

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

بانک اطلاعاتی

رشته کامپیوتر

گروه تحصیلی کامپیوتر

زمینه خدمات

شاخه آموزش فنی و حرفه ای

۵۰۰	یمقانی، محمدرضا
۷۵ /	بانک اطلاعاتی / مؤلف: محمدرضا یمقانی. - تهران: شرکت چاپ و نشر کتاب های درسی
ب ۷۳ ی /	ایران، ۱۳۹۴.
۱۳۹۴	۱۹۹ ص. : مصور. - (آموزش فنی و حرفه ای)
	متون درسی رشته کامپیوتر گروه تحصیلی کامپیوتر، زمینه خدمات.
	برنامه ریزی و نظارت، بررسی و تصویب محتوا: کمیسیون برنامه ریزی و تألیف کتاب های درسی
	رشته کامپیوتر دفتر تألیف کتاب های درسی فنی و حرفه ای و کار دانش وزارت آموزش و پرورش.
	۱. پایگاه های اطلاعاتی. الف. ایران. وزارت آموزش و پرورش. دفتر تألیف کتاب های درسی
	فنی و حرفه ای و کار دانش. ب. عنوان.

همکاران محترم و دانش آموزان عزیز :
پیشنهادات و نظرات خود را درباره محتوای این کتاب به نشانی
تهران - صندوق پستی شماره ۴۸۷۴/۱۵ دفتر تألیف کتاب‌های درسی فنی و
حرفه‌ای و کاردانش، ارسال فرمایند.

info@tvoccd.sch.ir

پیام‌نگار (ایمیل)

www.tvoccd.sch.ir

وب‌گاه (وب‌سایت)

وزارت آموزش و پرورش سازمان پژوهش و برنامه‌ریزی آموزشی

برنامه‌ریزی محتوا و نظارت بر تألیف : دفتر تألیف کتاب‌های درسی فنی و حرفه‌ای و کاردانش

نام کتاب : بانک اطلاعاتی - ۴۵۱/۷

مؤلف : محمدرضا یمقانی

اعضای کمیسیون تخصصی : حمید احدی ، علیرضا جباریه ، سید حمیدرضا ضیایی ، هادی عابدی ، عسگر قندچی ،

محمد مشاهری فرد و ملیحه طزری

آماده‌سازی و نظارت بر چاپ و توزیع : اداره کل نظارت بر نشر و توزیع مواد آموزشی

تهران : خیابان ایرانشهر شمالی - ساختمان شماره ۴ آموزش و پرورش (شهید موسوی)

