

جزوه خلاصه کتاب

توسعه برنامه سازی و پایگاه داده

پایه یازدهم متوسطه (۲۱۱۲۸۷)

تالیف مهر ۱۴۰۲

توضیحات:

- هنرآموز شبکه و نرم افزار رایانه
- ویژه آزمون آموزش و پرورش
- خلاصه شده در ۲۱ صفحه
- حیطة تخصصی

برای دانلود رایگان جدیدترین سوالات استخدامی هنرآموز شبکه، اینجا بزنید

برای دانلود رایگان مرجع این جزوه، کتاب توسعه برنامه سازی و پایگاه داده یازدهم اینجا بزنید

« انتشار یا استفاده غیر تجاری از این فایل، بدون حذف لوگوی ایران عرضه، مجاز می باشد »



پیاده سازی پایگاه داده

تولید روزافزون داده ها در عصر حاضر و نیاز به دسترسی سریع و مناسب به آنها، ایجاد پایگاه داده را اجتناب ناپذیر کرده است. عملیات اصلی روی داده شامل درج، حذف، بهنگام سازی، جستجو و سازمان دهی در پایگاه داده قابل انجام است و در صورت طراحی مناسب، سرعت و سهولت استفاده از داده ها را تأمین می کند. با طراحی درست جدول ها و ارتباط بین آنها، تولید گزارش های مختلف برای تحلیل داده ها امکان پذیر می شود و ضمن اجتناب از ذخیره سازی افزونه، نیازهای اطلاعاتی محیط کسب و کار پاسخ داده می شود.

نرم افزار ۲۰۱۶ Access دارای ویژگی هایی از جمله مدیریت داده در قالب ساختار مشخص و قابل تعریف به وسیله کاربر و مدیریت دسترسی به اطلاعات است.

تاریخچه ذخیره سازی و بازیابی اطلاعات

بشر از ابتدای خلقت تا به امروز با داده ها و اطلاعات مختلفی سروکار داشته است. اولین داده های ثبت شده به وسیله انسانها شامل تصاویری است که همیشه با آنها روبه رو بوده، آنها را در حافظه خود ثبت و نگهداری میکنند و برای دسترسی به این اطلاعات تنها کافی است که آنها را به خاطر بیاورد.

اولین ابزار محاسباتی احتمالاً انگشتان دست و پا بوده است. انسان های اولیه محاسبات عددی را با شباهت زیادی به آنچه شما روی تخته سفید انجام میدهید، روی دیوار غارها انجام می دادند. اما شواهد نشان میدهد که آنها از یک سری ابزار نیز برای ثبت اعداد بهره می گرفتند.

در نتیجه گذشت زمان و افزایش هوش و مهارت آنها و نیاز بیشتر آنها به ذخیره سازی اطلاعات، درک آنها از فناوری نیز با کشف ابزارها بیشتر شد. نیاز به ذخیره سازی، بازیابی، اشتراک گذاری و استفاده از اطلاعات با حجم روزافزون، میتواند یکی از دلایل به وجود آمدن رایانه های امروزی و پایگاه های داده Databases باشد.

وجود حجم انبوه داده ها و اطلاعات در همه زمینه ها، نیاز به جستجو و استخراج اطلاعات جدید از آنها را بیش از پیش پر رنگ تر ساخته است. به همین دلیل با استفاده از داده کاوی Data Mining که مبتنی بر علوم مختلفی شکل می گیرد و از ابزارها و امکانات مختلفی بهره می برد، سعی میشود طی فرایندی از انبوه داده ها، دانش استخراج کند و بانک اطلاعاتی از نیازهای اولیه آن به شمار می رود.

در واقع ایجاد پایگاه داده ها و فناوری آن، تأثیر زیادی در افزایش استفاده از رایانه ها داشته است و در حوزه های خدمات، صنعت، کشاورزی و هنر نقش مهمی ایفا می کند. در قدیم هنرستانها به دلیل استفاده نکردن از رایانه و پایگاه داده های مربوط و مناسب، با مشکلات زیادی روبهرو بودند و انجام کارها به کندی صورت می گرفت.

برخی از این مشکلات عبارتند از:

- * وجود داده های تکراری Redundancy
- * وجود تداخل Conflict یا مغایرت در داده ها
- * دشواری در به روز رسانی Update

مفاهیم پایگاه داده

محیط عملیاتی

محلی که میخواهیم برای آن پایگاه داده ایجاد کنیم، یک محیط عملیاتی است. هنرستانی که برای آن میخواهیم پایگاه داده ایجاد کنیم، یک محیط عملیاتی محسوب میشود. کتابخانه و یا تاکسی تلفنی نیز می تواند یک محیط عملیاتی باشد. در تمامی این موارد با ایجاد پایگاه داده مناسب میتوان بهراحتی اطلاعات را در قالب پایگاه داده سازماندهی و با سهولت بیشتری آنها را مدیریت کرد و به اطلاعات موردنیاز با اطمینان دستیابی داشت.

موجودیت (Entity)

هر شخص، محل، شی (Object) یا مفهومی در دنیای واقعی که می خواهیم درباره آن اطلاعاتی را در پایگاه داده ذخیره کنیم، موجودیت نام دارد.

صفت های موجودیت

هر موجودیت دارای تعدادی صفت (attribute) است. صفت برای بیان ویژگی های یک موجودیت استفاده میشود. برای مثال، موجودیت کتاب میتواند دارای صفت های عنوان، شابک و قیمت باشد.

صفت کلیدی Primary key

صفتی که می توان با استفاده از آن، موجودیت را به صورت یکتا و منحصر به فرد شناسایی کرد، صفت کلیدی می گویند. برای مثال هنگامی که مدیر از نام «علی» برای صدا زدن یکی از هنرجویان استفاده کرد، به دلیل منحصر به فرد نبودن مقدارهای صفت نام، مشخص نبود که کدامیک از هنرجویان مخاطب مدیر بوده اند. در محیط عملیاتی هنرستان، هنرجویان را میتوان با استفاده از شماره دانش آموزی به صورت یکتا شناسایی کرد. شماره دانش آموزی، صفت مناسبی برای کلید بودن است، زیرا هیچ دو هنرجویی دارای شماره دانش آموزی یکسان نیستند. بنابراین صفت کلید برای یک موجودیت، صفت یا مجموعه صفاتی است که مقدار آن مشخص و غیر تکراری است.

ارتباط (Relationship)

در هر محیط عملیاتی، ارتباط منطقی بین موجودیت ها و صفت ها وجود دارد. پس از تعیین موجودیت ها و صفت های مربوط به آنها، می باید ارتباط بین آنها شناسایی شود. در پایگاه داده رابطه ای (Relational database)، داده های مربوط به هر موجودیت در جدول سازماندهی میشود و برای جلوگیری از تکرار داده ها، مقادیر مختلف داده های در جدول های مجزا نگهداری میشود.

ارتباط یک به یک (one_to_one یا 1:1)

فرض کنید در محیط عملیاتی اداره آموزش و پرورش، موجودیت اول، مدیر و موجودیت دوم، هنرستان باشد. اگر قانونی را در نظر بگیریم که به وسیله آن هر مدیر تنها بتواند در یک هنرستان به عنوان مدیر فعالیت کند در این شرایط هر هنرستان تنها یک مدیر دارد و هر مدیر تنها مدیریت یک هنرستان را به عهده دارد. بنابراین یک ارتباط یک به یک بین دو موجودیت برقرار است. در ارتباط یک به یک، هر نمونه از یک موجودیت حداکثر با یک نمونه از موجودیت دیگر ارتباط دارد.

ارتباط یک به چند one_to_many یا 1:N

هر هنرآموز رشته شبکه و نرم افزار رایانه چندین عنوان درسی را درس میدهد و از طرف دیگر، چند عنوان درسی به وسیله تنهای ک هنرآموز ارائه میشود. در ارتباط یک به چند، هر نمونه از موجودیت اول ممکن است با چند نمونه از موجودیت دوم ارتباط داشته باشد، اما هر نمونه از موجودیت دوم نمیتواند با چند نمونه از موجودیت اول در ارتباط باشد.

