پاک کرده و متن جدید را می‌نویسد، ولی AppendAllText متن جدید را به انتهای محتوای قبلی اضافه می‌کند و اطلاعات قبلی حفظ می‌شود. اگر پرونده وجود نداشته باشد، هر دو آن را می‌سازند.

```
private void btnSave_Click(object sender, EventArgs e)
{
    string data = String.Format("{0}\t{1}\r\n",
                  txtFirstName.Text, txtLastName.Text);
    File.AppendAllText("Info.txt", data);
}
```

۸ ـ در هر بار ثبت، یک خط جدید به پرونده افزوده می‌شود که شامل نام، یک فاصله‌ی tab و نام خانوادگی است؛ مثلاً علی رضایی. دنباله‌ی t\ نویسه‌ی Tab و r\n رفتن به خط بعد است.

۹ ـ کافی است قالب رشته را تغییر دهیم:

```
string data = String.Format("{0}\t{1}\t{2}\t{3}\r\n",
    txtFirstName.Text, txtLastName.Text,
    txtFatherName.Text, txtNationalCode.Text);
File.AppendAllText("Info.txt", data);
```

کنجکاوی: نام پرونده را همراه مسیر کامل می‌نویسیم، مثلاً:

```
File.AppendAllText(@"D:\Data\Info.txt", data);
```

پوشه‌ی مقصد باید از قبل وجود داشته باشد، وگرنه خطای DirectoryNotFoundException رخ می‌دهد (می‌توان از Directory.CreateDirectory برای ساخت آن استفاده کرد).

۲۳۵ صفحه ۲۱۰ کارگاه ۲

۴ ـ ویژگی Filter کادر محاوره‌ای SaveFileDialog را طوری تنظیم کنید که پرونده‌های متنی را نمایش دهد.

ایران توشه
توشه ای برای موفقیت

۵ ـ متد رویداد کلیک گزینه Save As منوی File را بنویسید. برنامه را اجرا کنید؛ متن دلخواهی بنویسید و در یک پرونده‌ی متنی جدید ذخیره کنید.

۶ ـ برای گزینه‌های Save و New منوی File کدنویسی کنید.

۷ ـ برنامه را توسعه دهید. برنامه‌ای بنویسید که قابلیت ذخیره‌ی مخاطبان در دفترچه تلفن را داشته باشد (شکل ۴). به کمک متد AppendAllText همه‌ی اطلاعات یک مخاطب را در یک خط به پرونده‌ی متنی ContactList.txt اضافه کنید.

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

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

۴ ـ در پنجره‌ی Properties مقدار Filter را روی Text files|*.txt|پرونده متنی تنظیم می‌کنیم (نیمه‌ی اول عنوان نمایشی و پس از خط عمودی الگوی پرونده‌هاست).

```
saveFileDialog1.Filter = "Text files|*.txt";
```

۵ ـ کد گزینه‌ی Save As (برای نگهداری مسیر، متغیر documentFilePath در سطح کلاس تعریف شده است):

```
private void saveAsToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        documentFilePath = saveFileDialog1.FileName;
        File.WriteAllText(documentFilePath, txtMainDocument.Text);
    }
}
```

۶ ـ گزینه‌ی Save: اگر پرونده قبلاً ذخیره شده باشد (مسیر خالی نیست) مستقیم در همان مسیر ذخیره می‌شود، وگرنه مانند Save As عمل می‌کند. گزینه‌ی New: متن را پاک کرده و مسیر را خالی می‌کند.

```
private void saveToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (documentFilePath == "")
        saveAsToolStripMenuItem_Click(sender, e);
    else
        File.WriteAllText(documentFilePath, txtMainDocument.Text);
}
private void newToolStripMenuItem_Click(object sender, EventArgs e)
{
    txtMainDocument.Clear();
    documentFilePath = "";
}
```

۷ ـ دفترچه تلفن: فرم شامل کادرمتن‌های txtFirstName، txtLastName، txtPhone، txtAddress، دکمه‌ی «افزودن به فهرست» (btnAdd) و کادرفهرست lstContacts است. هر مخاطب در یک خط ذخیره می‌شود:

```
private void btnAdd_Click(object sender, EventArgs e)
{
    string contact = String.Format("{0}\t{1}\t{2}\t{3}\r\n",
        txtFirstName.Text, txtLastName.Text,
        txtPhone.Text, txtAddress.Text);
    File.AppendAllText("ContactList.txt", contact);
    lstContacts.Items.Add(contact);
}
```

توسعه‌ی ۱: برای تأیید کاربر، یک متغیر منطقی isChanged در رویداد TextChanged کادرمتن true می‌شود و پیش از New یا خروج، پرسیده می‌شود:

```
private bool isChanged = false;
private void txtMainDocument_TextChanged(object sender, EventArgs e)
{
    isChanged = true;
}
private bool ConfirmSave()
{
    if (!isChanged) return true;
    DialogResult r = MessageBox.Show("Save changes?", "Notepad",
        MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);
    if (r == DialogResult.Cancel) return false;
    if (r == DialogResult.Yes) saveToolStripMenuItem_Click(null, null);
    return true;
}
private void newToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (!ConfirmSave()) return;
    txtMainDocument.Clear();
    documentFilePath = "";
    isChanged = false;
}
private void exitToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (ConfirmSave()) Application.Exit();
}
```

توسعه‌ی ۲ (بازی اعصاب‌سنج): در پایان بازی، نام کاربر (از یک کادرمتن یا InputBox/فرم کوچک) و امتیاز را در پرونده می‌نویسیم:

```
string line = String.Format("{0}\t{1}\r\n", txtPlayer.Text, score);
File.AppendAllText("Scores.txt", line);
```

این جواب ممکن است دقیق نباشد. (طراحی دقیق فرم دفترچه تلفن و بازی اعصاب‌سنج به شکل‌های کتاب بستگی دارد؛ کد بالا نمونه‌ی معقولی از آن است.)

ایران توشه
توشه ای برای موفقیت

۳ ـ متد رویداد کلیک گزینه Open را بنویسید. برنامه را اجرا کرده، یک پرونده‌ی متنی دلخواه را باز کنید.

عملکرد متد رویداد کلیک گزینه Open چیست؟

۴ ـ برنامه را توسعه دهید. کد مرحله‌ی ۳ را طوری تغییر دهید تا قبل از بازکردن پرونده‌ی جدید، کادر محاوره‌ای ذخیره‌ی پرونده‌ی بازشده‌ی قبلی نمایش داده شود.

فعالیت منزل: با تنظیم ویژگی راست‌به‌چپ فرم و ترجمه‌ی گزینه‌های منو به زبان فارسی، برنامه Notepad را فارسی کنید. برای گزینه‌های منوی Edit برنامه Notepad کدنویسی کنید. (راهنمایی: از ویژگی‌های SelectionStart، SelectedText و SelectionLength استفاده کنید.)

عملکرد: با کلیک روی Open کادر محاوره‌ای انتخاب پرونده نمایش داده می‌شود؛ اگر کاربر پرونده‌ای را انتخاب و OK کند، مسیر آن در documentFilePath ذخیره شده و کل محتوای پرونده با File.ReadAllText خوانده و در کادرمتن txtMainDocument نمایش داده می‌شود.

```
private void openToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (isChanged)
    {
        DialogResult r = MessageBox.Show("Save changes?", "Notepad",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (r == DialogResult.Yes)
            saveToolStripMenuItem_Click(sender, e);
    }
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        documentFilePath = openFileDialog1.FileName;
        txtMainDocument.Text = File.ReadAllText(documentFilePath);
        isChanged = false;
    }
}
```

فعالیت منزل: برای فارسی‌کردن، ویژگی‌های RightToLeft = Yes و RightToLeftLayout = True فرم را تنظیم کرده و Text گزینه‌ها را فارسی می‌کنیم (پرونده، ویرایش، جدید، باز کردن، ذخیره، ذخیره به عنوان، خروج، برش، کپی، چسباندن، انتخاب همه). کد منوی Edit:

ایران توشه
توشه‌ای برای موفقیت

```
private void cutToolStripMenuItem_Click(object sender, EventArgs e)
{ txtMainDocument.Cut(); }
private void copyToolStripMenuItem_Click(object sender, EventArgs e)
{ txtMainDocument.Copy(); }
private void pasteToolStripMenuItem_Click(object sender, EventArgs e)
{ txtMainDocument.Paste(); }
private void selectAllToolStripMenuItem_Click(object sender, EventArgs e)
{ txtMainDocument.SelectAll(); }
private void undoToolStripMenuItem_Click(object sender, EventArgs e)
{ txtMainDocument.Undo(); }
private void deleteToolStripMenuItem_Click(object sender, EventArgs e)
{
    int start = txtMainDocument.SelectionStart;
    int len = txtMainDocument.SelectionLength;
    if (len > 0)
        txtMainDocument.Text = txtMainDocument.Text.Remove(start, len);
    txtMainDocument.SelectionStart = start;
}
```

SelectedText متن انتخاب‌شده، SelectionStart شماره‌ی نویسه‌ی ابتدای انتخاب و SelectionLength طول آن است.

۲۳۷ صفحه ۲۱۴ کارگاه ۴

۲ ـ متد رویداد Form1_Load را بنویسید. برنامه را اجرا کنید. چه اطلاعاتی را در فهرست مشاهده می‌کنید؟ با تعریف یک نقطه‌ی توقف روی دومین خط متد رویداد Load فرم، برنامه را با کلید F11 گام‌به‌گام اجرا کنید. رشته‌ی data حاوی چه مقادیری است؟ عملکرد متد ReadAllLines چیست؟ عملکرد متد AddRange چیست؟

۳ ـ کد مرحله‌ی ۲ و ۳ را با هم مقایسه کنید. برنامه را اجرا کنید. تفاوت متد ReadAllLines با متد ReadAllText در چیست؟

۴ ـ پرونده‌ی ContactList.txt را از پوشه‌ی Debug برنامه حذف کنید. برنامه را اجرا کنید. چه خطایی رخ می‌دهد؟ چرا؟

۵ ـ کد با دستور if و File.Exists را بنویسید و اجرا کنید. نوع خروجی متد Exists چیست؟

۲ ـ در فهرست lstContacts، هر خط پرونده یک عنصر جداگانه است؛ یعنی هر مخاطب (نام، نام خانوادگی، تلفن و نشانی) در یک سطر نمایش داده می‌شود. آرایه‌ی data به تعداد خطوط پرونده عنصر دارد و هر عنصر یک خط از پرونده است. ReadAllLines پرونده را باز می‌کند، خط‌به‌خط می‌خواند، می‌بندد و خطوط را در یک آرایه‌ی رشته‌ای برمی‌گرداند. AddRange همه‌ی عناصر آرایه را یک‌جا به مجموعه‌ی Items کادرفهرست اضافه می‌کند.