تلفن : ۹-۸۸۸۳۱۱۶۱، دورنگار : ۰۹۲۶۶۰۸۸۳، کدپستی : ۱۵۸۴۷۴۷۳۵۹

وب‌سایت : www.chap.sch.ir

مدیر امور فنی و چاپ : لیدا نیک‌روش

رسم : حسن عبداله‌زاده، فاطمه رئیسیان فیروزآباد

طراح جلد : طاهره حسن‌زاده

صفحه‌آرا : شهرزاد قنبری

حروفچین : کبری اجابتی

مصصح : علی نجمی ، سیف‌الله بیگ محمددلیوند

امور آماده‌سازی خیر : ناهید خیام‌پاشی

امور فنی رایانه‌ای : حمید ثابت کلاچاهی، پیمان حبیب‌پور

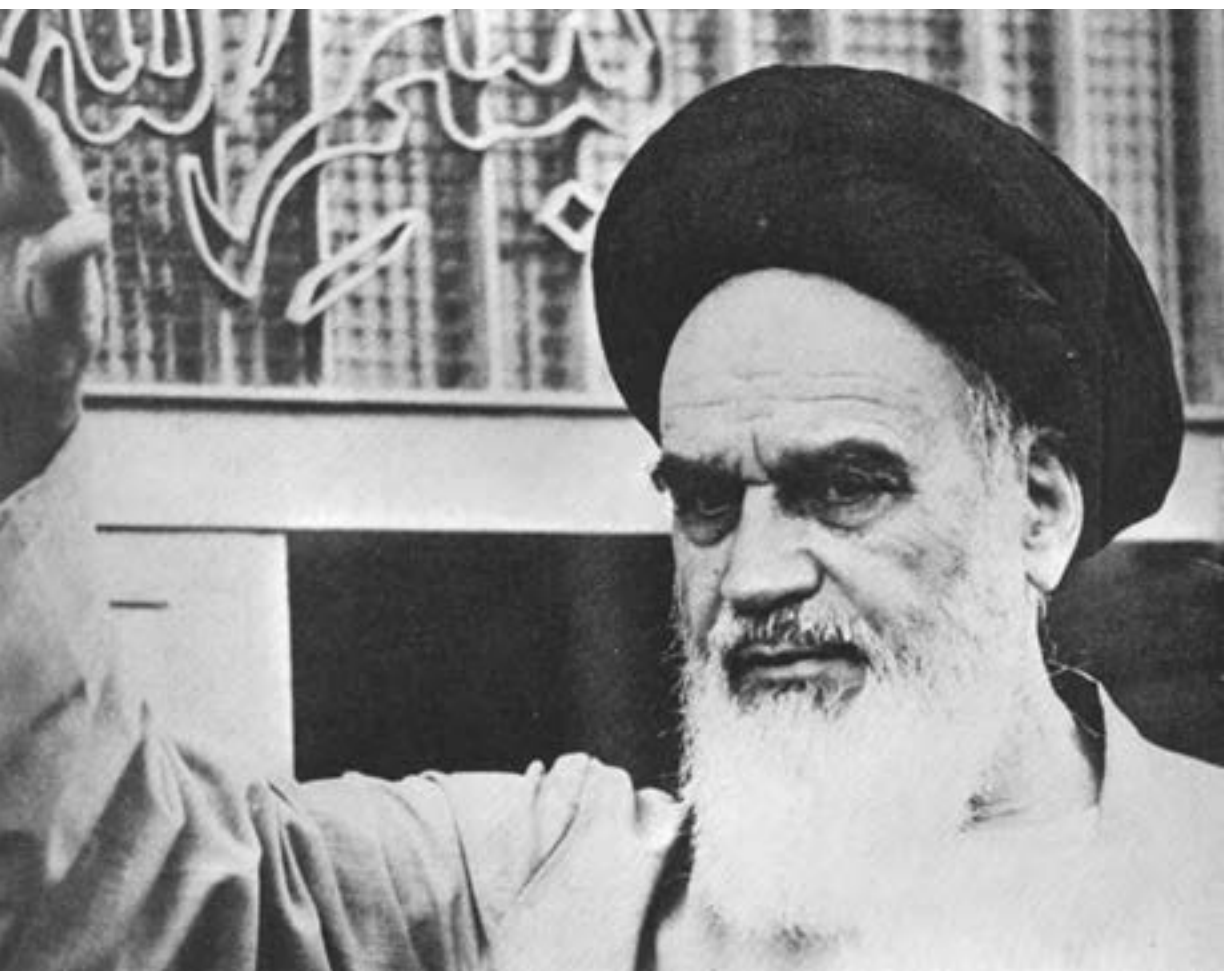
ناشر : شرکت چاپ و نشر کتاب‌های درسی ایران : تهران - کیلومتر ۱۷ جاده مخصوص کرج - خیابان ۶۱ (داروبخش)

تلفن : ۵-۴۴۹۸۵۱۶۱، دورنگار : ۰۴۴۹۸۵۱۶۰، صندوق پستی : ۱۳۹-۳۷۵۱۵

چاپخانه : شرکت چاپ و نشر کتاب‌های درسی ایران «سهامی خاص»

سال انتشار و نوبت چاپ : چاپ دهم ۱۳۹۴

حق چاپ محفوظ است.