ارتباط چند به چند many_to_many یا M:N

محیط عملیاتی اداره آموزش و پرورش هنرآموزان زیادی دارد و هر هنرآموز میتواند در چندین هنرستان تدریس کند. در ارتباط چند به چند، هر نمونه از یک موجودیت میتواند با چند نمونه از موجودیت دیگر ارتباط داشته باشد و برعکس. توجه داشته باشید که اینجا، هنرستان به عنوان محیط عملیاتی مطرح نیست، بلکه به عنوان یک موجودیت در نظر گرفته شده است.

نمودار رابطه بین موجودیتها ERD: اولین قدم پس از شناسایی موجودیتها و صفات آنها، تعیین ارتباط موجودیتهاست. با استفاده از ERD (Relationship Entity Diagram) می توان به صورت گرافیکی موجودیت ها و روابط بین آنها را نمایش داد.

برای ترسیم این نمودار، نمادهایی را قرارداد میکنیم:

مستطیل: برای نمایش موجودیتها

بیضی: برای نمایش صفات موجودیتها

لوزی: برای نشان دادن رابطه یا عملکرد بین موجودیتها

زیرخط: برای مشخص کردن صفت کلیدی

پایگاه داده یکی از انواع روش های ذخیره و بازیابی اطلاعات Information Storage and Retrieval است و به کاربر امکان می دهد تا داده ها و اطلاعات خود را به صورت سازماندهی شده ذخیره کرده، با سرعت بیشتری به اطلاعات دسترسی داشته و بتواند آنها را مدیریت و پردازش کند. از ویژگیهای پایگاه داده میتوان به طراحی یکپارچه و مدیریت متمرکز آن اشاره کرد که باعث جلوگیری از تکرار در ذخیره سازی (افزونگی) داده ها خواهد شد و در نتیجه از حافظه استفاده بهتری صورت میگیرد. با پیشرفت هایی که در حوزه شبکه های رایانه رخ داده است، پایگاه داده می تواند توسط چند کاربر به طور همزمان مورد استفاده قرار گیرد.

سیستم مدیریت پایگاه داده (DBMS)

سیستم مدیریت پایگاه داده Database Management System برنامه ای است که عمل ذخیره سازی، بازیابی، امنیت و به طور کل ارتباط با پایگاه داده را کنترل می کند. کاربران درخواست خود را به این نرم افزار ارسال میکنند و از طریق آن با پایگاه داده ارتباط برقرار می کنند. در واقع سیستم مدیریت پایگاه داده واسط بین پایگاه داده و کاربران و نرم افزارها است.

Access یکی از نرم افزارهای مدیریت (خلاصه شده توسط ایران عرضه) پایگاه داده رابطه ای است که به وسیله شرکت Microsoft در بسته نرم افزاری Office گنجانده شده است. داده ها در Access به وسیله جدولهایی نگهداری می شوند که داده های آنها به هم مرتبط هستند. برای مثال جدول مشتری میتواند به داده های جدول سفارشیهای Access مرتبط باشد. رابطه بین جدول های مرتبط را ایجاد و حفظ کرده، در نتیجه میتوان به راحتی داده های یک مشتری و تمامی سفارش های او را بازیابی کرد.

فیلد (Field) و رکورد (Record)

هر موجودیت دارای صفتهای مختلفی است که هر یک از این صفتها فیلد نامیده میشود. هر ستون جدول نشان دهنده یک فیلد است. برای مثال در محیط عملیاتی هنرستان، موجودیت هنرجو دارای صفتهای مختلفی از جمله شماره دانش آموزی، نام، نام خانوادگی و شماره ملی است که هر کدام از این موارد یک فیلد محسوب میشود.

هر فیلد دارای نوع داده ای خاصی است. به عبارت دیگر، نوع داده ای یک فیلد مشخص می کند چه نوع اطلاعاتی می تواند در آن قرار گیرد.

مجموعه ای از فیلدهای مربوط به یک نمونه موجودیت را یک رکورد میگویند. هر سطر جدول نشان دهنده یک رکورد است.

جدول ۲- ویژگی های رایج فیلدها

ویژگی	توضیحات
Field Size	تعداد نویسه های نوع فیلد Short Text را به تعداد مشخصی از ۱ تا ۲۵۵ نویسه محدود می کند. مقدار پیش فرض ۲۵۵ نویسه است. برای تعیین نوع دقیق داده عددی فیلد Number باید از ویژگی Field Size استفاده شود.
Format	قالب بندی نمایش داده در نمای Datasheet را تعیین می کند.
Input Mask	قالب داده در هنگام ورود اطلاعات را تعیین می کند. مثلاً برای ورود شماره تلفن ثابت قالب خاص * * * * _ * * * * * (*) می توان مشخص کرد. در این حالت کاربر ابتدا باید کد شهر و سپس شماره تلفن ثابت را وارد کند. نمونه داده: ۰۲۱- ۱۲۳۴۵۶۷۸
Decimal Places	تعداد ارقام سمت راست نقطه اعشار را تعیین می کند. برای نوع داده های Currency، Single، Double و Decimal که زیرمجموعه نوع Number هستند، استفاده می شود.
Caption	برچسب اختیاری است که به جای نام فیلد، در جدول، فرم ها و گزارش ها نمایش داده می شود.
Default Value	مقدار پیش فرضی است که در زمان ایجاد یک رکورد جدید، اگر مقداری به فیلد مربوطه اختصاص داده نشده باشد، در آن ذخیره می شود.
Validation Rule	شرطی است که برای ورود داده ها باید رعایت شود. به طور مثال، عدد وارد شده باید کمتر از ۱۰۰ باشد یا تاریخ وارد شده باید قبل از تاریخ روز جاری باشد.
Validation Text	متنی است که در صورت رعایت نکردن شرط Validation Rule به عنوان پیام خطا برای کاربر نمایش داده خواهد شد.
Required	مشخص می کند که در زمان ورود داده ها، حتماً باید برای این فیلد مقدار وارد شود و نمی تواند خالی رها شود.
Allow Zero Length	مشخص می کند که آیا می توان رشته خالی "" را برای فیلدی از نوع Short Text یا Long Text وارد کرد.
Index	باعث افزایش سرعت دسترسی اطلاعات خواهد شد. در صورتی که برای آن مقدار YES انتخاب شود، می توان تعیین کرد که مقدار تکراری در آن فیلد مجاز باشد یا خیر.

کلید خارجی (Foreign key)

کلید خارجی به فیلدی از یک جدول گفته میشود که رکوردی را به صورت منحصر به فرد و یکتا در جدول دیگر مشخص میکند. به عبارت دیگر، کلید خارجی در جدول دوم تعریف میشود اما به کلید اصلی در جدول اول اشاره دارد و ارتباط بین جدولها را امکانپذیر میسازد. برای مثال در پایگاه داده school، جدول teach دارای فیلد کد کارمندی (Teacher Code) مربوط به هنرآموز است که آن را به عنوان کلید خارجی در نظر میگیریم و مشخص میکند که هر درس به وسیله کدام هنرآموز تدریس میشود. در واقع ارتباط بین دو جدول teach و teacher از طریق همین فیلد برقرار میشود. همچنین با قراردادن فیلدهای کد درس (Course Code) و شماره دانش آموزی (Student Code) به عنوان کلید خارجی در جدول teach، ارتباط جدول teach با جدولهای course و student نیز برقرار می شود.

ایجاد ارتباط بین جدولها و کنترل یکپارچگی

پیش از ایجاد ارتباط بین جدول ها لازم است بین آنها ی کپارچگی صورت بگیرد. فرض کنید در جدولی اطلاعات هنرآموزان نگهداری میشود و در جدول دیگر جزئیات کلاس های آنها وجود دارد. از آنجایی که این دو جدول با یکدیگر در ارتباط هستند اگر نام هنرآموزی از جدول هنرآموزان حذف شود، اطلاعات مربوط به این هنرآموز در جدول جزئیات کلاس نیز باید حذف شود، در غیر اینصورت این اطلاعات بالتکلیف میمانند. برای جلوگیری از چنین ناسازگاری ها، باید در زمان ایجاد ارتباط بین جداول، یکپارچگی را نیز برای آنها در نظر بگیریم. Access این امکان را به کاربر می دهد که برای جلوگیری از حذف اطلاعات و آسیب رسیدن به آنها و همچنین کنترل یکپارچگی، از امکانی به نام Referential Integrity استفاده کند. البته میتوان بدون کنترل یکپارچگی، ارتباط بین جدولها را ایجاد کرد اما امتیاز استفاده از کنترل یکپارچگی در این است که Access روی ارتباط بین جدولها در زمان به روزرسانی، حذف و سایر عملیات روی رکوردها، کنترل و نظارت خواهد داشت.