ایران توشه
توشه ای برای موفقیت

۳ ـ ReadAllText کل پرونده را در یک رشته‌ی واحد برمی‌گرداند و در فهرست همه‌ی مخاطبان به‌صورت یک عنصر (در یک سطر) نمایش داده می‌شوند؛ اما ReadAllLines پرونده را به آرایه‌ای از خطوط تقسیم می‌کند و هر خط یک عنصر فهرست می‌شود.

۴ ـ خطای FileNotFoundException (پرونده پیدا نشد) رخ می‌دهد و برنامه متوقف می‌شود، زیرا پرونده‌ای با این نام در مسیر جاری وجود ندارد که خوانده شود.

```
private void Form1_Load(object sender, EventArgs e)
{
    if (File.Exists("ContactList.txt"))
        lstContacts.Items.AddRange(File.ReadAllLines("ContactList.txt"));
    else
        MessageBox.Show("Contact file not found", "Error",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
}
```

۵ ـ متد File.Exists مسیر پرونده را می‌گیرد و خروجی آن از نوع bool است: اگر پرونده موجود باشد true و در غیر این صورت false برمی‌گرداند. با این کار پیش از خواندن، از وجود پرونده مطمئن می‌شویم و برنامه دچار خطا نمی‌شود.

۲۳۸ صفحه ۲۱۶ کارگاه ۵

۲ ـ کد متد رویداد کلیک دکمه‌ی «افزودن به فهرست» را به‌صورت زیر تغییر دهید (ذخیره در a:\ContactList.txt). برنامه را اجرا کنید. چه خطایی مشاهده می‌کنید؟

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

۴ ـ با استفاده از ساختار try-catch خطای رخ‌داده را مدیریت کنید. برنامه را اجرا کنید. اجرای برنامه در این حالت چه تفاوتی با حالت قبل دارد؟

پیش از آن (فیلم): رضا هر روز با دوچرخه به محل کار می‌رود. اگر دوچرخه‌ی او دچار مشکل فنی شود، آیا بروز این مشکل اجرای برنامه‌ی روزانه‌ی او را متوقف می‌کند؟

ایران توشه
توشه‌ای برای موفقیت

پیش‌پرسش: خیر؛ او راه‌حل جایگزین دارد (اتوبوس یا مترو). در برنامه نیز با try-catch برای حالت خطا راه‌حل جایگزین تعریف می‌کنیم.

۲ ـ اگر درایو A: (فلاپی/فلش) وجود نداشته باشد، برنامه با خطای زمان اجرا از نوع DirectoryNotFoundException یا IOException متوقف می‌شود. نمونه‌ی پیام: Could not find a part of the path 'a:\ContactList.txt'.

۳ ـ معنی فارسی: «بخشی از مسیر 'a:\ContactList.txt' پیدا نشد» (یعنی درایو یا پوشه‌ی موردنظر وجود ندارد).

```
string contact = string.Format("{0}\t{1}\t{2}\t{3}\t",
    txtFirstName.Text, txtLastName.Text,
    txtPhone.Text, txtAddress.Text);
try
{
    File.AppendAllText("a:\\ContactList.txt", contact);
}
catch
{
    MessageBox.Show("Operation failed. Please try again.", "Error",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
}
```

۴ ـ در حالت قبل برنامه با پنجره‌ی خطا و توقف اجرا مواجه می‌شد؛ اکنون به‌جای توقف، کنترل برنامه به بخش catch می‌رود، پیام «اجرای عملیات امکان‌پذیر نیست. لطفاً یک بار دیگر سعی کنید» نمایش داده می‌شود و برنامه به کار خود ادامه می‌دهد.

۲۳۹ صفحه ۲۱۷ فعالیت کارگاهی

با استفاده از راهنمای مایکروسافت جدول ۱ را تکمیل کنید. (متدهای GetDirectories، GetFiles، Exists)

جدول ۱ـ عملکرد برخی از متدهای کلاس Directory

نام متد	نوع ورودی	نوع خروجی	عملکرد
GetDirectories	رشته (مسیر)	آرایه‌ی رشته‌ای	نام و مسیر همه‌ی زیرپوشه‌های مسیر ورودی را در یک آرایه برمی‌گرداند.
GetFiles	رشته (مسیر)	آرایه‌ی رشته‌ای	نام و مسیر همه‌ی پرونده‌های مسیر ورودی را در یک آرایه برمی‌گرداند.
Exists	رشته (مسیر)	bool	بررسی می‌کند پوشه‌ای با این مسیر وجود دارد یا نه (true یا false).

۲۴۰ صفحه ۲۱۸ فعالیت کارگاهی

با استفاده از راهنمای مایکروسافت جدول ۲ را تکمیل کنید. (GetFileName، Combine)

جدول ۲ـ عملکرد برخی از متدهای کلاس Path

ایران توشه
توشه‌ای برای موفقیت

نام متد	نوع ورودی	نوع خروجی	عملکرد
GetFileName	رشته (مسیر)	رشته	نام و پسوند پرونده‌ی ورودی را برمی‌گرداند.
Combine	دو رشته (مسیر) یا بیشتر	رشته	رشته‌های ورودی را با جداکننده‌ی مسیر (\) به هم می‌چسباند و یک مسیر کامل می‌سازد.

نمونه (اجراشده با mono): Path.GetFileName(@"C:\Users\Ali\Info.txt") مقدار Info.txt، و Path.Combine(@"C:\Users", "Ali", "a.txt") مقدار C:\Users\Ali\a.txt را برمی‌گرداند. متدهای GetExtension (مثلاً .txt) و GetFileNameWithoutExtension (مثلاً Info) نیز در این کلاس هستند.

۲۴۱ صفحه ۲۱۹ **کارگاه ۶**

۴ ـ متد رویداد کلیک دکمه‌ی «فهرست پوشه‌ها» را بنویسید. برنامه را اجرا کنید و روی دکمه «فهرست پوشه‌ها» کلیک کنید. چه عناصری در فهرست مشاهده می‌کنید؟

۵ ـ کد مرحله‌ی ۴ را تغییر دهید تا مسیر directory را از کادرمتن دریافت کند.

۶ و ۷ ـ متد رویداد کلیک دکمه‌ی «فهرست پرونده‌ها» را بنویسید. چه عناصری در فهرست مشاهده می‌کنید؟ سپس مسیر را از کادرمتن دریافت کند.

۸ ـ کد مرحله‌ی ۴ و ۸ را با هم مقایسه کنید. ۹ ـ متد دکمه‌ی «فهرست پرونده‌ها» را ویرایش کنید تا نام پرونده‌ها بدون مسیر نمایش داده شود.

۴ ـ نام و مسیر کامل همه‌ی پوشه‌های درون درایو C:\ (مثل C:\Windows، C:\Users، C:\Program Files و ...) در فهرست نمایش داده می‌شود. به‌همین ترتیب در مرحله‌ی ۶ نام و مسیر پرونده‌های موجود مستقیماً در C:\ (مثل pagefile.sys) نمایش داده می‌شود.

۵ و ۷ ـ برای دریافت مسیر از کادرمتن txtPath:

```
private void btnShowDirectories_Click(object sender, EventArgs e)
{
    string directory = txtPath.Text;
    lstBoard.Items.Clear();
    if (Directory.Exists(directory))
        lstBoard.Items.AddRange(Directory.GetDirectories(directory));
}
private void btnShowFiles_Click(object sender, EventArgs e)
{
    string directory = txtPath.Text;
    lstBoard.Items.Clear();
    if (Directory.Exists(directory))
        lstBoard.Items.AddRange(Directory.GetFiles(directory));
}
```

ایران توشه
توشه‌ای برای موفقیت

۸ ـ در کد مرحله‌ی ۴ نام کامل (به‌همراه مسیر، مثل C:\Windows) نمایش داده می‌شود، ولی در کد مرحله‌ی ۸ با Path.GetFileName فقط نام زیرپوشه (مثل Windows) در فهرست قرار می‌گیرد.

۹ ـ (کد کتاب)

```
private void btnShowFiles_Click(object sender, EventArgs e)
{
    string directory = txtPath.Text;
    lstBoard.Items.Clear();
    string[] files = Directory.GetFiles(directory);
    foreach (string file in files)
        lstBoard.Items.Add(Path.GetFileName(file));
}
```

برای نمونه اجرای واقعی همین متدها روی پوشه‌ای با دو زیرپوشه و دو پرونده: زیرپوشه‌ها d1/sub1 و d1/sub2، پرونده‌ها (با GetFileName) a.txt و b.txt.

۲۴۲ صفحه ۲۲۱ **فعالیت گروهی**

به کمک هم‌گروهی خود یک پایگاه داده برای ذخیره‌ی مخاطبان ایجاد کنید و به سؤالات زیر پاسخ دهید: چگونه پایگاه داده را در Access ایجاد می‌کنید؟ چگونه در پایگاه داده جدول مخاطبان را ایجاد می‌کنید؟ چه فیلدهایی را برای جدول مخاطبان تعریف می‌کنید؟ نوع داده‌ی این فیلدها چیست؟ چگونه رکورد جدید در جدول مخاطبان وارد می‌کنید؟ چگونه یک رکورد را حذف می‌کنید؟

چند نرم‌افزار پیرامون خود را نام ببرید که از پایگاه داده برای ذخیره و بازیابی اطلاعات استفاده می‌کنند. نام بازی‌هایی را بنویسید که از پایگاه داده برای ذخیره‌ی اطلاعات بازی استفاده می‌کنند. چند نرم‌افزار فارسی تلفن هوشمند را نام ببرید که از پایگاه داده استفاده می‌کند.