شما عزیزان کوشش کنید که از این وابستگی بیرون آید و احتیاجات کشور خودتان را برآورده سازید، از نیروی انسانی ایمانی خودتان غافل نباشید و از اتکای به اجانب بپرهیزید.

امام خمینی «قدس سرّه الشریف»

فهرست

فصل اول : مفاهیم بانک اطلاعاتی

۲	
۲	۱-۱- کلیات
۳	۱-۲- مفاهیم مقدماتی در ذخیره و بازیابی
۵	۱-۳- عملیات روی رکوردها
۷	۱-۴- سیستم فایلینگ (FS)
۸	۱-۵- پایگاه داده‌ها
۹	۱-۶- سیستم مدیریت پایگاه داده
۱۲	۱-۷- مفاهیم پایه پایگاه داده رابطه‌ای
۱۸	۱-۸- قدم‌های طراحی بانک اطلاعاتی
۲۶	خودآزمایی

فصل دوم : مقدمات بانک اطلاعاتی اکسس ۲۰۰۷

۲۷	
۲۷	۲-۱- راه‌اندازی نرم‌افزار اکسس
۲۷	۲-۲- محیط کار، منوها و ابزارهای محیط
۲۹	۲-۳- انواع اشیاء و اجزای پایگاه داده
۳۷	۲-۴- ایجاد یک پایگاه داده در نرم‌افزار اکسس
۳۸	۲-۵- بستن پایگاه داده
۳۸	۲-۶- باز کردن پایگاه داده موجود
۴۰	خودآزمایی

۴۱	فصل سوم : ایجاد و کار با جدول‌ها
۴۳	۳-۱- ایجاد جدول در نمای Design
۵۲	۳-۲- ایجاد جدول به وسیله قالب‌های آماده
۵۴	۳-۳- ایجاد جدول به وسیله نمای Datasheet
۵۷	۳-۴- ایجاد جدول به وسیله Import Table و Link Table
۶۱	۳-۵- ورود داده‌ها به جدول‌ها
۶۴	۳-۶- مرتب‌سازی و فیلتر کردن داده‌ها
۶۵	۳-۷- حذف و اضافه ستون‌ها
۶۶	۳-۸- ایجاد ارتباط بین جدول‌ها
۷۰	خودآزمایی

۷۱	فصل چهارم : ایجاد و کار با فرم‌ها
۷۲	۴-۱- مفاهیم اولیه فرم و اشیای آن
۷۴	۴-۲- ایجاد فرم
۸۰	۴-۳- افزودن رکورد در فرم
۸۱	۴-۴- ویرایش فرم‌ها
۸۴	۴-۵- تغییر طراحی فرم
۸۸	۴-۶- ایجاد فرم جزئیات در فرم اصلی (SubForm)
۹۰	خودآزمایی

۹۱	فصل پنجم : ایجاد و کار با پرس و جوها
۹۲	۵-۱- ایجاد پرس و جو به وسیله Simple Query Wizard
۹۳	۵-۲- ایجاد پرس و جو به وسیله Design View
۹۳	۵-۳- استفاده از جدول در پرس و جو
	۵-۴- انتخاب فیلدهای جدول‌ها / پرس و جوهای اضافه شده مورد نیاز
۹۴	پرس و جو جدید

۹۶	۵-۵ - حذف فیلد از پرس و جو
۹۷	۵-۶ - مرتب سازی نتیجه پرس و جو
۹۸	۵-۷ - معیارها در پرس و جو
۹۹	۵-۸ - استفاده از تاریخ در معیارها
۱۰۳	۵-۹ - فیلدهای محاسباتی
۱۰۷	۵-۱۰ - فیلدهای محاسباتی تجمعی (توابع Aggregate)
۱۰۹	۵-۱۱ - پرس و جوهای عملیاتی Action Query
۱۱۳	۵-۱۲ - ایجاد پرس و جوی CrossTab
۱۱۴	۵-۱۳ - زبان پرس و جوی ساخت یافته SQL
۱۲۷	خودآزمایی