برای اینکه Integrity Referential به درستی اعمال شود باید شرایط زیر وجود داشته باشد:

- * هر دو جدول متعلق به یک پایگاه داده باشند.
- * فیلدهای برقرارکننده ارتباط در هر دو جدول، دارای نوع دادهای یکسان باشند.
- * فیلد برقرارکننده ارتباط در جدول اصلی، بهعنوان کلید اصلی تعریف شده باشد.

شایستگی توسعه پایگاه داده

SQL زبانی است که برای ارتباط با پایگاه داده مورد استفاده قرار میگیرد. با استفاده از دستورات SQL میتوان مواردی از جمله به روزرسانی یا ویرایش داده ها و همچنین بازیابی داده ها از یک پایگاه داده را انجام داد. بسیاری از پایگاه های داده رابطه ای از جمله Access، از این زبان پشتیبانی میکنند. بیشتر کارهایی که به وسیله واسط کاربری پایگاه داده انجام میشود، از طریق کد نویسی با SQL نیز امکان پذیر هستند. در واقع Access در پشت صحنه، برای اجرای وظایفی که با استفاده از واسط کاربری برنامه انجام میدهم، از SQL استفاده میکند.

SQL نسبت به بزرگی و کوچکی حروف، حساس نیست. اما برای خوانایی بیشتر، دستورات را معمولا با حروف بزرگ مینویسند.

دستور INSERT INTO یکی از دستورات SQL است که برای اضافه کردن رکورد به جدول استفاده میشود.

شکل کلی دستور INSERT INTO

```
INSERT INTO table_name [(field1[, field2[,...]])]
VALUES (value1[, value2[,...])
```

نکته: در زمان نوشتن دستورات SQL، مقادیر رشته ای باید حتما در داخل ' ' قرار بگیرند.

با استفاده از دستور UPDATE، میتوان مقادیر درون فیلدهای یک جدول را بر اساس معیار (criteria) مشخصی تغییر داد. عملکرد این دستور مشابه فرمان and Replace Find است که در نرم افزارهای دیگر Office از جمله Microsoft Word با آن آشنا شده اید.

شکل کلی دستور UPDATE

UPDATE table
SET newvalue
WHERE criteria

جدول ۴- بخشهای مختلف دستور UPDATE

بخش	توضیحات
table	نام جدولی که قصد تغییر اطلاعات آن را داریم.
newvalue	مقادیر جدیدی که قرار است در فیلدهای خاصی از رکورد به روزرسانی شده، قرار گیرد.
criteria	عبارتی که مشخص می کند چه رکوردهایی به روزرسانی شوند، تنها رکوردهایی که دارای این معیار هستند، به روزرسانی خواهند شد.

با استفاده از دستور DELETE میتوان رکورد یا رکوردهایی را از یک یا چندین جدول بر اساس معیار مشخصی که با عبارت WHERE تعیین میشود، حذف کرد.

شکل کلی دستور DELETE

DELETE
FROM table_name
WHERE criteria

جدول ۵- بخش های مختلف دستور DELETE

بخش	توضیحات
table_name	نام جدولی که قرار است رکوردهای آن را حذف کنید.
criteria	عبارتی که تعیین می کند چه رکوردهایی از جدول حذف شوند.

پرس و جو (Query)

جستجو در اطلاعات یک پایگاه داده یکی از مهمترین عملیاتی است که روی آن انجام میشود. برای مثال در پایگاه داده هنرستان.

پرس و جو، درخواستی برای دریافت داده ها و اطلاعات از جدول یا جدول های یک پایگاه داده است. خروجی این درخواست ممکن است با استفاده از SQL بازگردانده شود و یا به صورت تصویری مانند نمودار یا نتایج پیچیده باشد. آنچه در اینجا مورد نظر است، استفاده از SQL برای ایجاد پرس و جو و دریافت نتیجه به صورت جدولی از داده ها است. داده های بازگردانده شده به عنوان نتیجه پرس و جو در جدولی نمایش داده خواهند شد که به آن مجموعه نتایج set-result میگویند. البته پرس و جو در مفهوم گسترده تر شامل تمامی دستوراتی است که تا به اینجا مورد بررسی قرار گرفتهاند. تمامی دستورات SQL از جمله، UPDATE، DELETE، INSERT INTO بهنوعی به عنوان ابزار پرس و جو در نظر گرفته میشود. گاهی یک پرس و جو تغییری را در پایگاه داده ایجاد نمی کند و تنهای ک خروجی را دربر خواهد داشت مانند دستور SELECT. گاهی نیز دستوراتی مانند INSERT INTO یا UPDATE، تغییراتی را روی داده های جدولها اعمال میکنند اما خروجی خاصی را باز نمی گردانند.

دستور SELECT

این دستور رایجترین دستور SQL است. دستور SELECT به موتور پایگاه داده Access دستور میدهد که اطلاعاتی از یک پایگاه داده را به صورت مجموعه ای از رکوردها بازگرداند. بهعبارتدیگر از این دستور برای انتخاب و استخراج اطلاعات از یک یا چند جدول و سپس مشاهده نتایج در یک جدول موقت استفاده میشود.

شکل کلی دستور SELECT

```
SELECT field_list
FROM table_name
[WHERE ...]
[GROUP BY ...]
[ORDER BY ...]
```

جدول ۶- بخش های مختلف دستور SELECT

بخش	توضیحات
field – list	نام فیلدهای حاوی داده است که می‌خواهیم آنها را بازیابی کنیم. اگر بیش از یک فیلد مشخص شده باشد، با همان ترتیب قرار گرفته‌شده، بازیابی می‌شوند.
table – name	نام جدولی که حاوی فیلدهایی است که رکوردها از آن انتخاب می‌شوند.

در فرایند اجرای پرس و جو، ابتدا موتور پایگاه داده Access جدول یا جدولهای مشخص شده را جستجو کرده، ستون‌های مورد نظر را استخراج میکند، سطرهایی که با معیار تعیین شده مطابقت دارند را انتخاب می‌کند و سپس بر اساس ترتیب خواسته شده، سطرها را مرتب‌سازی و گروه‌بندی کرده، آنها را نمایش می‌دهد.

نکته: برای تشخیص اینکه فیلد متعلق به کدام جدول است، از شکل " نام فیلد، نام جدول " استفاده میشود.

نکته: در شکل کلی دستورات SQL، علامت | به معنی "یا" است و هنگامی که بین دو مقدار مختلف قرار میگیرد، میتوان یکی از آن دو را انتخاب کرد و نباید از هر دو مقدار در یک دستور استفاده کرد. از تاپ علامت های [] و |، اجتناب کنید.

جدول ۷- بخشهای مختلف SELECT به همراه BY ORDER

بخش	توضیحات
field_list	نام فیلد یا فیلدهایی که قرار است بازیابی شوند. در این قسمت می‌توان نام مستعار ستون‌ها را نیز مشخص کرد.
table_name	نام جدولی که قرار است فیلدها از آن بازیابی شوند.
selectcriteria	مشخص‌کننده معیار انتخاب است. در صورتی که از WHERE استفاده شود، Access پس از اعمال معیار جست‌وجو برای رکوردها، آنها را مرتب می‌کند.
field1, field2	نام فیلدهایی که مرتب‌سازی بر اساس آنها انجام خواهد شد.

استفاده از عبارت ORDER BY اختیاری است. اگر از عبارت ORDER BY استفاده نشود، رکوردها با همان ترتیبی که در جدول قرار دارند، نمایش داده میشوند. در صورتیکه جلوی عبارت ORDER BY، چندین فیلد ذکر شود، مرتب‌سازی ابتدا بر اساس فیلد اول و در صورت یکسان بودن مقادیر فیلدها، بر اساس فیلد دوم خواهد بود. اگر صعودی Ascending یا نزولی Descending بودن مرتب‌سازی مشخص نشود، ترتیب نمایش پیشفرض به صورت صعودی از ۰ تا ۹ و از A تا Z خواهد بود.