آیا امکانات موجود در نرم‌افزارهای مدیریت پایگاه داده نظیر Access برای تولید یک نرم‌افزار که کار ذخیره و بازیابی اطلاعات را انجام دهد، کافی نیست؟ چرا؟

کنجکاوی: یک بازی برخط (online) را بررسی کنید که مشخصات، سطح و امتیاز کاربران در آن ذخیره می‌شود. فکر می‌کنید پایگاه داده‌ی این بازی از چه جدول‌هایی تشکیل شده است؟

1. اجرای Access ← Blank database ← تعیین نام (مثلاً Contacts) و پوشه ← Create.
2. ساخت جدول: Create ← Table Design؛ تعریف فیلدها و نوع داده؛ تعیین کلید اصلی؛ ذخیره با نام Contacts.
3. ورود رکورد: باز کردن جدول در Datasheet View و تایپ مقادیر در سطر خالی انتهای جدول.
4. حذف رکورد: انتخاب سطر ← راست‌کلیک ← Delete Record (یا کلید Delete) و تأیید.

فیلدهای پیشنهادی جدول مخاطبان

نام فیلد	نوع داده	توضیح
ContactID	AutoNumber	کلید اصلی
FirstName	Short Text	نام
LastName	Short Text	نام خانوادگی
Phone	Short Text	شماره‌ی تلفن (به‌دلیل صفر ابتدایی، متنی)
Address	Long Text	نشانی

نرم‌افزارهایی که از پایگاه داده استفاده می‌کنند: برنامه‌های فروشگاهی و حسابداری، نرم‌افزار بانکی و خودپرداز، سامانه‌ی کتابخانه، سامانه‌ی ثبت‌نام و نمره‌ی مدرسه، دفترچه‌ی مخاطبان گوشی. بازی‌ها: بازی‌های برخط (مانند Clash of Clans و PUBG که اطلاعات بازیکن، سطح و امتیاز را نگه می‌دارند)، بازی‌های موبایلی دارای جدول امتیازات. نرم‌افزارهای فارسی تلفن همراه: مثل دیوار، بله، ایتا، اسنپ، تپسی، بازار، دیجی‌کالا، دفترچه‌ی مخاطبان.

امکانات Access کافی نیست، زیرا Access فقط ساخت و مدیریت داده را به کاربر آشنا ارائه می‌کند؛ ولی کاربران عادی (فروشنده، کتابدار) نیاز به یک واسط گرافیکی ساده و اختصاصی، منطق برنامه (محاسبه‌ی مبلغ، کنترل ورود کاربر، چاپ و ...) و ارتباط با سایر اجزای برنامه دارند که با زبان برنامه‌نویسی ساخته می‌شود.

کنجکاوی: جدول‌های پیشنهادی بازی برخط: Players (کد بازیکن، نام کاربری، گذرواژه، تاریخ ثبت‌نام)، Levels (کد مرحله، عنوان، حداقل امتیاز لازم)، PlayerLevel (کد بازیکن، کد مرحله، امتیاز، تعداد ستاره)، و در صورت نیاز Items برای اشیای بازی.

۲۴۳ صفحه ۲۲۲ فعالیت کارگاهی

جدول بالا را در Access پیاده‌سازی کرده، رکوردهای بالا را در جدول وارد کنید. اگر دو کارمند فرهنگی نام و نام خانوادگی یکسانی داشته باشند، آیا برای صورت‌حساب آنها مشکلی پیش می‌آید؟ چرا؟ مشکل را چگونه حل می‌کنید؟

ـ برگه‌ی خرید شامل چه داده‌هایی می‌شود؟

ـ جدول «کالا» شامل چه فیلدهایی می‌شود؟ ـ جدول «مشتری» شامل چه فیلدهایی می‌شود؟

ـ جدول «برگه خرید» شامل چه فیلدهایی می‌شود؟ ـ جدول «ثبت برگه خرید» شامل چه فیلدهایی می‌شود؟

جدول اولیه (Access)

نام	نام خانوادگی	تاریخ	مبلغ خرید (تومان)
محمد	محمدی	1396/2/1	52000
علی	غلامی	1396/2/1	120000
فرهاد	سهرابی	1396/2/2	30000
محمد	محمدی	1396/2/3	45000

بله، مشکل پیش می‌آید؛ چون دو نفر با نام و نام خانوادگی یکسان از روی نام قابل تشخیص نیستند و خریدهای آنها با هم جمع می‌شود. راه‌حل: در جدول مشتری یک فیلد یکتا مانند CustomerCode (کد مشتری، یا کد ملی/کد پرسنلی) به‌عنوان کلید اصلی تعریف می‌کنیم و خریدها را با آن کد ثبت می‌کنیم.

برگه‌ی خرید شامل: شماره‌ی برگه، تاریخ، مشخصات مشتری (نام و نام خانوادگی)، فهرست اقلام خرید (نام کالا، تعداد، قیمت واحد، قیمت کل هر قلم) و مبلغ کل قابل پرداخت است.

ساختار جدول‌های پایگاه داده‌ی MyShop

جدول	فیلدها
Goods (کالا)	GoodsCode (کلید)، GoodsName، GoodsUnitPrice، GoodsStock
Customers (مشتری)	CustomerCode (کلید)، CustomerFirstName، CustomerLastName، CustomerPhone، CustomerAddress
Factors (برگه خرید)	FactorCode (کلید)، CustomerCode (کلید خارجی)، FactorDate
FactorItems (ثبت برگه خرید)	FactorCode و GoodsCode (کلید ترکیبی/خارجی)، Amount (تعداد)، UnitPrice

جدول «ثبت برگه خرید» لازم است زیرا هر مشتری در هر بار مراجعه ممکن است چند قلم کالا بخرد (رابطه‌ی یک به چند میان برگه و اقلام).

ایران توشه
توشه ای برای موفقیت

۲۴۴ صفحه ۲۲۳ فعالیت کارگاهی

در یک بازی می‌توان با ایجاد یک نام کاربری وارد بازی شد. هر کاربر با کسب حداقل امتیاز به سطح بالاتر می‌رود و با توجه به امتیاز خود ستاره کسب می‌کند. پایگاه داده‌ی مناسب را برای این بازی طراحی کنید تا نشان دهد که هر کاربر در هر سطح چند امتیاز و چند ستاره دارد.

پایگاه داده‌ی MyShop را با توجه به شکل ۶ در Access ایجاد کرده، داده‌ی مناسب را در جدول‌ها وارد کنید.

فعالیت منزل: فرض کنید در فروشگاه فرهنگیان بیش از یک فروشنده کار فروش کالا را انجام می‌دهند. اگر قرار باشد در برگ خرید، نام فروشنده نیز قید شود، چه تغییری باید در ساختار پایگاه داده‌ی فروشگاه صورت گیرد؟

طراحی پایگاه داده‌ی بازی

جدول	فیلدها
Users	UserID (کلید)، UserName، Password
Levels	LevelID (کلید)، LevelName، MinScore (حداقل امتیاز برای ورود)
UserLevels	UserID، LevelID (کلید ترکیبی)، Stars، Score

هر ردیف جدول UserLevels نشان می‌دهد هر کاربر در هر سطح چند امتیاز و چند ستاره دارد. Users و Levels از طریق کلید خارجی به آن وصل می‌شوند (رابطه‌ی چند به چند میان کاربر و سطح).

MyShop: چهار جدول Goods، Customers، Factors و FactorItems را با فیلدهای جدول صفحه‌ی ۲۲۲ می‌سازیم (کلید اصلی GoodsCode در Goods، CustomerCode در Customers، FactorCode در Factors) و در پنجره‌ی Relationships روابط را برقرار کرده، چند رکورد نمونه وارد می‌کنیم؛ مثلاً (۱، خودکار، ۵۰۰۰، ۱۰۰)، (۲، دفتر، ۱۲۰۰۰، ۵۰).

فعالیت منزل: یک جدول جدید Sellers (SellerCode به‌عنوان کلید، SellerName، ...) ایجاد می‌کنیم و یک فیلد SellerCode به‌عنوان کلید خارجی به جدول Factors (برگه‌ی خرید) اضافه می‌کنیم تا در هر برگه مشخص باشد کدام فروشنده آن را صادر کرده است.

ایران توشه
توشه ای برای موفقیت

۲۴۵ صفحه ۲۲۴ کارگاه ۷

۲ ـ جدول Computer را با فیلدهای زیر ایجاد کنید. فیلد WordName باید یکتا باشد. چگونه از تکراری بودن آن جلوگیری می‌کنید؟

۶ ـ کنترل DataGridView را به منبع اطلاعات متصل کنید. پس از مشاهده‌ی فیلم: چه کنترل‌هایی به فرم برنامه اضافه شده است؟ در بخش Solution Explorer چه تغییری رخ داده است؟ چه پرونده‌ای به این بخش اضافه شده است؟ کلید F7 را فشار دهید. آیا کدی به برنامه اضافه شده است؟

ساختار جدول Computer

نام فیلد	نوع داده	توضیح
WordCode	AutoNumber	کلید اصلی
WordName	Short Text	واژه (یکتا)
WordMeans	Long Text	معنی واژه

برای جلوگیری از تکراری بودن WordName در طراحی جدول، ویژگی Indexed فیلد را روی Yes (No Duplicates) می‌گذاریم (یا آن را کلید اصلی/یکتا قرار می‌دهیم).

۶ ـ کنترل‌های اضافه‌شده به فرم (در سینی کنترل‌های غیرمرئی): dictionaryDataSet، computerBindingSource و computerTableAdapter (و برای راه‌اندازی خودکار، tableAdapterManager و bindingNavigator در صورت انتخاب). در Solution Explorer پرونده‌ی dictionaryDataSet.xsd (و پایگاه داده‌ی dictionary.mdb) به پروژه افزوده می‌شود. با فشردن F7 کد فرم باز می‌شود: wizard در رویداد Load فرم خودکار این خط را می‌نویسد:

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

یعنی بله، کدی به برنامه اضافه شده است (کد تولیدی طراح در Form1.Designer.cs و خط Fill بالا در Form1.cs).