فصل ششم : ایجاد و کار با گزارش ها

۱۳۰	۶-۱ - ایجاد گزارش ساده به وسیله AutoReports
۱۳۲	۶-۲ - ایجاد گزارش به وسیله Report Wizard
۱۴۲	خودآزمایی

فصل هفتم : نرمال سازی

۱۴۴	۷-۱ - ویژگی های رابطه
۱۴۴	۷-۲ - وابستگی تابعی
۱۴۶	۷-۳ - وابستگی تابعی کامل
۱۴۷	۷-۴ - شرح آنومالی ها
۱۴۸	۷-۵ - سطوح نرمال
۱۵۵	خودآزمایی

- ۱۵۷ ۸-۱ - اطلاعات کودکان Children
- ۱۵۷ ۸-۲ - اطلاعات والدین Parents
- ۱۵۷ ۸-۳ - تحلیل
- ۱۵۹ ۸-۴ - استاندارد نام گذاری اشیاء بانک
- ۱۵۹ ۸-۵ - ایجاد پایگاه داده
- ۱۵۹ ۸-۶ - ایجاد جدول ها
- ۱۶۰ ۸-۷ - اصلاح طراحی جدول ها
- ۱۶۳ ۸-۸ - نمای Datasheet جدول
- ۱۶۳ ۸-۹ - استفاده از فرم ها
- ۱۶۵ ۸-۱۰ - روابط Relationship
- ۱۶۵ ۸-۱۱ - افزودن جدول دیگر
- ۱۶۹ ۸-۱۲ - پرس و جوها Queries
- ۱۷۱ ۸-۱۳ - گزارش ها Reports
- ۱۷۶ ۸-۱۴ - اضافه شدن نیازهای جدید
- ۱۷۸ ۸-۱۵ - طراحی منطقی جدول رزرو Booking
- ۱۷۹ ۸-۱۶ - طراحی منطقی فرم رزرو Booking
- ۱۸۰ ۸-۱۷ - ایجاد جدول رزرو Booking
- ۱۸۱ ۸-۱۸ - ایجاد فرم رزرو FormBooking
- ۱۸۶ پروژه های نمونه
- ۱۸۷ ضمیمه ۱ : میانبرهای صفحه کلید در Access
- ۱۸۹ ضمیمه ۲ : واژه نامه تشریحی
- ۱۹۲ ضمیمه ۳ : مرجع سریع Access
- ۱۹۹ منابع

گسترش دانش بشری، تولید «اطلاعات» در حوزه‌های مختلف و نیاز به استفاده کارآمد از آن، محافل علمی، دانشگاهی و فنی را بر آن داشته است که با دقت و تأمل بیشتر و با روش‌های سریع و مناسب از اطلاعات استفاده نمایند. امروزه نرم‌افزارهای مختلف با توانایی‌های متفاوت برای مدیریت داده‌ها ایجاد شده‌اند و استفاده می‌شوند.

در این کتاب مفاهیم عام سیستم‌های ذخیره و بازیابی، فایلینگ و بانک اطلاعاتی در فصل‌های ۱ و ۷ بیان شده است و در فصل‌های ۲ تا ۶ و ۸ این مفاهیم با استفاده از نرم‌افزار مدیریت بانک اطلاعاتی اکسس ۲۰۰۳ به کار گرفته شده است. پیشنهاد می‌شود قبل از انجام کار عملی، مفهوم یا مفهوم‌های مورد نیاز به دقت بیان گردد تا هنرجو بداند «چرا» و برای رسیدن به چه «هدفی» می‌باید از بانک اطلاعاتی استفاده کند.