گزارش چیست؟

یکی از نیازهای کاربران، ایجاد گزارش از اطلاعات موجود در پایگاه داده است. اگر شخصی مانند معاون اجرایی بخواهد از موارد مختلفی از جمله تعداد هنرجویانی که ثبت‌نام کرده‌اند، تعداد هنرآموزان یا هنرجویان یک رشته خاص و مواردی از این قبیل گزارشی تهیه نماید، چگونه میتواند این کار را انجام دهد؟ گزارش روشی برای نمایش و چاپ اطلاعات خلاصه شده محسوب میشود که از کارایی و انعطاف پذیری بسیاری برخوردار است. با تولید گزارش میتوان اطلاعات را با هر سطحی از جزئیات در معرض مشاهده

قرار داد و یا آن را با قالب های مختلف چاپ کرد. بیشتر اوقات گزارشها دارای خلاصه ای از اطلاعات ذخیره شده در پایگاه داده هستند. داده های موجود در گزارشها میتوانند گروه بندی شده، با ترتیب دلخواه مرتب سازی شوند. علاوه بر آن میتوان در صورت نیاز برای آنها جمع کل مقادیر در نظر گرفت و از سایر عملگرهای آماری نیز در آنها استفاده کرد. گزارشها میتوانند شامل تصویر و سایر عناصر گرافیکی باشند. همچنین میتوان برای آنها عنوان، سرصفحه، پاصفحه و شماره صفحه نیز در نظر گرفت.

گزارشها در Access در نماهای مختلفی قابل مشاهده هستند. از مهمترین آنها میتوان به موارد زیر اشاره کرد:

نمای Layout :

در این نما، داده های موجود در گزارش را میتوان با شباهت زیادی به آنچه در چاپ ظاهر میشود، مشاهده و تغییراتی را در طراحی گزارش ایجاد کرد. از آنجا که در هنگام تغییر گزارش، میتوان داده ها را نیز مشاهده کرد، نمای مناسبی برای تنظیم پهنای ستون ها، اضافه کردن سطوح گروه بندی به حساب می آید. انجام هرگونه تغییراتی که در ظاهر و خوانایی گزارش تأثیر دارد نیز در این نما امکانپذیر است.

نمای Design :

این نما برای طراحی گزارش استفاده میشود و جزئیات بیشتری از ساختار گزارش در اختیار کاربر قرار میدهد. میتوان سرصفحه، پاصفحه مربوط به گزارش را تنظیم کرد. از آنجا که گزارش در این نما در حالت اجرایی قرار نمی گیرد، نمی توان در هنگام کار کردن با آن، داده ها را مشاهده کرد. اعمال برخی تنظیمات در این نما، نسبت به نمای Layout راحت تر (تهیه شده توسط سایت ایران عرضه) انجام میشود.

نمای Preview Print :

در این نما میتوانید گزارش را مانند آنچه روی کاغذ چاپ میشود مشاهده و آن را تأیید نهایی کنید.

برخی روشهای مختلف برای ایجاد گزارش در ۲۰۱۶ Access

Report: ایجاد یک گزارش ساده به صورت جدول.

Report Design: یک گزارش خالی در نمای طراحی ایجاد کرده و به کاربر این امکان را میدهد تا فیلدها، اطلاعات و کنترلهای موردنظر را در آن وارد کند.

Blank Report: یک گزارش خالی در نمای Layout ایجاد کرده و فهرستی از فیلدها را در اختیار کاربر قرار میدهد تا از میان آنها فیلدهای موردنظر را برای نمایش در گزارش انتخاب نماید.

Report Wizard: از طریق wizard به کاربر این امکان را میدهد تا گزارش ایجاد کند.

پودمان دوم

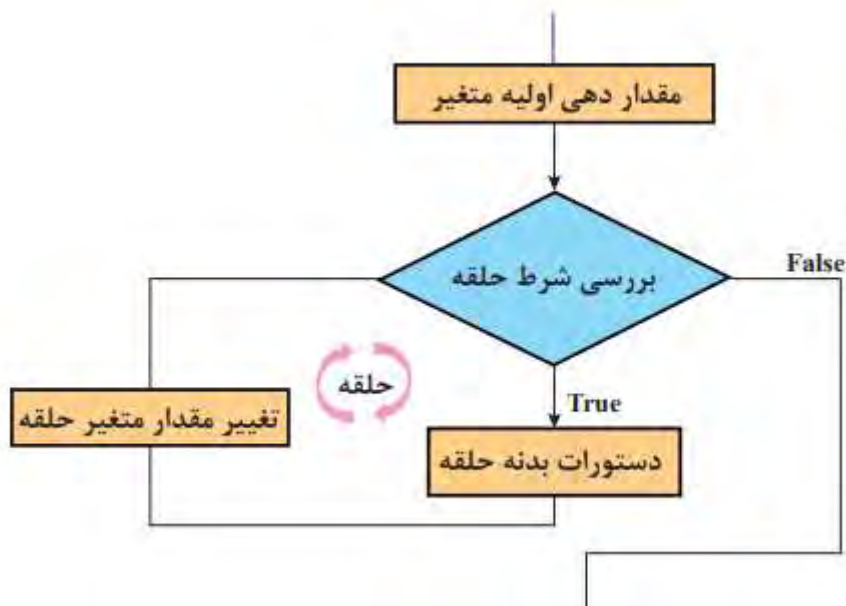
مدیریت مجموعه داده

هر مسئله دارای بخشهای مختلفی، شامل داده، عملیات، حالتیهای شرط و تکرار است. تشخیص هر کدام از این بخش ها به حل درست مسئله منجر میشود. یک زبان برنامه نویسی برای پیاده سازی مسئله برای هر یک از این موارد، دستوراتی دارد که باید از خوانایی و سهولت استفاده برخوردار باشند. در برخی از مسئله ها نیاز به ایجاد ساختار تکرار و یا تعدادی داده همونوع داریم که نمایش آنها در ساختار آرایه مناسب است. آرایه ها شامل داده های ی همونوع و همنام هستند که در برخی مسائل نظیر جستجو و مرتب سازی استفاده از آنها ضروری است. زبان برنامه نویسی C# دارای روشهای متنوعی برای کار با انواع دستورات شرط، تکرار و آرایه است.

حلقه for

ساختار تکرار for برای پیاده سازی حلقه معین مناسب است.

روندنمای حلقه for



ابتدا مقداردهی اولیه متغیر انجام شده (۱)، در صورت صحیح بودن شرط حلقه (دستورات بدنه حلقه ۳) اجرا میشود، پس از اجرای دستورات بدنه حلقه، مقدار متغیر حلقه تغییر میکند (۴) و دوباره شرط حلقه بررسی میشود. با نادرست شدن شرط حلقه کنترل برنامه به دستورات پس از حلقه منتقل میشود.
نکته: برای سرعت برنامه نویسی در VS، پس از نوشتن دستور for دو بار کلید tab را فشار دهید و دستور را ویرایش کنید.

حلقه while

آیا تا به حال بازی حدس عدد را با دوستان خود انجام داده اید؟ در این بازی بازیکن اول عددی را در ذهن خود انتخاب کرده و بازیکن دوم عددهایی را حدس میزند تا وقتی که عدد را پیدا کند. به نظر شما آیا از ابتدا مشخص است که بازیکن دوم با چند حدس به جواب میرسد؟ آیا میزان مسافتی که خودرو با یک باک پر از بنزین طی می کند، همیشه یکسان است؟ آیا میدانید در زمانی که چراغ راهنمایی یک تقاطع سبز است، چه تعدادی خودرو از تقاطع عبور میکنند؟ تعیین تعداد خودروهایی که از چهارراه تا زمانی که چراغ راهنمایی سبز است، عبور می کنند یک حلقه نامعین است. چون تعداد خودروهایی که می توانند در آن زمان از چهار راه عبور کنند مشخص نیست. آیا مشخص است که یک فرد برای دریافت گواهینامه راهنمایی و رانندگی چند بار آزمون می دهد؟ همه مثال های بالا یک نوع حلقه نامعین هستند، که از ابتدا تعداد تکرار مشخص نیست. در اینگونه موارد اغلب از دستور while استفاده می شود.

<p>روندمای حلقه while</p>	<p>ساختار دستور while</p> <p>(شرط حلقه) while</p> <pre>{ دستور یا دستورات }</pre>
----------------------------------	--

در صورت درست بودن شرط حلقه، دستور یا دستورات بدنه حلقه اجرا میشود و شرط خروج از حلقه while، نادرست بودن شرط حلقه است.