۲۴۶ صفحه ۲۲۶ کارگاه ۸

۲ ـ فیلدهایی را که می‌خواهید نمایش داده نشود، حذف کنید. کدام فیلد نباید در واژه‌نامه نمایش داده شود؟ ۳ ـ HeaderText را فارسی کنید. در DataGridView چه تغییری اتفاق می‌افتد؟ ۴ ـ ویژگی Dock را به Fill تغییر دهید. چه تغییری مشاهده می‌کنید؟ ۵ ـ ویژگی AutoSizeColumnsMode را به Fill تغییر دهید. چه تغییری مشاهده می‌کنید؟

۶ ـ مقدار DataSource را None کنید. چه تغییری مشاهده می‌کنید؟ مقدار قبلی را بگذارید. چه تغییری مشاهده می‌کنید؟

۷ ـ رنگ قلم و زمینه‌ی کنترل را تغییر دهید. برای قالب‌بندی DataGridView چگونه ویژگی DefaultCellStyle را تغییر می‌دهید؟ برنامه را اجرا کنید و روی سرستون «واژه» کلیک کنید. چه تغییری رخ می‌دهد؟ آیا می‌توانید از طریق این کنترل رکوردی را حذف یا اضافه کنید؟ چه ویژگی را باید تغییر دهید؟

جدول ۳ را تکمیل کنید.

فعالیت منزل: جدول مخاطبان تلفن همراه را روی فرم نمایش دهید. از پایگاه داده‌ی فروشگاه، جدول کالا را روی فرم نمایش دهید. پژوهش: در مورد ویژگی‌های دیگر کنترل DataGridView تحقیق کنید.

۲ ـ فیلد WordCode (کد واژه، که فقط کلید اصلی خودکار است) نباید نمایش داده شود؛ با انتخاب آن در Edit Columns و کلیک Remove حذف می‌شود.

ایران توشه
توشه‌ای برای موفقیت

۳ ـ سرستون‌ها به‌جای نام فیلد انگلیسی، عنوان فارسی (مثلاً «واژه» و «معنی») نشان می‌دهند. ۴ ـ با Dock = Fill جدول تمام فضای فرم/ظرف را پر می‌کند و با تغییر اندازه‌ی فرم کش می‌آید. ۵ ـ با AutoSizeColumnsMode = Fill پهنای ستون‌ها طوری تنظیم می‌شود که کل عرض کنترل را پر کنند و نوار پیمایش افقی لازم نباشد.

۶ ـ با DataSource = None جدول خالی می‌شود و هیچ داده‌ای (و ستونی) نمایش داده نمی‌شود، چون کنترل به منبع داده وصل نیست. با برگرداندن مقدار قبلی (computerBindingSource) دوباره داده‌ها نمایش داده می‌شوند.

۷ ـ در Properties، DefaultCellStyle را باز می‌کنیم (دکمه‌ی ...) و ForeColor، BackColor، Font و Alignment را تنظیم می‌کنیم. با کلیک روی سرستون، داده‌ها بر اساس آن ستون مرتب (صعودی/نزولی) می‌شوند. از طریق کنترل می‌توان رکورد حذف یا اضافه کرد اگر ویژگی‌های AllowUserToAddRows و AllowUserToDeleteRows برابر True باشند (و منبع داده قابل ویرایش باشد).

جدول ۳ـ ویژگی‌های DataGridView

نام ویژگی	شرح
Dock	مشخص می‌کند که کدام لبه از کنترل به فرم (ظرف) محدود شود.
AutoSizeColumnsMode	وضعیت اندازه‌ی خودکار برای ستون را مشخص می‌کند.
DefaultCellStyle	قالب (رنگ، قلم، تراز و ...) پیش‌فرض سلول‌های داده را مشخص می‌کند.
DataSource	منبع داده‌ی کنترل (مثلاً BindingSource) را مشخص می‌کند.
HeaderText	عنوان نمایش‌داده‌شده‌ی سرستون را مشخص می‌کند.
ColumnHeadersDefaultCellStyle	قالب پیش‌فرض سلول‌های سرستون را مشخص می‌کند.
EnableHeadersVisualStyles	اگر False باشد، قالب سفارشی سرستون (رنگ و ...) اعمال می‌شود و سبک ظاهری ویندوز نادیده گرفته می‌شود.

پژوهش: ویژگی‌های دیگر مانند ReadOnly (فقط‌خواندنی)، AllowUserToAddRows، AllowUserToDeleteRows، SelectionMode (انتخاب کل سطر)، MultiSelect، AlternatingRowsDefaultCellStyle (رنگ یک‌درمیان سطرها)، RowHeadersVisible، ColumnCount، Columns و Rows.

فعالیت منزل: همانند کارگاه ۷ برای جدول Contacts و جدول Goods با wizard یک DataGridView به منبع داده متصل می‌کنیم؛ مراحل: DataSet ← Database ← Add New Data Source ← انتخاب پرونده ← انتخاب جدول ← کشیدن جدول به فرم.

ایران توشه
توشه‌ای برای موفقیت

۲ ـ برای فراهم کردن امکان نمایش چند خط در کادرمتن txtWordMeans، کدام ویژگی کادرمتن را باید تغییر داد؟

۳ ـ کادرهای متن را به فیلد موردنظر مقید کنید. برنامه را اجرا کنید و روی یک سطر دلخواه در DataGridView کلیک کنید. چه مقداری در txtWordName قرار می‌گیرد؟

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

۴ ـ مقیدسازی ListBox: برای مقیدسازی کنترل کادرفهرست چگونه ویژگی DataSource و DisplayMember را تغییر می‌دهید؟

۵ ـ ComboBox را به فیلد WordName مقید کنید. ۶ ـ عملکرد کنترل BindingNavigator را بررسی کنید.

پس از طراحی پایگاه داده‌ی فروشگاه و ورود داده‌های مناسب در جدول‌ها، فرم برنامه‌ی فروشگاه را به‌صورت شکل ۸ طراحی کنید.

---

۲ ـ ویژگی MultiLine را True می‌کنیم (برای نوار پیمایش هم ScrollBars = Vertical).

۳ ـ در DataBindings ← Advanced ← Text، منبع computerBindingSource و فیلد WordName را انتخاب می‌کنیم. با کلیک روی هر سطر DataGridView، مقدار فیلد WordName همان رکورد (یعنی واژه‌ی همان سطر) در txtWordName قرار می‌گیرد؛ txtWordMeans هم به همین شکل به WordMeans مقید می‌شود.

کنجکاوی: کنترل‌هایی مانند TextBox، Label، ListBox، ComboBox، CheckBox، DateTimePicker و PictureBox قابل مقید شدن هستند. برای نامقید کردن با کد، مقیدشدن را پاک می‌کنیم:

```
txtWordName.DataBindings.Clear();   // removes all bindings of the control
```

۴ و ۵ ـ مقیدسازی کادرفهرست و کادر ترکیبی با کد (یا از Properties):

```
listBox1.DataSource = computerBindingSource;
listBox1.DisplayMember = "WordName";
comboBox1.DataSource = computerBindingSource;
comboBox1.DisplayMember = "WordName";
// binding a textbox by code:
txtWordName.DataBindings.Add("Text", computerBindingSource, "WordName");
```

۶ ـ BindingNavigator یک نوار ابزار برای پیمایش و مدیریت رکوردهاست (اولین، قبلی، شماره‌ی رکورد جاری، بعدی، آخرین، افزودن، حذف و ذخیره)، که با تنظیم BindingSource = computerBindingSource بدون کدنویسی کار می‌کند.

ایران توشه
توشه ای برای موفقیت

فرم فروشگاه (شکل ۸): یک TabControl با نام tabShop و سربرگ‌های «کالا» (tabPageGoods)، «مشتری»، «برگه خرید» (tabPageFactor) و ... ساخته می‌شود؛ در سربرگ کالا یک DataGridView، چهار کادرمتن برای مقادیر فیلدها، دکمه‌های پیمایش و دکمه‌های «ثبت»، «حذف» و «ویرایش» قرار می‌گیرد.

این جواب ممکن است دقیق نباشد. (چیدمان دقیق شکل ۸ در متن موجود نیست؛ طرح بالا بر اساس سربرگ‌های ذکرشده در کتاب است.)

---

۲۴۸ صفحه ۲۲۹ کارگاه ۱۰

کنجکاوی: myShopDataSet شامل چه جدول‌های دیگری است؟

۳ ـ متد Load فرم را بنویسید. برنامه را اجرا کنید. آیا می‌توان برای مقیدسازی کنترل dgvGoods دستور دیگری نوشت؟

۴ ـ برای رفتن به اولین رکورد کد زیر را بنویسید. برای رفتن به آخرین رکورد چه دستوری می‌نویسید؟

۵ ـ برای رفتن به رکورد قبلی کد بنویسید. برای رفتن به رکورد بعدی چه کدی می‌نویسید؟ (جدول نام و عملکرد دکمه‌ها)

کنجکاوی: myShopDataSet علاوه بر Goods، شامل جدول‌های Customers، Factors، FactorItems و Users است.

نام کنترل‌های پیمایش

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

(در متن کتاب ترتیب نام‌ها و عملکردها در جدول به‌هم‌ریخته است؛ جدول بالا صورت درست است.)

```
private void MainForm_Load(object sender, EventArgs e)
{
    this.goodsTableAdapter.Fill(this.myShopDataSet.Goods);
    dgvGoods.DataSource = goodsBindingSource;
}
private void btnFirst_Click(object sender, EventArgs e)
{ goodsBindingSource.MoveFirst(); }
private void btnPrev_Click(object sender, EventArgs e)
{ goodsBindingSource.MovePrevious(); }
private void btnNext_Click(object sender, EventArgs e)
{ goodsBindingSource.MoveNext(); }
private void btnLast_Click(object sender, EventArgs e)
{ goodsBindingSource.MoveLast(); }
```

ایران توشه
توشه‌ای برای موفقیت

دستور Fill جدول Goods را از پایگاه داده می‌خواند و داده‌ها را در DataSet می‌ریزد. برای مقیدسازی dgvGoods دستور دیگر: dgvGoods.DataSource = myShopDataSet.Goods; یا dgvGoods.DataSource = myShopDataSet; dgvGoods.DataMember = "Goods"; برای رفتن به آخرین رکورد ()MoveLast و برای رکورد بعدی ()MoveNext نوشته می‌شود.