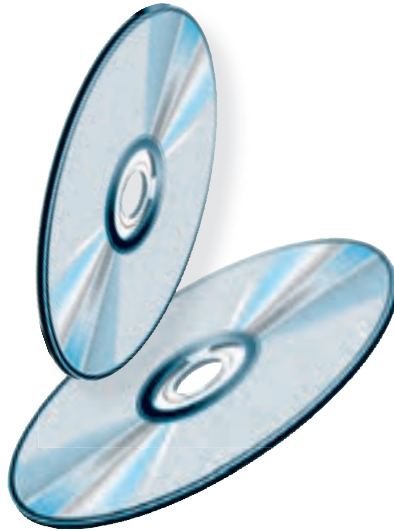
مناسب است با یادآوری مطالب مربوط به فایلینگ در کتاب برنامه‌سازی ۳، تفاوت فایلینگ با بانک اطلاعاتی به درستی شناسایی گردد. در فصل ۸، مراحل ایجاد یک بانک اطلاعاتی نمونه، راهنمایی و توضیح داده شده است.

لازم به ذکر است مفاهیم و مطالب آموزشی کتاب حاضر، قابلیت استفاده در کتاب‌های برنامه‌سازی (۳) و بسته‌های نرم‌افزاری (۳) را دارا می‌باشد. هدف این کتاب یادگیری دقیق مفاهیم اولیه بانک اطلاعاتی و کاربست آن در نرم‌افزار اکسس ۲۰۰۳ است. بنابراین هرگونه سهل‌انگاری در انتقال مفاهیم و تکیه صرف بر انجام عملیات، هنرجو را از اهداف این کتاب دور می‌سازد.

در پایان، از معلمان محترم، دانش‌آموزان گرامی و تمامی خوانندگان ارجمند که مؤلف را از راهنمایی‌های خویش بهره‌مند می‌سازند، سپاسگزاری می‌کنم.

هدف کلی

آشنایی با مفاهیم بانک اطلاعاتی و کار با DBMS متداول



فصل ۱

مفاهیم بانک اطلاعاتی^۱

در این فصل به معرفی تعاریف و مفاهیم اولیه بانک‌های اطلاعاتی خواهیم پرداخت. اگرچه نرم‌افزارهای کاربردی به‌سهولت قابل استفاده هستند، اما بدون دانستن دقیق مفاهیم نمی‌توان از آن‌ها به‌طور مناسب استفاده کرد و هیچ‌گاه نمی‌توان بانک اطلاعاتی کارآمدی را ایجاد و پیاده‌سازی کرد.

هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

- انواع سیستم‌های ذخیره و بازیابی را توضیح دهد.
- مفهوم فیلد، رکورد، فایل را توضیح دهد.
- بانک اطلاعاتی را تعریف کند.
- سیستم مدیریت بانک اطلاعاتی و معماری آن را شرح دهد.
- مفاهیم مدل رابطه‌ای را شرح دهد.
- سیستم بانک اطلاعاتی را طراحی کند.

۱-۱- کلیات

بانک‌های اطلاعاتی یکی از انواع سیستم‌های «ذخیره و بازیابی اطلاعات»^۲ است. سیستم ذخیره و بازیابی سیستمی است که به کاربر امکان می‌دهد تا داده‌ها و اطلاعات خود را ذخیره، بازیابی و پردازش نماید.

۱- در این کتاب پایگاه داده‌ها، بانک اطلاعاتی، بانک داده و بانک به یک معنی استفاده شده است.

نظر به تعریف عام بالا، موارد زیر قابل ذکر است :

۱- کاربر می تواند دارای قابلیت «برنامه سازی» یا فاقد این مهارت باشد.

۲- داده می تواند متن، تصویر، صوت و نظایر آن باشد.

سیستم ذخیره و بازیابی اطلاعات با گذشت زمان، از مفهوم ساده فایلینگ^۱ به صورت زیر توسعه

پیدا کرده است :

۱- سیستم فایلینگ Filing System (FS)

۲- سیستم مدیریت داده ها Data Management System (DMS)

۳- سیستم مدیریت پایگاه داده ها Database Management System (DBMS)

۴- سیستم مدیریت پایگاه دانش Knowledge Base Management System (KBMS)

پیش از تعریف سیستم مدیریت پایگاه داده بهتر است، برخی از مفاهیم، تعاریف مقدماتی و مفهوم

سیستم فایلینگ بررسی شود تا با دقت بیشتری بتوانیم مفهوم بانک اطلاعاتی را بشناسیم.