حلقه های متداخل

در برخی از مواقع، یک دستور تکرار را در داخل دستور تکرار دیگری به کار میبریم. به عبارت دیگر هنگامیکه در داخل یک حلقه، حلقه دیگری قرار داشته باشد، حلقه های تو در تو یا متداخل نامیده می شوند.

دبیر ورزش از هنرجویان خود خواسته است تا پنج بار دور حیاط مدرسه دویده، در پایان هر دور سه حرکت بارفیکس انجام دهند.

میخواهیم در یک مجتمع ۱۰ طبقه ای که هر طبقه دارای ۸ واحد است، غذای نذری پخش کنیم. چند غذا برای این مجتمع نیاز است؟

همه این مثالها کارهای تکراری است که در درون کار تکراری دیگری قرار دارند و کاربرد حلقه های متداخل را بیان میکنند.

آرایه

آرایه، فهرستی از عناصر هم نوع است که تحت یک نام مشترک کنار هم در حافظه ذخیره میشوند. هر خانه آرایه، یک عنصر آرایه نامیده میشود. برای تفکیک و دسترسی به هر عنصر، از یک عدد صحیح به نام اندیس استفاده میشود. در زبان سی شارپ، اندیس اولین عنصر آرایه صفر است، اندیس عنصر بعدی یک و به همین ترتیب اندیس عناصر یکی یکی اضافه می شوند.

فهرست اسامی ۵ هنرجو را میتوان به شکل زیر در نظر گرفت:

اولین اندیس	۰	۱	۲	۳	۴	آخرین اندیس
عنصر	سعید	مجید	محمد	علی	حسین	تعداد عناصر عدد ۵ است

تعداد عناصر آرایه را طول آرایه مینامند. آرایه بالا دارای طول ۵ است و عناصر آن از ۰ تا ۴ شماره گذاری شده است.

اعلان آرایه

به یاد دارید برای اعلان یک متغیر از نوع رشته ای به شکل زیر عمل میکردیم:

```
string name;
```

در زبان سی شارپ میتوان تعریف و ایجاد آرایه را در دو مرحله انجام داد. برای تعریف یک آرایه که از نوع رشته ای است، در مرحله اول، مشابه اعلان متغیر رشته ای به شکل زیر عمل میشود:

```
string[] name;
```

در این دستور متغیر `name`، آرایه ای از نوع رشته ای اعلان شده است؛ اما تعداد خانه های آرایه مشخص نیست و حافظه ای برای آرایه تخصیص نیافته است. در مرحله دوم با استفاده از عملگر `new` و مشخص کردن اندازه آرایه، حافظه مناسب به آرایه اختصاص داده شده، آرایه ایجاد میشود.

```
name = new string[5];
```

نمایش عناصر آرایه

برای نمایش محتوای عناصر آرایه میتوان از متدهای `Write` یا `Line Write` استفاده کرد. برای مثال برای نمایش محتوای عناصر اول و دوم آرایه `price` خواهیم داشت:

```
Console.WriteLine(price[0]);  
Console.Write(price[1]);
```

گاهی لازم است به بخشی یا تمامی عناصر آرایه مراجعه شود. برای مثال معلم هنگام حضور و غیاب، اسامی را از ابتدا تا انتهای فهرست هنرجویان می خواند. به این عمل پیمایش یا مرور آرایه میگوئیم.

فرض کنید میخواهیم اسامی روزهای هفته را نمایش دهیم. یک روش آن است که عناصر آرایه را از ابتدا تا انتهای کی یکی نمایش دهیم.

```
string[] week = {"saturday", .....,"friday"};
Console.WriteLine (week[0]);
Console.WriteLine (week[1]);
Console.WriteLine (week[2]);
Console.WriteLine (week[3]);
Console.WriteLine (week[4]);
Console.WriteLine (week[5]);
Console.WriteLine (week[6]);
```

در این روش باید به تعداد عناصر آرایه از متد WriteLine استفاده شود.

اگر تعداد عناصر آرایه زیاد باشد و بخواهیم تمام عناصر آن را پیمایش کنیم، چه روشی پیشنهاد میکنید؟ برای پیمایش آرایه های بزرگ، از حلقه تکرار استفاده میشود. برای مثال دستورات نمایش اسامی روزهای هفته را میتوان به صورت زیر نوشت:

```
for(int i = 0; i < 7; i++)
    Console.WriteLine(week[i]);
```

عملیات روی آرایه

آرایه یک نوع داده است که میتواند تعداد زیادی داده همنوع را نگهداری کند. به وسیله حلقه های تکرار می توانیم عملیات یکسانی روی تمام یا برخی از عناصر آرایه انجام دهیم.

جستجو در آرایه

فرض کنید در دفتر تلفن میخواهیم شماره تلفن یک شخص خاص را پیدا کنیم. اگر اسامی دفتر تلفن مرتب نباشد، لازم است از ابتدای دفتر تلفن نام مورد نظر را به ترتیب با تمام اسامی موجود در دفتر تلفن مقایسه کنیم. نام مورد نظر در هرکجای دفتر تلفن میتواند باشد، بنابراین عمل مقایسه تا یافتن نام موردنظر و یا رسیدن به انتهای دفتر تلفن ادامه مییابد. به این روش جست وجو، جست وجوی خطی یا ترتیبی (Search Linear) میگویند. جست وجوی یکی از عملیات رایج و پرکاربرد روی آرایه ها است. در جست وجو، شماره مکان عنصر مورد جست وجو مشخص میشود.

اگر اسامی دفتر تلفن بر اساس حروف الفبا مرتب باشند، به سرعت میتوان نام موردنظر را پیدا کرد. در این حالت وسط دفتر را باز میکنیم، اگر نام مورد نظر را دیدیم، جست وجو پایان یافته است، در غیر این صورت با توجه به ترتیب الفبایی و نام موردنظر، در اسامی قبلی یا بعدی دفتر تلفن جست وجو را ادامه میدهیم. به این روش جست وجو، جست وجوی دودویی (Binary Search) میگویند.

کلاس Array

کتابخانه غنی NET Framework دارای کلاسهای آماده فراوانی است. این کلاس های آماده و از قبل ایجادشده Built_in دارای اعضای مانند ویژگی ها و متدها هستند که انجام عملیات را آسانتر کرده، به توسعه سریعتر نرم افزار کمک میکنند. با کلاس Console و برخی متدهای آن مثل WriteLine و ReadLine آشنا هستیم. برای عملیات مختلف روی آرایه ها از کلاس آماده Array استفاده می کنیم.

پودمان سوم

طراحی واسط گرافیکی

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

آشنایی با پروژه های ویندوزی

با ورود سیستم عامل های گرافیکی به دنیا پراپانه، زبانهای برنامه نویسی به وجود آمدند که در محیط گرافیکی ایننوع سیستم عاملها قادر به اجرا و فعالیت باشند. از جمله این زبانهای برنامه نویسی میتوان به #C، Visual Basic، Microsoft Visual C++، Borland Delphi و #C اشاره کرد. خروجی این نوع برنامه ها در محیط ویندوز و با ظاهری زیبا و کاربرپسند قابل مشاهده است.

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

واسط گرافیکی کاربر

پروژه های کنسول با پروژه های ویندوزی تفاوتی دارند. در پروژه های ویندوزی، صفحه ای وجود دارد که روی آن ابزارهایی برای ورود اطلاعات، نمایش خروجی ها و کلیدهایی برای ثبت و کنترل پروژه قرار میگیرند. ابزارهای طراحی و نمایش خروجی های پروژه ویندوزی روی صفحه ای قرار میگیرند که به آن فرم (Form) میگوییم. فرم دربرگیرنده همه اجزای گرافیکی پروژه است و به عنوان یک واسط بین اجزای گرافیکی برنامه و کاربر عمل میکند. به همین دلیل به آن واسط گرافیکی کاربر (GUI) میگویند.

کنترل

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

ویژگی های فرم

با شناخت ویژگیهای یک کنترل و تنظیم آنها میتوانید ظاهری زیبا برای پروژه طراحی کنید. برای مقداردهی ویژگی های یک کنترل در VS از پنجره ویژگی ها استفاده می کنیم.