۲۴۹ صفحه ۲۳۰ کارگاه ۱۱

کنجکاوی: مقدار اندیس (شماره‌ی) اولین رکورد چیست؟

۳ ـ متد رویداد CellClick کنترل dgvGoods را بنویسید و برنامه را اجرا کنید و با کلیک روی یک سطر دلخواه نتیجه‌ی کار را ببینید.

فعالیت منزل: در سربرگ مشتری، فرمی همانند فرم کالا ایجاد کنید و از طریق کدنویسی بارگذاری داده در DataGridView را انجام دهید و دکمه‌های پیمایش رکوردها را ایجاد کنید. (فعالیت کارگاهی: برای سربرگ مشتری برنامه‌ی فروشگاه، عملیات بالا را انجام دهید.)

اندیس اولین رکورد (سطر) صفر است.

```
private void dgvGoods_CellClick(object sender, DataGridViewCellEventArgs e)
{
    int index = dgvGoods.CurrentCell.RowIndex;
    txtGoodsCode.Text = dgvGoods.Rows[index].Cells[0].Value.ToString();
    txtGoodsName.Text = dgvGoods.Rows[index].Cells[1].Value.ToString();
    txtUnitPrice.Text = dgvGoods.Rows[index].Cells[2].Value.ToString();
    txtStock.Text     = dgvGoods.Rows[index].Cells[3].Value.ToString();
}
```

با کلیک روی هر سطر، مقدار چهار خانه‌ی همان سطر (کد، نام، قیمت واحد و موجودی) در چهار کادرمتن نشان داده می‌شود.

سربرگ مشتری: همان کارها را با customersTableAdapter، customersBindingSource، dgvCustomers و کادرمتن‌های مشتری انجام می‌دهیم:

```
this.customersTableAdapter.Fill(this.myShopDataSet.Customers);
dgvCustomers.DataSource = customersBindingSource;

private void btnCFirst_Click(object sender, EventArgs e)
{ customersBindingSource.MoveFirst(); }
private void btnCLast_Click(object sender, EventArgs e)
{ customersBindingSource.MoveLast(); }
private void dgvCustomers_CellClick(object sender, DataGridViewCellEventArgs e)
{
    int i = dgvCustomers.CurrentCell.RowIndex;
    txtCustomerCode.Text = dgvCustomers.Rows[i].Cells[0].Value.ToString();
    txtCustomerName.Text = dgvCustomers.Rows[i].Cells[1].Value.ToString();
}
```

ایران توشه
توشه‌ای برای موفقیت

۳ ـ پنل سمت راست را به‌صورت افقی به دو بخش تقسیم کنید. ویژگی Orientation آن را Horizontal قرار دهید. در این حالت چه تغییری مشاهده می‌کنید؟

۴ ـ کنترل کادرفهرست را به پنل پایین در سمت راست فرم اضافه کنید و Dock آن را Fill قرار دهید. در این حالت چه تغییری مشاهده می‌کنید؟

کنجکاوی: روی چه کنترل‌هایی می‌توان کنترل SplitContainer را قرار داد؟

۸ ـ برای جستجوی واژه، متد رویداد TextChanged کادرمتن جستجو (txtSearchWord) را بنویسید. عملکرد دستور if چیست؟ عملکرد دستور SELECT چیست؟ علامت «سؤال» در پرسوجو چه چیزی است؟

۹ ـ برای استفاده از متد GetDataByWordName دستور if را ویرایش کنید. این کد را با کد مرحله‌ی ۸ مقایسه کنید و تفاوت دو متد GetDataByWordName و FillByWordName را بنویسید. نوع ویژگی DataSource کنترل computerBindingSource چیست؟

۱۰ ـ پرسوجو را برای جستجوی کلمات مشابه با LIKE ویرایش کنید. عملکرد دستور Like چیست؟

فعالیت منزل: قابلیت جستجو بر اساس فیلد نام و کدکالا را برای سربرگ کالا ایجاد کنید. قابلیت جستجو بر اساس نام و شماره‌ی همراه را برای جدول مخاطبان ایجاد کنید.

---

۳ ـ پنل سمت راست به دو بخش بالایی و پایینی (به‌جای چپ و راست) تقسیم می‌شود و بین آنها یک نوار افقی قرار می‌گیرد. ۴ ـ کادرفهرست تمام فضای پنل پایین را پر می‌کند.

کنجکاوی: SplitContainer را می‌توان روی Form، Panel، GroupBox، TabPage و حتی یک SplitContainer دیگر (پنل‌های آن) قرار داد.

۷ ـ پرسوجوی ساخته‌شده در wizard:

```
SELECT WordCode, WordName, WordMeans
FROM Computer
WHERE (WordName = ?)
```

SELECT برای انتخاب (بازیابی) داده‌ها از جدول است و ستون‌های ذکرشده را از رکوردهای جدول Computer که شرط WHERE را دارند برمی‌گرداند. علامت ? جای‌نگهدار ورودی (پارامتر) متد است که هنگام فراخوانی، مقدار آن (متن کادر جستجو) جایگزین می‌شود.

```
private void txtSearchWord_TextChanged(object sender, EventArgs e)
{
    if (txtSearchWord.TextLength > 0)
        computerTableAdapter.FillByWordName(dictionaryDataSet.Computer,
                                            txtSearchWord.Text);
    else
        computerTableAdapter.Fill(dictionaryDataSet.Computer);
}
```

ایران توشه
توشه‌ای برای موفقیت

دستور if: اگر کادر جستجو خالی نباشد، فقط رکورد(های) مطابق متن در جدول جستجو و نمایش داده می‌شوند، وگرنه همه‌ی رکوردها دوباره بارگذاری می‌شوند. متد FillByWordName مستقیماً DataTable را پر می‌کند و خروجی ندارد؛ اما GetDataByWordName یک DataTable جدید برمی‌گرداند که باید به DataSource داده شود:

```
if (txtSearchWord.TextLength > 0)
    computerBindingSource.DataSource =
        computerTableAdapter.GetDataByWordName(txtSearchWord.Text);
else
    computerTableAdapter.Fill(dictionaryDataSet.Computer);
```

نوع DataSource در BindingSource از نوع object است (می‌پذیرد DataTable ،DataSet، فهرست و ...).

برای جستجوی کلمات مشابه:

```
SELECT WordCode, WordName, WordMeans
FROM Computer
WHERE (WordName Like ? + '%')
```

Like برای مقایسه‌ی الگو است؛ با الگوی '%' + ? همه‌ی واژه‌هایی را که با متن تایپ‌شده شروع می‌شوند برمی‌گرداند (مثلاً با تایپ «Co» همه‌ی واژه‌های شروع‌شونده با Co). این رفتار با sqlite در پایتون نیز آزمایش شد: فیلتر '%' || 'N' Like فقط رکورد «Note» را برگرداند.

فعالیت منزل: برای جدول کالا یک متد FillByGoodsNameOrCode با پرسوجوی زیر می‌سازیم و در رویداد TextChanged کادر جستجو فراخوانی می‌کنیم:

```
SELECT GoodsCode, GoodsName, GoodsUnitPrice, GoodsStock
FROM Goods
WHERE (GoodsName Like ? + '%') OR (GoodsCode = ?)
```

برای جدول مخاطبان: WHERE (FirstName Like ? + '%' OR LastName Like ? + '%') AND (Phone Like ? + '%').

این جواب ممکن است دقیق نباشد. (در Access برای Like معمولاً از علامت * یا % بسته به ADO استفاده می‌شود؛ در پرسوجوهای TableAdapter با OleDb علامت % کار می‌کند.)

۲۵۱ صفحه ۲۳۵ کارگاه ۱۳

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

۵ ـ در LoginForm فیلدی برای اعتبارسنجی کاربر به نام succeeded تعریف کنید. ۶ ـ متد رویداد کلیک دکمه‌ی btnLogin را بنویسید. مفهوم شرط دستور if چیست؟ عملکرد متد btnLogin_Click چیست؟

۷ ـ متد رویداد Load فرم اصلی را بنویسید.

فعالیت منزل: برنامه‌ی ورود به فروشگاه را طوری تغییر دهید که با زدن تیک کادر علامت، نام کاربری برای ورود بعدی ذخیره شود و تصویر کاربر نیز نمایش داده شود.

پژوهش: با بازکردن جدول کاربران در پایگاه داده، هر فردی می‌تواند نام کاربری و گذرواژه‌ی همه‌ی کاربران را ببیند و به برنامه‌ی فروشگاه وارد شود. برای امنیت بیشتر برنامه چه کار باید کرد؟

۴ ـ پرسوجوی FillByUserNameAndPass:

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

```
public partial class LoginForm : Form
{
    public bool succeeded;     // true if the user is valid
    ...
    private void btnLogin_Click(object sender, EventArgs e)
    {
        usersTableAdapter.FillByUserNameAndPass(myShopDataSet.Users,
            txtUserName.Text, txtPassword.Text);
        if (myShopDataSet.Users.Rows.Count > 0)
        {
            succeeded = true;
            this.Close();
        }
        else
            MessageBox.Show("Invalid user name or password", "Error",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
```

```
private void MainForm_Load(object sender, EventArgs e)
{
    LoginForm frmLogin = new LoginForm();
    frmLogin.ShowDialog();
    if (!frmLogin.succeeded)
        this.Close();
}
```

شرط if بررسی می‌کند آیا پرسوجو حداقل یک رکورد (کاربری با این نام کاربری و گذرواژه) پیدا کرده است (Rows.Count > 0)؛ اگر بله، کاربر معتبر است. btnLogin_Click با کلیک روی دکمه، کاربر را با نام کاربری و گذرواژه جستجو می‌کند؛ اگر یافت شد فرم ورود بسته می‌شود و succeeded برابر true می‌شود، وگرنه پیام خطا نشان می‌دهد. در MainForm_Load فرم ورود به‌صورت ShowDialog (مودال) نمایش داده می‌شود و اگر ورود موفق نبود، فرم اصلی بسته می‌شود.