۱-۲- مفاهیم مقدماتی در ذخیره و بازیابی

۱-۲-۱- فیلد^۲ : یک قطعه داده است^۳.

هر فیلد دارای دو جزء می باشد :

۱- نام

۲- مقدار

مقدار، حکم داده^۴ را دارد و نام فیلد به همراه مقدار در حکم اطلاع^۵ خواهد بود. به عنوان

مثال :

$$\underbrace{\text{نام خانوادگی}} = \underbrace{\text{علوی}} \\ \text{نام فیلد} \quad \text{مقدار فیلد}$$

در واقع هر فیلد یک ویژگی یا صفت به حساب می آید. توجه کنید که از نام فیلد و مقدار فیلد به

همراه هم اطلاع پیدا می کنیم که نام خانوادگی فرد مورد نظر علوی است. مثال جالب تر این که : مقدار

عددی (۸۸۹۶۵۴۲۳) هیچ معنی خاصی ندارد. حال به این عبارت توجه کنید : «تلفن = ۸۸۹۶۵۴۲۳»،

۱- Filing

۲- Field

۳- فیلد واحد معنایی داده است. (Semantic Data Unit)

۴- Data

۵- Information

۶- Attribute

با درج نام فیلد در کنار مقدار عددی، شما اطلاع پیدا می‌کنید که آن عدد شماره تلفن است. اگر به‌جای تلفن، شماره شناسنامه قرار گیرد اطلاع کسب شده متفاوت خواهد بود.



یکی از خصوصیات مهم فیلد، نوع داده‌ای است که در آن قرار می‌گیرد. مثلاً
Real ، Integer یا Text و

۱-۲-۲ رکورد^۱: به مجموعه فیلدهای مرتبط، یک رکورد می‌گوییم.
به‌عنوان مثال، قالب رکورد مربوط به یک دانش‌آموز می‌تواند شامل نام فیلدهای زیر باشد:
نام، نام خانوادگی، سال تولد، مقطع تحصیلی، کلاس و ...
محتوای رکوردهای دانش‌آموزان می‌تواند برای دو دانش‌آموز هم‌کلاسی به‌صورت زیر باشد:
۱- مجید، شجاعی، ۱۳۷۰، اول دبیرستان، کلاس ۱ الف و ...
۲- حسن، کاملیان، ۱۳۷۱، اول دبیرستان، کلاس ۱ الف و ...
محتوای رکورد می‌تواند دارای طول ثابت یا متغیر باشد. در حالت رکورد با طول ثابت، تعداد، ترتیب و اندازه فیلدهای متناظر در تمام رکوردها یکسان در نظر گرفته می‌شود و در حالت رکورد با طول متغیر، چنین الزامی وجود ندارد.

۱-۲-۳ فایل^۲: مجموعه‌ای از رکوردها، تشکیل فایل را می‌دهند. به‌عنوان مثال: فایل تحصیلی دانش‌آموزان، محتوای فیلدهای رکوردهای مربوط به دانش‌آموزان را دارا می‌باشد.

۱-۲-۴ کلید^۳: به فیلد یا زیرمجموعه‌ای از فیلدها، که باعث یکتایی هریک از رکوردها شود، کلید گویند. به‌وسیله کلید می‌توان به یک رکورد مشخص دسترسی داشت. به‌عنوان مثال، به رکوردهای زیر توجه کنید:

ردیف	شماره دانش‌آموزی	نام	نام خانوادگی
۱	۱۰۰	علی	حسینی
۲	۱۱۰	حسن	احمدی
۳	۱۰۸	علی	علوی
۴	۹۴	رضا	علوی