واکنش برنامه به رویدادها

هنگام کار با ویندوز برای انجام عملیات مختلف بارها از راست کلیک استفاده کرده اید. آیا راست کلیک در بخش های مختلف ویندوز و برنامه های متفاوت نتیجه یکسان دارد؟ راست کلیک یک رویداد Event و واکنش برنامه به آن، متد رویداد است. رویداد، یک کلاس یا یک شیء را قادر میسازد تا دیگر کلاس ها و اشیاء را از رخ دادن اتفاقی باخبر کند. واکنشی که برنامه در مقابل رویدادها نشان خواهد داد، باید پیش بینی شده و متدهای مربوط به آنها نوشته شود. به این متدها Event Handler میگویند. متد قطعه کدی شامل تعدادی دستور است که این مجموعه دستورات با فراخوانی متد اجرا می شوند.

در پروژه های کنسولی، برنامه معمولا به وسیله دستورات متنی اجرا میشود. در صورتیکه مشخصه اصلی پروژه های ویندوزی رویدادگرا بودن آن است یعنی برنامه در زمان اجرا به رویدادهای مختلف اشیای برنامه مانند Componentها، کنترلرها و دیگر کلاس ها واکنش نشان میدهد.

شکل عمومی متد رویدادها

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

نام پیش فرض متد رویداد به صورت " نام رویداد - نام کنترل " است. تا زمانی که شما دستوری در بدنه متد btnExit_Click ننویسید، این متد کاری انجام نمیدهد. برای پاک کردن محتوای کادرهای متن، متد رویداد Click دکمه " پاک کردن " را بنویسید. محتوای کادر متن در ویژگی Text آن نگهداری میشود. برای تغییر این محتوا باید ویژگی Text را مقداردهی کنید.

شکل کلی تغییر مقدار ویژگی یک کنترل هنگام کدنویسی

مقدار جدید ویژگی = نام ویژگی. نام کنترل

نکته: برای تبدیل مقدار یک متغیر عددی به نوع رشته ای از متد ToString استفاده میشود.

کادرهای محاوره ای

در برنامه های مختلف کادرهایی شبیه شکل زیر بسیار دیده آید. به این کادرها، کادر محاوره ای DialogBox میگوییم. از کادر محاوره ای برای نمایش پیام و دریافت اطلاعات از کاربر استفاده میشود. کادر محاوره ای میتواند براساس موقعیت برنامه دارای ظاهری متفاوت باشد. برخی از کادرهای محاوره ای دارای عملکرد از پیش تعیین شده هستند. مانند کادر محاوره ای OpenFileDialog برای انتخاب پرونده، و ColorDialog برای انتخاب رنگ و FontDialog برای انتخاب قلم که در بیشتر برنامه ها استفاده میشوند.

کادر پیام

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



شکل ۲۶- کادر محاوره ای پیام

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

جدول ۵ - پارامترهای متد Show

پارامتر	نوع	توضیح
پیام	string	متنی است که در کادر نمایش داده می شود. این پارامتر اجباری است.
عنوان	string	متنی است که در نوار عنوان کادر نمایش داده می شود.
دکمه ها	نوع داده شمارشی MessageBoxButtons	دکمه هایی را مشخص می کند که باید در کادر پیام نمایش داده شوند.
نماد	نوع داده شمارشی MessageBoxIcon	نماد کادر پیام
دکمه پیش فرض	نوع داده شمارشی MessageBoxDefaultButton	دکمه پیش فرض کادر پیام را تعیین می کند.

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

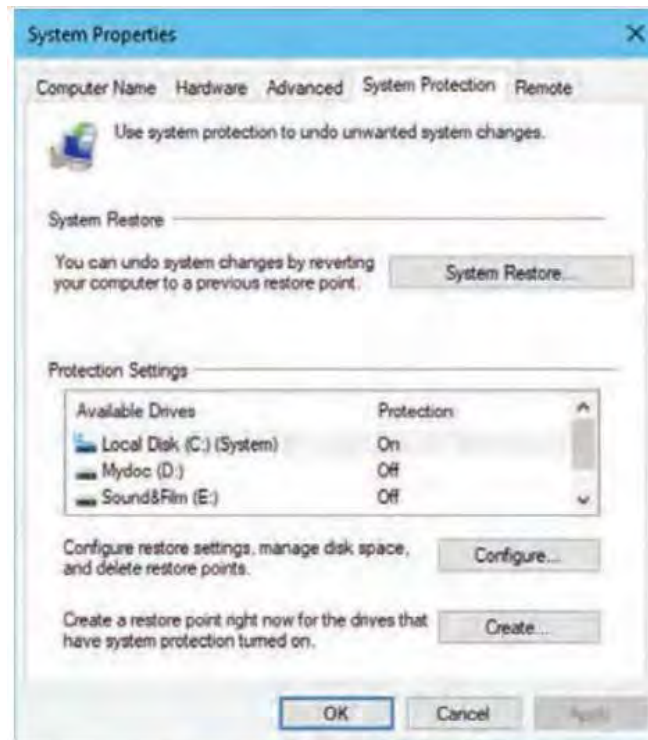
کنترل های دکمه رادیویی (RadioButton) و کادر علامت (CheckBox)

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



کنترل سربرگ TabControl

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



شکل ۳۶- کنترل سربرگ در پنجره های ویندوزی

کنترل های کادر فهرست (Listbox) و کادر ترکیبی (ComboBox)

کنترل کادر فهرست به کاربر امکان میدهد که از بین عناصری که در فهرست وجود دارد، یک عنصر را انتخاب کند. کادر ترکیبی از یک کادر فهرست و یک کادر متن تشکیل میشود که عنصر انتخاب شده از کادر فهرست در کادر متن قابل نمایش است و ویژگی Text، متن موجود در کادر متن را نشان میدهد مهمترین ویژگیها و متدهای کادر فهرست و کادر ترکیبی در جدول ۸ و ۹ نمایش داده شده است.

جدول ۸- ویژگی های مهم کادر فهرست و کادر ترکیبی

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

جدول ۹- متدهای مهم کادر فهرست و کادر ترکیبی

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

پودمان چهارم

توسعه واسط گرافیکی

با توسعه ساختارهای کنترلی در زبان برنامه نویسی و ایجاد واسط گرافیکی مناسب، میزان تعامل کاربران با برنامه کاربردی افزوده شده است و محیطهای طراحی شده در قالب استاندارد گرافیکی عرضه میشود. در واقع با راهنمایی کاربر در محیط نرم افزار تولید شده از بروز خطاهای احتمالی، جلوگیری میشود. کار با تجهیزات ورودی سخت افزاری نظیر صفحه کلید و ماوس، ممکن است محدودیت هایی را هنگام ورود داده به رایانه، ایجاد کند. در محیط برنامه نویسی #C کلاس های آماده و متنوعی برای کنترل عملکرد صفحه کلید و ماوس وجود دارد که سبب میشود واسط گرافیکی کاربر طوری توسعه یابد که میزان تعامل کاربر با نرم افزار تولید شده را تا حد زیادی افزایش دهد. در این پودمان رویدادهای صفحه کلید و ماوس و کاربرد برخی کنترل های پیشرفته مرتبط با واسط گرافیکی نظیر کادر محاورهای رنگ و قلم و منو و متدهای کار روی نویسه و رشته را فراخواهید گرفت.

عملکرد ماوس در ویندوز و برنامه های کاربردی

یکی از مهمترین ابزار کار با واسط گرافیکی کاربر در سیستم عامل ویندوز و برنامه های کاربردی، ماوس است، به همین دلیل کار با ماوس در برنامه نویسی ویندوز اهمیت فراوانی دارد.

رویدادهای ماوس وابسته به دکمه ماوس

در اینجا به رویدادهایی که با عملکرد دکمه های ماوس، روی کنترل ها رخ میدهد، رویدادهای ماوس وابسته به دکمه میگوییم. هر ماوس، دارای حداقل سه دکمه است که با فشردن یکی از سه دکمه، رویداد ماوس وابسته به دکمه، رخ میدهد.