فعالیت منزل: یک CheckBox به نام chkRemember اضافه می‌کنیم. با تیک‌خوردن، نام کاربری در یک پرونده‌ی متنی (یا Properties.Settings) ذخیره می‌شود و در Load فرم ورود خوانده می‌شود؛ تصویر کاربر هم با فیلد مسیر تصویر در جدول Users و یک PictureBox نمایش داده می‌شود:

ایران توشه
توشه‌ای برای موفقیت

```
// on successful login
if (chkRemember.Checked)
    File.WriteAllText("user.txt", txtUserName.Text);
// in LoginForm_Load
if (File.Exists("user.txt"))
{
    txtUserName.Text = File.ReadAllText("user.txt");
    chkRemember.Checked = true;
}
// picture (UserPicture holds the image path)
if (File.Exists(path)) picUser.Image = Image.FromFile(path);
```

پژوهش: گذرواژه‌ها را به‌صورت متن ساده ذخیره نکنیم؛ آنها را با الگوریتم درهم‌ساز (Hash) مانند SHA-256 همراه با Salt ذخیره و هنگام ورود، هش گذرواژه‌ی واردشده را با هش ذخیره‌شده مقایسه کنیم؛ برای پایگاه داده رمز عبور بگذاریم، دسترسی کاربران به پرونده‌ی پایگاه داده را محدود کنیم، از پرسوجوی پارامتری (نه چسباندن رشته) استفاده کنیم و کادر گذرواژه را با UseSystemPasswordChar = True بپوشانیم.

این جواب ممکن است دقیق نباشد. (فیلم «اتصال به پایگاه داده» در دسترس نبود؛ مراحل آن بر اساس همان کارگاه ۷ نوشته شد.)

۲۵۲ صفحه ۲۳۷ کارگاه ۱۴

۲ ـ متد «اضافه‌کردن کالا» را ایجاد کنید. برای ایجاد این متد روی فایل myShopDataSet.xsd دابل‌کلیک کنید. چه جدول‌هایی مشاهده می‌کنید؟

کنجکاوی: ساختار دستور INSERT INTO در زبان SQL چگونه است؟

۳ ـ متد رویداد کلیک دکمه‌ی «ثبت» را بنویسید. متد InsertQuery دارای چند آرگومان ورودی است؟ چرا برخی از ورودی‌ها عددی است؟ چرا از دستور if در برنامه استفاده شده است؟ عملکرد متد btnAddGoods_Click و دستور goodsTableAdapter.InsertQuery چیست؟ برنامه را اجرا کنید و یک رکورد جدید را وارد کنید و «ثبت» را کلیک کنید. آیا رکورد جدید در dgvGoods نمایش داده می‌شود؟ چرا؟ برنامه را ببندید و دوباره اجرا کنید. آیا رکورد اضافه‌شده در dgvGoods وجود دارد؟ جدول Goods پایگاه داده‌ی MyShop را باز کنید. آیا رکورد اضافه‌شده در جدول وجود دارد؟

۴ ـ ویژگی Copy to Output Directory پایگاه داده را تنظیم کنید. دوباره برنامه را اجرا کنید و یک رکورد جدید را وارد کرده، «ثبت» را کلیک کنید و برنامه را ببندید. دوباره اجرا کنید. آیا در جدول Goods این کالا اضافه شده است؟

فعالیت منزل: برای جدول مشتری برنامه‌ی درج رکورد را بنویسید. برای جدول مخاطبان برنامه‌ی ثبت مخاطب جدید را بنویسید.

---

۲ ـ جدول‌های Goods، Customers، Factors، FactorItems و Users (با GoodsTableAdapter و ... پایین هر جدول) در طراح مشاهده می‌شود.

ساختار کلی: INSERT INTO نام_جدول (فیلد۱, فیلد۲, ...) VALUES (مقدار۱, مقدار۲, ...). پرسوجوی ساخته‌شده:

```
INSERT INTO Goods (GoodsCode, GoodsName, GoodsUnitPrice, GoodsStock)
VALUES (?, ?, ?, ?)
```

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

متد InsertQuery چهار آرگومان دارد (به تعداد علامت‌های «؟»)؛ کد، قیمت و موجودی در جدول عددی‌اند، بنابراین با int.Parse از متن به عدد تبدیل می‌شوند. if بررسی می‌کند فیلدهای ضروری خالی نباشند تا خطای تبدیل/درج رخ ندهد. btnAddGoods_Click رکورد جدید را در جدول Goods درج کرده و جدول نمایش‌داده‌شده را دوباره بارگذاری (Fill) می‌کند؛ goodsTableAdapter.InsertQuery دستور INSERT را روی پایگاه داده اجرا می‌کند.

اجرا: بله، رکورد جدید در dgvGoods نمایش داده می‌شود (چون بعد از درج، Fill اجرا شده). اما با بستن و اجرای دوباره، رکورد وجود ندارد و در پرونده‌ی پایگاه داده‌ی اصلی هم نیست؛ زیرا هر بار اجرا، Visual Studio نسخه‌ی پایگاه داده‌ی پروژه را روی نسخه‌ی پوشه‌ی bin\Debug کپی (بازنویسی) می‌کند و تغییرات در همان نسخه‌ی خروجی بوده است.

۴ ـ با تغییر Copy to Output Directory از Copy always به Copy if newer، پایگاه داده فقط وقتی کپی می‌شود که نسخه‌ی پروژه جدیدتر باشد؛ پس تغییرات در نسخه‌ی خروجی حفظ می‌شود و در اجرای بعدی رکورد اضافه‌شده موجود است.

فعالیت منزل: برای جدول مشتری یک پرسوجوی InsertQuery مشابه بسازید و کد زیر را بنویسید (نام کنترل‌ها نمونه است):

ایران توشه
توشه ای برای موفقیت

```
private void btnAddCustomer_Click(object sender, EventArgs e)
{
    if (txtCustomerCode.TextLength > 0 && txtCustomerName.TextLength > 0)
    {
        customersTableAdapter.InsertQuery(int.Parse(txtCustomerCode.Text),
            txtCustomerName.Text, txtPhone.Text);
        customersTableAdapter.Fill(myShopDataSet.Customers);
        MessageBox.Show("Saved successfully");
    }
}
// Contacts table: same idea
contactsTableAdapter.InsertQuery(txtFirstName.Text, txtLastName.Text,
                                 txtPhone.Text, txtAddress.Text);
contactsTableAdapter.Fill(dictionaryDataSet.Contacts);
```

۲۵۳ صفحه ۲۳۹ کارگاه ۱۵

۳ ـ متد رویداد کلیک دکمه‌ی «حذف» را بنویسید. برنامه را اجرا کرده، یک رکورد دلخواه را حذف کنید.

۴ ـ دستور را تغییر دهید تا حذف بر اساس نام کالا باشد. ۵ ـ دستور را برای حذف تمامی رکوردها تغییر دهید.

فعالیت منزل: برای جدول مشتری برنامه‌ی حذف رکورد را بنویسید. برای جدول مخاطبان برنامه‌ی حذف مخاطب را بنویسید.

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

```
private void btnDeleteGoods_Click(object sender, EventArgs e)
{
    if (txtGoodsCode.TextLength > 0)
    {
        goodsTableAdapter.DeleteQuery(int.Parse(txtGoodsCode.Text));
        goodsTableAdapter.Fill(myShopDataSet.Goods);
        MessageBox.Show("Deleted successfully");
    }
}
```

ایران توشه
توشه‌ای برای موفقیت

۴ ـ حذف بر اساس نام: شرط پرسوجو را به WHERE (GoodsName = ?) تغییر می‌دهیم (متد DeleteQuery یک ورودی رشته‌ای می‌گیرد) و فراخوانی می‌شود:

goodsTableAdapter.DeleteQuery(txtGoodsName.Text);.

۵ ـ حذف همه‌ی رکوردها: دستور بدون شرط نوشته می‌شود و متد بدون ورودی خواهد بود:

```
DELETE FROM Goods
```

(این دستور خطرناک است؛ بهتر است پیش از اجرا از کاربر تأیید گرفته شود.) آزمایش با sqlite: پس از DELETE FROM Goods WHERE GoodsCode = 1 از سه رکورد، دو رکورد باقی ماند.

```
private void btnDeleteCustomer_Click(object sender, EventArgs e)
{
    if (txtCustomerCode.TextLength > 0)
    {
        customersTableAdapter.DeleteQuery(int.Parse(txtCustomerCode.Text));
        customersTableAdapter.Fill(myShopDataSet.Customers);
    }
}
// Contacts
contactsTableAdapter.DeleteQuery(int.Parse(txtContactID.Text));
contactsTableAdapter.Fill(dictionaryDataSet.Contacts);
```

۲۵۴ صفحه ۲۴۰ کارگاه ۱۶

با استفاده از دکمه‌ی Query Builder دستور SQL ایجادشده را طوری تغییر دهید که ویرایش رکورد بر اساس کد کالا انجام شود. ۳ ـ متد رویداد کلیک دکمه‌ی «ویرایش» را بنویسید. برنامه را اجرا کنید و با انتخاب یک رکورد عمل ویرایش را انجام دهید.

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

```
UPDATE Goods
SET GoodsName = ?, GoodsUnitPrice = ?, GoodsStock = ?
WHERE (GoodsCode = ?)
```

```
private void btnUpdateGoods_Click(object sender, EventArgs e)
{
    if (txtGoodsCode.TextLength > 0 && txtGoodsName.TextLength > 0 &&
        txtUnitPrice.TextLength > 0)
    {
        goodsTableAdapter.UpdateQuery(txtGoodsName.Text,
            int.Parse(txtUnitPrice.Text), int.Parse(txtStock.Text),
            int.Parse(txtGoodsCode.Text));
        goodsTableAdapter.Fill(myShopDataSet.Goods);
        MessageBox.Show("Updated successfully");
    }
}
```

نکته: ترتیب آرگومان‌های UpdateQuery باید با ترتیب علامت‌های «؟» در پرسوجو یکی باشد (نام، قیمت، موجودی و در پایان کد کالا در شرط WHERE). آزمایش sqlite: دستور UPDATE با شرط کد ۱ رکورد را به (Pencil، ۴۰۰۰، ۲۰) تغییر داد. توجه: ترتیب ورودی‌ها در متن چاپی کتاب (کد ابتدا) به‌هم‌ریخته است؛ در پرسوجوی بالا کد در انتها آمده است.