فیلد نام نمی‌تواند کلید باشد زیرا دو مقدار مشابه «علی» در رکوردهای ردیف ۱ و ۳ وجود دارد. هم‌چنین فیلد نام خانوادگی نمی‌تواند کلید باشد زیرا مقادیر «علوی» در رکوردهای ردیف ۳ و ۴ وجود دارد. اما شماره دانش‌آموزی می‌تواند کلید باشد زیرا هیچ دو شماره دانش‌آموزی برابر وجود ندارد. با توجه به رکوردهای این جدول، فیلدهای نام و نام خانوادگی نیز با هم می‌توانند کلید باشند. اگرچه در یک محیط عملیاتی مدرسه واقعی، ممکن است دو دانش‌آموز نام و نام خانوادگی یکسان داشته باشند.



آیا ممکن است در یک سیستم ذخیره و بازیابی کلید وجود نداشته باشد؟

در انتخاب کلید موارد زیر قابل توجه است :

- ۱- کلید می‌تواند ساده یا مرکب باشد.
- ۲- ممکن است کلیدهای مختلفی اعم از ساده یا مرکب بتوان تشخیص داد (کلید کاندید) ولی یکی از آن‌ها را با توجه به ملاحظات به‌عنوان کلید اصلی^۱ در نظر می‌گیریم.
- ۳- در برخی موارد به‌جای استفاده از فیلدها و صفات مورد نیاز شناسایی شده، به‌صورت مجازی فیلدی با مقدار یکتا (صفت مجازی)، به‌عنوان کلید اضافه می‌شود. مانند شماره دانش‌آموزی. مجازی بودن یعنی یک فرد زمانی که در یک مدرسه ثبت‌نام می‌کند و دانش‌آموز آن مدرسه می‌شود، یک شماره دانش‌آموزی می‌گیرد و زمانی که فارغ‌التحصیل می‌شود دیگر این شماره را از دست می‌دهد.
- ۴- مقدار فیلدی که کلید در نظر گرفته می‌شود، تهی (خالی) نیست و طبق تعریف کلید، مقدار تکراری ندارد.

۱-۳- عملیات روی رکوردها

این عملیات عبارت است از :

Retrieve

۱- بازیابی رکوردها

Insert

۲- درج رکوردها

^۱ - Primary Key

Delete

۳- حذف رکوردها

Update

۴- به هنگام سازی رکوردها

Restructure

۵- تغییر ساختار

۱-۳-۱- بازیابی رکوردها (Retrieve): منظور از بازیابی به دست آوردن رکورد یا

رکوردهایی است که دارای شرایط خاصی هستند. به عنوان مثال می‌خواهیم اطلاعات دانش‌آموزانی که معدل آنها بیش از ۱۶ است را به دست آوریم. به این کار بازیابی گفته می‌شود. ممکن است نتیجه بازیابی یک یا چند رکورد باشد.

به طور کلی برای دستیابی و بازیابی رکوردها دو روش وجود دارد:

۱- ترتیبی^۱

۲- مستقیم^۲

در روش ترتیبی، رکوردها از ابتدا یکی پس از دیگری بررسی می‌شوند تا «محتوای» مورد نظر بازیافت شود. مثلاً فرض کنید می‌خواهیم منزل شخصی به نام علی علوی را در یک ساختمان چند طبقه پیدا کنیم. برای یافتن آپارتمان مورد نظر باید مثلاً از طبقه اول، زنگ واحدها را پشت سر هم بزیم تا به منزل این شخص برسیم. این روش کند و زمان بر است.

در روش مستقیم با توجه به محتوا، «آدرس» رکورد به دست می‌آید و رکورد مورد نظر به طور مستقیم با آن آدرس بازیابی می‌شود. روش مستقیم به وسیله ساختارهای مختلف پیاده‌سازی می‌شود مثل شاخص^۲. در مثال بالا اگر به دفتر سرایدار آپارتمان مراجعه کنیم، وی می‌گوید منزل این شخص مثلاً طبقه چهارم، واحد ۲ است. ما می‌توانیم از آسانسور استفاده کنیم و سریع به منزل این شخص برسیم. حتی اگر آسانسور در دسترس نباشد از راه‌پله‌ها خود را به طبقه چهارم می‌رسانیم بدون آن که در طبقه‌ها وقت صرف جستجو کنیم^۲.