جدول ۱- برخی رویدادهای ماوس وابسته به دکمه ماوس

شرح رویداد در VS	شرح رویداد	نام رویداد
Occurs when the control is clicked by the mouse.	زمانی رخ می‌دهد که کنترل با ماوس کلیک شود.	MouseDown
Occurs when the control is double clicked by the mouse.		MouseDoubleClick
Occurs when the mouse pointer is over the control and a mouse button is pressed.		MouseDown
Occurs when the mouse pointer is over the control and a mouse button is released.		MouseUp

رویدادهای ماوس مستقل از دکمه

با حرکت ماوس روی نمادهای برنامه در میز کار و نوار وظیفه چه تغییراتی در شکل نمادها مشاهده میکنید؟ لحظه ای ماوس را بدون آنکه کلیک کنید، روی یکی از نمادهای نوار وظیفه نگهدارید. چه مشاهده میکنید؟ با حرکت ماوس و بدون کلیک در فضای برنامه های ویندوزی هم رویدادهایی رخ میدهد. ما در اینجا به این رویدادها، رویدادهای ماوس مستقل از دکمه میگوییم.

کلاس Cursors در سی شارپ، دارای مجموعه ای از اشاره گرهای ماوس است. برای تغییر شکل اشاره گر ماوس روی کنترل میتوان ویژگی Cursor کنترل را تغییر داد. کلاس ToolTip برای ایجاد متن راهنما در یک کادر مستطیلی کوچک استفاده می شود.

آرگومان های رویدادهای ماوس

به متد رویداد نمونه زیر توجه کنید:

```
private void Form1_MouseMove(object sender, MouseEventArgs e)
```

معمولا در متد رویدادها، دو آرگومان sender و e وجود دارد:

- آرگومان sender، معمولا کلاسی است که این رویداد به وسیله آن منتشر میشود.
- در رویداد Mouse Move آرگومان e از کلاس MouseEventArgs است که شامل مختصات اشارهگر ماوس نسبت به کنترل و نوع دکمه ماوس و تعداد دفعات کلیک و چرخش آن است. ویژگیهای آرگومان e در رویداد MouseEventArgs و MouseEventArgs با هم تفاوت دارد.

چندین کنترل با یک متد رویداد

در متدهای رویداد آرگومان sender معمولا از نوع object است. برای دسترسی به ویژگیهای کنترلی که رویداد برای آن رخ داده است باید این آرگومان را به نوع آن کنترل تبدیل کنیم.

```
sender (نام کلاس کنترل) = نام متغیر نام کلاس کنترل
```

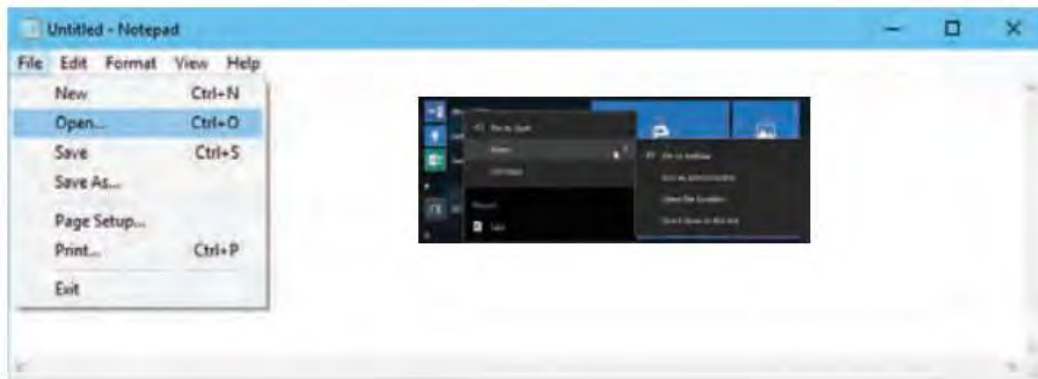
به طور مثال، اگر رویداد برای کنترل Label باشد، دستور بالا به صورت زیر نوشته میشود:

```
Label lbl = (Label) sender;
```

دستور بالا یک تبدیل نوع از object به Label انجام میدهد و برنامه نویس برای مدیریت دکمه ای که رویداد روی آن رخ داده کافی است، نام نمونه کنترل (lbl) را در برنامه استفاده کند.

ایجاد منو

یکی از اجزای مهم واسط گرافیکی کاربر در سیستم عامل ویندوز، منو (Menu) است که دسترسی آسان به امکانات نرم افزار کاربردی را ممکن می سازد. نرم افزار هایی که دارای امکانات فراوانی هستند از ابزار منو برای دسترسی به قابلیت های نرم افزار استفاده میکنند.



برای دیدن پنجره ویژگیهای یک گزینه، روی گزینه راست کلیک کرده و Properties را انتخاب کنید.

منوی استاندارد

در بیشتر نرم افزارها مانند واژه پردازهای ک استاندارد برای منوهای نرم افزار وجود دارد. در سی شارپ هم میتوان یک منوی استاندارد ایجاد کرد و به دلخواه گزینه های منوی ایجاد شده را با توجه به عملیات برنامه خود ویرایش کرد.

عملکرد صفحه کلید در ویندوز

یکی از مهمترین دستگاه های ورود داده به رایانه، صفحه کلید است. برای ورود اطلاعات به برنامه و کنترل برنامه ها از صفحه کلید استفاده میشود.

کلیدهای تحریری، کلیدهایی هستند که با فشردن آنها نویسههای چاپ شدنی Printable Characters روی صفحه نمایش ظاهر میشوند. کلیدهایی که با فشردن آنها علامتی روی صفحه ظاهر نمیشود، کلیدهای غیر تحریری هستند.

رویدادهای صفحه کلید برای کنترلهای دیگر

اگر به پنجره ویژگیهای برخی از کنترلها نظیر کادمتن (TextBox) نگاه کنید، رویدادهای صفحه کلید برای این کنترلها هم وجود دارد. زمانی یک کنترل به رویدادهای صفحه کلید واکنش نشان میدهد که فوکوس روی کنترل باشد.

نوع داده char نویسه (char) و متدهای آن

متغیر نوع char میتواند (طراحی شده توسط ایران عرضه) فقط شامل یک نویسه باشد و به صورت زیر اعلان میشود:

```
char ch = 'A';
```

نوع داده char در سیشارپ دارای متدهایی است که نوع نویسه شامل رقم بودن، حرف کوچک بودن، حرف بزرگ بودن و ... را مشخص میکند.

متد رویداد KeyDown کنترل txtStudentNO

```
private void txtStudentNO_KeyDown(object sender, KeyEventArgs e)
{
    e.SuppressKeyPress = true;
}
```

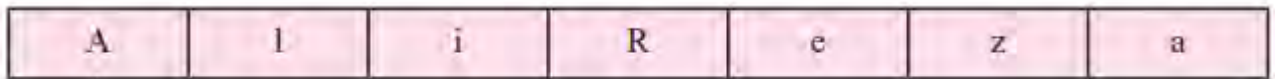
هر متغیر از نوع رشته به صورت زیر اعلان میشود:

نام متغیر رشته‌ای `string`

هنگام اعلان یک متغیر رشته ای میتوان آن را مقداردهی اولیه کرد.

`string name = "AliReza";`

هر متغیر رشته‌ای، مجموعه متوالی از نویسه هاست که میتوان به آسانی به آنها دسترسی پیدا کرد.



اولین نویسه رشته `name`، عنصر `name[0]` است که در آرایه، نویسه `A` است.

پودمان پنجم

مدیریت پایگاه داده

مدیریت پرونده و پوشه های موجود در سیستم عامل به وسیله زبان برنامه نویسی موجب کارایی نرم افزار های تولیدشده و استفاده از امکانات سیستمعامل در محیط برنامه نویسی میشود. در زبان برنامه نویسی #C کلاس های آماده‌ای وجود دارد که قابلیت دسترسی به پرونده، پوشه و دیسک گردانهای دیسک سخت در رایانه را برای کاربر فراهم میکند. در محیط برنامه نویسی #C امکان اتصال برنامه های تولیدشده به بانک اطلاعاتی به وسیله کنترل های پیشرفته وجود دارد تا بتوان محتویات پایگاه داده را به وسیله برنامه مدیریت کرد. در این پودمان مفهوم و کاربرد انواع پرونده متنی و غیرمتنی و پوشه و متدهای ایجاد و دسترسی به پرونده ها و نحوه اتصال برنامه به پایگاه داده به وسیله واسطه ای برنامه نویسی را فرا خواهید گرفت.