فعالیت منزل ۱ ـ اگر کالا موجود بود فقط موجودی افزایش یابد:

```
int code = int.Parse(txtGoodsCode.Text);
goodsTableAdapter.FillByGoodsCode(myShopDataSet.Goods, code);
if (myShopDataSet.Goods.Rows.Count > 0)
{
    int stock = int.Parse(myShopDataSet.Goods.Rows[0]["GoodsStock"].ToString());
    goodsTableAdapter.UpdateStockQuery(stock + int.Parse(txtStock.Text), code);
}
else
    goodsTableAdapter.InsertQuery(code, txtGoodsName.Text,
        int.Parse(txtUnitPrice.Text), int.Parse(txtStock.Text));
goodsTableAdapter.Fill(myShopDataSet.Goods);
```

که UpdateStockQuery برابر UPDATE Goods SET GoodsStock = ? WHERE GoodsCode = ? است. برای مشتری و مخاطبان: همین الگو با پرسوجوی UPDATE Customers SET ... WHERE CustomerCode = ? و UPDATE Contacts SET ... WHERE ContactID = ? و فراخوانی UpdateQuery سپس Fill.

این جواب ممکن است دقیق نباشد. (نام کنترل‌ها و فیلدهای جدول‌های مشتری و مخاطبان در کتاب مشخص نشده و پیشنهادی است.)

۲۵۵ صفحه ۲۴۱ کارگاه ۱۷

۲ ـ ... کاربرد ویژگی‌های ImageLayout و FillWeight چیست؟

۴ ـ متد FillByGoodsCode را برای پرسوجو بر اساس کد کالا ایجاد کنید. ۵ ـ متد رویداد KeyDown_txtFGoodsCode را بنویسید.

۶ ـ متد رویداد کلیک دکمه‌ی انتقال به سربرگ کالا را بنویسید. ۷ ـ متد رویداد کلیک دکمه‌ی «درج در برگه خرید» سربرگ کالا را بنویسید.

۸ ـ متد رویداد کلیک دکمه‌ی جستجوی مشتری (btnCustomerTab) در سربرگ «برگه خرید» را بنویسید. ۹ ـ متد رویداد کلیک دکمه‌ی «درج در برگه خرید» در سربرگ مشتری را بنویسید.

۱۰ ـ متد رویداد کلیک دکمه‌ی «درج در سبد خرید» را بنویسید. چگونه می‌توان تعداد سطرهای یک DataGridView را تعیین کرد؟ عملکرد متد Add کنترل dgvFactors چیست؟ قیمت کل در کدام خانه‌ی dgvFactors قرار می‌گیرد و چگونه محاسبه می‌شود؟

۱۱ ـ کد مرحله‌ی ۱۰ را تغییر دهید تا در صورت خالی بودن کادرمتن کد کالا، خطا رخ ندهد. ۱۲ ـ محاسبه‌ی مبلغ قابل پرداخت را اضافه کنید و در lblAllCost قرار دهید.

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

---

۲ ـ ImageLayout نحوه‌ی نمایش تصویر در سلول را مشخص می‌کند (Stretch، Center، Tile، None، Zoom)؛ FillWeight وزن (سهم) پهنای ستون نسبت به ستون‌های دیگر را در حالت AutoSizeColumnsMode = Fill تعیین می‌کند.

```
SELECT GoodsCode, GoodsName, GoodsUnitPrice, GoodsStock
FROM Goods
WHERE (GoodsCode = ?)       -- FillByGoodsCode
```

```
private void txtFGoodsCode_KeyDown(object sender, KeyEventArgs e)
{
    lblPrice.Text = " ";
    lblGoodsName.Text = " ";
    if (e.KeyCode == Keys.Enter)
        if (txtFGoodsCode.Text.Length > 0)
        {
            goodsTableAdapter.FillByGoodsCode(myShopDataSet.Goods,
                int.Parse(txtFGoodsCode.Text));
            if (myShopDataSet.Goods.Rows.Count > 0)
            {
                lblGoodsName.Text = myShopDataSet.Goods.Rows[0]["GoodsName"].ToString();
                lblPrice.Text     = myShopDataSet.Goods.Rows[0]
["GoodsUnitPrice"].ToString();
                lblStock.Text     = myShopDataSet.Goods.Rows[0]
["GoodsStock"].ToString();
            }
        }
}
```

```
private void btnGoodsTab_Click(object sender, EventArgs e)
{
    tabShop.SelectedTab = tabPageGoods;   // activate Goods tab
    txtSGoodsName.Focus();
}
private void btnInsertGoodsFactor_Click(object sender, EventArgs e)
{
    tabShop.SelectedTab = tabPageFactor;  // activate Factor tab
    txtFGoodsCode.Text = txtGoodsCode.Text;
    lblGoodsName.Text = txtGoodsName.Text;
    lblPrice.Text = txtUnitPrice.Text;
    txtFGoodsCode.Focus();
}
private void btnCustomerTab_Click(object sender, EventArgs e)
{
    tabShop.SelectedTab = tabPageCustomers;
}
private void btnInsertCustomerFactor_Click(object sender, EventArgs e)
{
    tabShop.SelectedTab = tabPageFactor;
    lblCustomerCode.Text = txtCustomerCode.Text;
    lblCustomerName.Text = txtCustomerName.Text;
}
```

ایران توشه
توشه ای برای موفقیت

```
private void btnInsertGoods_Click(object sender, EventArgs e)
{
    if (txtFGoodsCode.TextLength == 0 || txtFAmount.TextLength == 0)
        return;                          // step 11: avoid the error
    int n = dgvFactors.Rows.Count - 1;   // index of the new row
    dgvFactors.Rows.Add();
    dgvFactors.Rows[n].Cells[0].Value = txtFGoodsCode.Text;
    dgvFactors.Rows[n].Cells[1].Value = lblGoodsName.Text;
    dgvFactors.Rows[n].Cells[2].Value = txtFAmount.Text;
    dgvFactors.Rows[n].Cells[3].Value = lblPrice.Text;
    dgvFactors.Rows[n].Cells[4].Value =
        int.Parse(txtFAmount.Text) * int.Parse(lblPrice.Text);
    // step 12: payable amount
    int sum = 0;
    for (int i = 0; i < dgvFactors.Rows.Count; i++)
        if (dgvFactors.Rows[i].Cells[4].Value != null)
            sum += int.Parse(dgvFactors.Rows[i].Cells[4].Value.ToString());
    lblAllCost.Text = sum.ToString();
}
```

تعداد سطرها با dgvFactors.Rows.Count به‌دست می‌آید (در حالتی که سطر خالی جدید فعال است، یکی بیشتر از رکوردهای واقعی است؛ برای همین 1 – Count نوشته شده). متد Add یک سطر خالی جدید به انتهای جدول اضافه می‌کند. قیمت کل در خانه‌ی شماره‌ی ۴ (ستون پنجم) هر سطر قرار می‌گیرد و از حاصل‌ضرب تعداد (ستون ۲) در قیمت واحد (ستون ۳) محاسبه می‌شود.

```
private void dgvFactors_CellContentClick(object sender,
                                         DataGridViewCellEventArgs e)
{
    if (dgvFactors.Columns[e.ColumnIndex] is DataGridViewImageColumn
        && e.RowIndex >= 0)
        dgvFactors.Rows.RemoveAt(e.RowIndex);
}
```

اپراتور is بررسی می‌کند ستونِ کلیک‌شده از نوع ستون تصویر (علامت حذف) باشد؛ در این حالت سطر انتخاب‌شده با RemoveAt حذف می‌شود. توجه: برای استفاده از Rows.Add باید DataGridView مقید نباشد.

این جواب ممکن است دقیق نباشد. (مراحل ۸ و ۹ در کتاب فقط توضیح متنی داشتند؛ کد آنها بر اساس همان توضیحات و نام‌های پیشنهادی نوشته شد.)

---

۲۵۶ صفحه ۲۴۵ کارگاه ۱۸

۲ ـ در سربرگ کالا، کالایی را ثبت کنید که کد کالای آن قبلاً ثبت شده باشد. با درج کد تکراری چه اتفاقی رخ می‌دهد؟

برای اینکه بدانیم هنگام عملیات مبنایی چه خطاهایی ممکن است رخ دهد، جدول (نام عمل ـ فهرست خطاها) را کامل کنید.

۳ ـ روی دکمه‌ی ثبت در سربرگ «کالا» دابل‌کلیک کنید و کدها را با try-catch ویرایش کنید. برنامه را اجرا کنید و کالایی را ثبت کنید که کد کالای آن تکراری باشد، چه اتفاقی رخ می‌دهد؟

۴ ـ مدیریت خطا را برای عملیات دیگری در سربرگ کالا انجام دهید. آیا دستور if در برنامه عمل کنترل خطا را انجام می‌دهد؟ اگر کد کالا غیرعددی نوشته شود، آیا خطایی رخ می‌دهد؟

۲ ـ چون GoodsCode کلید اصلی است، پایگاه داده درج رکورد تکراری را نمی‌پذیرد و خطای زمان اجرا (OleDbException: تغییرات به‌دلیل ایجاد مقدار تکراری در کلید اصلی/اندیس انجام نشد) رخ می‌دهد و برنامه بدون مدیریت خطا متوقف می‌شود. (در آزمایش sqlite نیز خطای UNIQUE constraint failed: Goods.GoodsCode مشاهده شد.)