پرونده و انواع آن

در هنگام کار با یک نرم افزار اداری، رایانه شما بهطور اتفاقی خاموش میشود. اگر آخرین تغییرات برنامه را در یک پرونده ذخیره نکرده باشید، برای داده ها و اطلاعاتی که ذخیره نکرده اید، چه اتفاقی میافتد؟ متغیرها و آرایههایی که برای نگهداری داده ها در برنامه تعریف شده اند، در حافظه اصلی سیستم ذخیره میشوند. این داده ها تا زمانی قابل استفاده هستند که برنامه درحال اجرا است و پس از خاتمه برنامه دیگر وجود نخواهند داشت. حافظه اصلی سیستم یک حافظه موقت است و داده های آن با قطع برق پاک میشود. برای حفظ و نگهداری داده های برنامه باید به روشی آنها را بهصورت دائمی ذخیره کرد. استفاده از پرونده های که روش متداول برای ذخیره داده ها روی حافظه جانبی است. پرونده دنباله‌ای از بایت ها است که روی حافظه جانبی تحت یک نام نگهداری میشود. برای دسترسی به پرونده ها از نام و نشانی آنها در حافظه جانبی استفاده میشود. برای ذخیره داده های مختلف، از انواع پرونده های متفاوتی استفاده میشود. سیستم عامل ویندوز نوع پرونده را از روی پسوند آن تشخیص میدهد. از نظر محتوا، پرونده ها به دو دسته تقسیم میشوند.

پرونده های متنی (TextFile): پرونده ای که محتوای آن نویسه‌های چاپشده است، پرونده متنی نامیده میشود.

پرونده های دودویی File Binary: اگر داده های موجود در پرونده مانند اعداد و رشته ها به همان شکلی که در حافظه قرار دارند، در پرونده ذخیره شوند، پرونده را دودویی می گوئیم.

محتوای پرونده های متنی با یک ویرایشگر ساده مانند Notepad و یا یک واژه‌پرداز مانند Word قابل مشاهده است. درحالیکه محتوای پرونده های دودویی را نمیتوان با برنامه های ویرایشگر به درستی مشاهده کرد.

نوشتن در پرونده متنی

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

۱ - باز کردن پرونده به منظور نوشتن اطلاعات

۲- نوشتن داده ها و اطلاعات درون پرونده باز شده

۳- بستن پرونده پس از اتمام کار هر سه عملیات میتواند با متدهای مختلفی انجام شود.

همچنین متدهایی وجود دارند که هر سه کار را پشت سرهم با یک دستور انجام میدهند.

شکل کلی متد WriteAllText

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

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

خواندن از پرونده متنی

همانطور که برای خواندن یک کتاب، ابتدا آن را باز کرده، شروع به خواندن صفحه ای میکنیم و در انتها کتاب را میبندیم، برای خواندن داده های یک پرونده نیز باید عملیات زیر را انجام دهیم:

۱ باز کردن پرونده به منظور خواندن داده ها

۲ خواندن داده های پرونده

۳ بستن پرونده پس از اتمام کار

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

مدیریت خطا در برنامه

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

مشاهده فهرست پرونده ها و پوشه ها

برنامه File Explorer ویندوز، ابزاری برای کاوش پرونده ها و پوشه های روی سیستم است. این برنامه به کاربر امکان میدهد که درایو های سیستم را انتخاب کند، پوشه ها را باز کند و بتواند پرونده های درون پوشه ها را مشاهده کند. همچنین این برنامه امکان انجام عملیاتی نظیر ایجاد نسخه مشابه، حذف و جابه جا کردن پرونده ها و پوشه ها را برای کاربر فراهم میکند.

کلاس Directory

در فضای نام System.IO کلاس Directory برای کار با پوشه ها در نظر گرفته شده است. با کمک متدهای موجود در این کلاس میتوان عملیاتی نظیر ایجاد، حذف، ایجاد نسخه مشابه، جابه جایی و برخی کارهای دیگر را روی پوشه ها انجام داد.

کلاس Path

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

کاربرد پایگاه داده

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

خیلی از نرم افزار هایی که ما در محیط پیرامون خود با آنها سروکار داریم، عملیات ذخیره و بازیابی اطلاعات را انجام می دهند. نرم افزارهایی نظیر برنامه فروشگاه ها، بانکها و کتابخانه ها و برخی از نرم افزار های بازی برای ذخیره مشخصات، سطح و امتیاز کاربران از پایگاه داده استفاده میکنند. برخی از برنامه ها و بازیهای تلفنهای هوشمند نیز از پایگاه داده برای ذخیره و بازیابی اطلاعات استفاده میکنند. در ساخت این برنامه ها ابتدا پایگاه داده مربوط به محیط عملیاتی طراحی میشود سپس به کمک زبانهای برنامه سازی نظیر سی شارپ، ویژوال بیسیک، جاوا و php، عملیات ارتباط با پایگاه داده و عملیات مبنایی روی پایگاه داده انجام میشود.

ارتباط با پایگاه داده

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

برای ارتباط پروژه ویندوزی VS ۲۰۱۲ با پایگاه داده Access با قالب (Access ۲۰۱۳ accdb Access) به بالا باید بسته نرم افزار ی Engine Database Access را متناسب با سیستمعامل خود از تارنمای مایکروسافت بارگیری و نصب کنید.

با ایجاد ارتباط برنامه به بانک اشیای زیر به صورت خودکار به فرم پروژه اضافه میشود:

الف) dictionaryDataset که از نوع کلاس Dataset است و برای ذخیره داده های پایگاه داده در حافظه به کار میرود و میتواند اطلاعات یک یا چند جدول را ذخیره کند.
ب) computerTableAdapter ارتباط بین پایگاه داده و نرم افزار کاربردی را مهیا میکند.

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

در کتابخانه NET. مجموعه ای از کلاس های قدرتمند برای اتصال به پایگاه داده و انجام عملیات مختلف روی داده، تحت عنوان ado.net تعریف شده است.

عملیات در پایگاه داده

جستجو، درج، حذف و ویرایش رکوردها از عملیات پر اهمیت در پایگاه داده است. پس از ایجاد ارتباط اولیه پایگاه داده MyShop با برنامه، یک پرونده جدید به نام xsd myShopDataSet در بخش Explorer Solution ایجاد میشود که با دابل کلیک روی آن میتوان جدول ها، نام فیلدهایشان، TableAdapter و متدهای مربوط به آنها را مشاهده کرد. با راست کلیک روی هر کدام از این جدولها و انتخاب گزینه Query از منوی Add می توان یک پرس و جو برای انتخاب داده ها (جستجو) و یا پرس و جو برای عملیات درج، حذف و ویرایش ایجاد کرد.

برای جست و جوی واژه، متد رویداد TextChanged کادرمتن جست و جو (txtSearchWord) استفاده میشود.

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

```
{
```

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

```

computerTableAdapter.FillByWordName(dictionaryDataSet.Computer, txtSearchWord.
Text);
else
computerTableAdapter.Fill(dictionaryDataSet.Computer);
}

```

متد FillByWordName، محتوای کادرمتن را در فیلد WordName جدول جست و جو میکند و جدول داده Computer را با رکوردهایی که پیدا کرده، پر میکند. همزمان کادرفهرست نیز واژه موردنظر و کادرمتن txtWordMeans معنی واژه را نمایش میدهند.

متدی برای ویرایش رکورد

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

شکل کلی دستور SQL ویرایش رکورد

```

UPDATE نام جدول
SET
مقدار ۱ = نام فیلد ۱،
مقدار ۲ = نام فیلد ۲،
.
.
WHERE (شرایط)

```

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

ایران عرضه

مرجع نمونه سوالات

آزمون های استخدامی

به همراه پاسخنامه تشریحی

خدمات ایران عرضه:

- ارائه اصل سوالات آزمون های استخدامی
- پاسخنامه های تشریحی سوالات
- جزوات و درسنامه های آموزشی

برای دانلود رایگان جدیدترین سوالات استخدامی هنرآموز شبکه، اینجا بزنید

برای دانلود رایگان مرجع این جزوه، کتاب توسعه برنامه سازی و پایگاه داده یازدهم اینجا بزنید

« انتشار یا استفاده غیر تجاری از این فایل، بدون حذف لوگوی ایران عرضه، مجاز می باشد »