خطاهای محتمل در عملیات مبنایی

نام عمل	فهرست خطاها
درج	• خطای ناشی از ساختار نادرست دستور درج • درج مقدار تکراری برای کلید اصلی • خالی‌ماندن فیلدی که نباید Null باشد • ناسازگاری نوع داده (مثلاً متن به‌جای عدد)
حذف	• خطای ناشی از ساختار نادرست دستور حذف • حذف رکوردی که در جدول دیگر به آن ارجاع داده شده (نقض یکپارچگی ارجاعی) • نبود اتصال به پایگاه داده
ویرایش	• خطای ناشی از ساختار نادرست دستور ویرایش • خالی‌ماندن فیلدی که نباید Null باشد • تغییر کلید اصلی به مقدار تکراری

```
private void btnAddGoods_Click(object sender, EventArgs e)
{
    if (txtGoodsCode.TextLength > 0 && txtGoodsName.TextLength > 0 &&
        txtUnitPrice.TextLength > 0)
    {
        try
        {
            goodsTableAdapter.InsertQuery(int.Parse(txtGoodsCode.Text),
                txtGoodsName.Text, int.Parse(txtUnitPrice.Text),
                int.Parse(txtStock.Text));
            goodsTableAdapter.Fill(myShopDataSet.Goods);
            MessageBox.Show("Saved successfully");
        }
        catch
        {
            MessageBox.Show("Error in saving the goods");
        }
    }
}
```

اکنون با ثبت کد تکراری برنامه متوقف نمی‌شود، بلکه پیام «خطا در ثبت کالا» نمایش داده شده و برنامه ادامه می‌یابد؛ رکورد تکراری هم ثبت نمی‌شود. برای حذف و ویرایش نیز دستورهای UpdateQuery/DeleteQuery و Fill را داخل try می‌گذاریم و در catch پیام «خطا در حذف/ویرایش کالا» می‌دهیم.

دستور if فقط خالی‌نبودن کادرها را بررسی می‌کند و کنترل کامل خطا نیست. اگر کد کالا غیرعددی باشد، int.Parse با FormatException خطا می‌دهد؛ در حالت بدون try-catch برنامه متوقف می‌شود و با try-catch پیام خطا نمایش داده می‌شود.

۲۵۷ صفحه ۲۴۶ کارگاه ۱۹

فعالیت منزل: مدیریت خطا را برای کدهای سربرگ مشتری با استفاده از متد TryParse انجام دهید.

فعالیت کارگاهی (فیلم «ذخیره برگه خرید در برنامه فروشگاه»): پس از مشاهده‌ی فیلم، برنامه‌ی جستجوی مشتری و کالا و درج در سبد خرید و ثبت برگه خرید در سربرگ را بنویسید.

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

TryParse بدون ایجاد استثنا بررسی می‌کند متن عدد است یا نه (خروجی bool) و در صورت موفقیت عدد را در پارامتر out می‌گذارد:

```
private void btnAddCustomer_Click(object sender, EventArgs e)
{
    int code;
    if (int.TryParse(txtCustomerCode.Text, out code)
        && txtCustomerName.TextLength > 0)
    {
        try
        {
            customersTableAdapter.InsertQuery(code, txtCustomerName.Text,
                                              txtPhone.Text);
            customersTableAdapter.Fill(myShopDataSet.Customers);
        }
        catch
        {
            MessageBox.Show("Error in saving the customer");
        }
    }
    else
        MessageBox.Show("Customer code must be a number");
}
```

ثبت برگه خرید (طراحی خودمان به‌جای فیلم): ۱) با کد مشتری و Enter نام مشتری پر می‌شود؛ ۲) با کد کالا و Enter نام و قیمت کالا نشان داده می‌شود؛ ۳) با «درج در سبد خرید» سطر به dgvFactors افزوده و مبلغ کل محاسبه می‌شود؛ ۴) با «ثبت برگه خرید»، یک رکورد در Factors و برای هر سطر یک رکورد در FactorItems درج می‌شود و موجودی کالاها کم می‌شود:

```
private void btnSaveFactor_Click(object sender, EventArgs e)
{
    if (dgvFactors.Rows.Count <= 1) return;     // cart is empty
    int factorCode = int.Parse(txtFactorCode.Text);
    try
    {
        factorsTableAdapter.InsertQuery(factorCode,
            int.Parse(lblCustomerCode.Text), DateTime.Now.ToString("yyyy/MM/dd"));
        for (int i = 0; i < dgvFactors.Rows.Count - 1; i++)
        {
            int goods  = int.Parse(dgvFactors.Rows[i].Cells[0].Value.ToString());
            int amount = int.Parse(dgvFactors.Rows[i].Cells[2].Value.ToString());
            int price  = int.Parse(dgvFactors.Rows[i].Cells[3].Value.ToString());
            factorItemsTableAdapter.InsertQuery(factorCode, goods, amount, price);
            goodsTableAdapter.UpdateStockByCode(amount, goods); // stock = stock -
amount
        }
        dgvFactors.Rows.Clear();
        lblAllCost.Text = "0";
        MessageBox.Show("Invoice saved");
    }
    catch
    {
        MessageBox.Show("Error in saving the invoice");
    }
}
```

سربرگ کاربران: برای محدود کردن دسترسی، پس از ورود موفق، فیلد TypeOfAccess کاربر نگه‌داشته می‌شود و سربرگ فقط برای مدیر فعال می‌شود:

```
// LoginForm: public string access;  -> set from myShopDataSet.Users.Rows[0]
["TypeOfAccess"]
if (frmLogin.access != "Admin")
    tabShop.TabPages.Remove(tabPageUsers);   // hide Users tab
```

در این سربرگ یک DataGridView برای جدول Users و دکمه‌های «ثبت»، «حذف» و «ویرایش» مانند کارگاه‌های ۱۴ تا ۱۶ (با usersTableAdapter.InsertQuery، DeleteQuery، UpdateQuery) نوشته می‌شود.

این جواب ممکن است دقیق نباشد. (فیلم «ذخیره برگه خرید» در دسترس نبود؛ برنامه بر اساس ساختار جدول‌ها طراحی شد و نام متدهای TableAdapter پیشنهادی است.)

۲۵۸ صفحه ۲۴۷ پروژه فاز ۵

در این پودمان آرایه‌های ایجادشده برای نگهداری اطلاعات را حذف کرده و اطلاعات را از پایگاه داده دریافت کرده و در پایگاه داده ذخیره می‌کنیم. پروژه را به پایگاه داده‌ی کتابخانه که در پودمان ۱ ایجاد کردید، متصل کنید. موارد زیر را به پروژه اضافه کنید: ثبت زمان login کردن مدیر در یک پرونده به نام Log.txt همراه با مدیریت خطا؛ اضافه کردن دکمه «دریافت پرونده» و کادر متنی برای دریافت مسیر ذخیره‌سازی پرونده (در صورت کلیک روی دکمه «دریافت پرونده»، پرونده‌ی Log.txt را در مسیر تعیین‌شده در کادر متن ذخیره کند)؛

اضافه کردن دو سربرگ نمایش اعضا و نمایش کتاب‌ها که داده‌های جدول اعضا و کتاب‌ها در این سربرگ مشاهده شود؛ فراهم کردن امکان حذف اعضا و ویرایش مشخصات آنها؛ فراهم کردن امکان حذف کتاب و ویرایش عنوان آنها.

ثبت زمان ورود مدیر در Log.txt (با مدیریت خطا) و دریافت پرونده:

```
// after a successful admin login
private void WriteLog()
{
    try
    {
        string line = String.Format("{0}\t{1}\r\n",
            txtUserName.Text, DateTime.Now);
        File.AppendAllText("Log.txt", line);
    }
    catch
    {
        MessageBox.Show("Could not write the log file");
    }
}

private void btnGetFile_Click(object sender, EventArgs e)
{
    try
    {
        string dest = Path.Combine(txtSavePath.Text, "Log.txt");
        File.Copy("Log.txt", dest, true);   // true = overwrite
        MessageBox.Show("File saved");
    }
    catch
    {
        MessageBox.Show("Invalid path or file not found");
    }
}
```

سربرگ‌های «نمایش اعضا» و «نمایش کتاب‌ها»: هر کدام یک DataGridView (dgvMembers و dgvBooks) دارند که با Fill پر می‌شوند:

ایران توشه
توشه ای برای موفقیت

```
private void MainForm_Load(object sender, EventArgs e)
{
    membersTableAdapter.Fill(libraryDataSet.Members);
    dgvMembers.DataSource = membersBindingSource;
    booksTableAdapter.Fill(libraryDataSet.Books);
    dgvBooks.DataSource = booksBindingSource;
}

private void btnDeleteMember_Click(object sender, EventArgs e)
{
    if (txtMemberCode.TextLength > 0)
        try
        {
            membersTableAdapter.DeleteQuery(int.Parse(txtMemberCode.Text));
            membersTableAdapter.Fill(libraryDataSet.Members);
        }
        catch { MessageBox.Show("Error in deleting the member"); }
}
private void btnUpdateMember_Click(object sender, EventArgs e)
{
    try
    {
        membersTableAdapter.UpdateQuery(txtMemberName.Text, txtMemberPhone.Text,
            int.Parse(txtMemberCode.Text));
        membersTableAdapter.Fill(libraryDataSet.Members);
    }
    catch { MessageBox.Show("Error in editing the member"); }
}
private void btnDeleteBook_Click(object sender, EventArgs e)
{
    try
    {
        booksTableAdapter.DeleteQuery(int.Parse(txtBookCode.Text));
        booksTableAdapter.Fill(libraryDataSet.Books);
    }
    catch { MessageBox.Show("Error in deleting the book"); }
}
private void btnUpdateBook_Click(object sender, EventArgs e)
{
    try
    {
        booksTableAdapter.UpdateTitleQuery(txtBookTitle.Text,
            int.Parse(txtBookCode.Text));
        booksTableAdapter.Fill(libraryDataSet.Books);
    }
    catch { MessageBox.Show("Error in editing the book"); }
}
```

پرسوجوها: DELETE و UPDATE روی Members و Books با شرط کلید اصلی (WHERE ... = ?). حذف عضو دارای امانت فعال در catch مدیریت می‌شود.

این جواب ممکن است دقیق نباشد. (نام جدول‌ها و فیلدهای پایگاه داده‌ی کتابخانه به طراحی هنرجو در پودمان ۱ بستگی دارد؛ نام‌ها نمونه‌اند.)

ایران توشه
توشه ای برای موفقیت