۲-۳-۱- درج رکوردها (Insert): منظور، افزودن رکورد یا رکوردهایی است که تاکنون

در سیستم وجود نداشته است. درج می‌تواند در انتهای فایل یا محل منطقی آن رکورد صورت گیرد. محل منطقی، محلی است که نظم فایل را حفظ می‌کند.

به عنوان مثال فرض کنید، فایلی به صورت زیر وجود دارد. اگر بخواهیم دانش‌آموزی با معدل ۱۵ را اضافه کنیم و نظم فایل که به صورت صعودی مقادیر معدل مرتب است حفظ شود، باید این رکورد

۱- Sequential

۲- Direct

۳- Index

۴- اگرچه از نظر زمانی نسبت به روش ترتیبی سریع‌تر است اما مصرف حافظه در آن بیشتر خواهد بود.

بین رکورد اول و دوم قرار گیرد. یعنی رکوردی بین اول و دوم اضافه کنیم و اطلاعات دانش آموز جدید را در آن ثبت کنیم.

	سایر فیلدها ...	فیلد معدل		سایر فیلدها ...	فیلد معدل
رکورد اول		۱۴	درج رکوردی با معدل ۱۵ در محل منطقی →	رکورد اول	۱۴
رکورد دوم		۱۷		رکورد دوم	۱۵
رکورد سوم		۱۸		رکورد سوم	۱۷
رکورد چهارم		۲۰		رکورد چهارم	۱۸
				رکورد پنجم	۲۰

شکل ۲-۱- درج در محل منطقی (نظم صعودی فیلد معدل حفظ شده است)

۳-۳-۱- حذف رکوردها (Delete): در صورتی که به رکورد یا رکوردهایی نیاز نداشته باشیم، آن را از فایل مربوطه حذف می‌کنیم.

۳-۳-۲- به‌هنگام‌سازی رکوردها (Update): اصلاح مقادیر فیلد یا فیلدهایی از رکورد، در اصطلاح به‌هنگام‌سازی نام دارد. به‌عنوان مثال تغییر نمرهٔ یک درس دانش‌آموز به‌هنگام‌سازی محسوب می‌شود.

۳-۳-۳- تغییر ساختار (Restructure): اگر به هر دلیل در ساختار رکوردها تغییر ایجاد شود مثلاً اندازهٔ فیلد تغییر کند یا فیلد جدیدی اضافه شود، تغییر ساختار صورت می‌گیرد.

۴-۱- سیستم فایلینگ (FS)

همان‌طور که گفتیم سیستم فایلینگ، اولین نوع سیستم ذخیره و بازیابی اطلاعات است. در این سیستم، نرم‌افزارهای کاربردی، مدیریت ذخیره و بازیابی داده‌ها را نیز برعهده داشتند. در این سیستم برای کاربردهای مختلف، فایل‌های جداگانه‌ای طراحی و پیاده‌سازی می‌شوند. در سیستم فایلینگ انجام عملیات به‌وسیلهٔ برنامه‌سازی و پیاده‌سازی الگوریتم‌ها انجام می‌شود. زمانی که سیستم بزرگ و پیچیده می‌شود، هر تغییر جزئی در خواسته‌ها، منجر به کدنویسی می‌گردد.

به علت وجود فایل‌های مختلف، کنترل رعایت استاندارد در زمان پیاده‌سازی سخت است. به‌عنوان مثال فیلد نام‌خانوادگی در دو فایل ممکن است اندازه‌های مختلف داشته باشد. در سیستم فایلینگ، ممکن است «ناسازگاری داده‌ها» به‌وجود آید. در نظر بگیرید شخصی در زیر سیستم اول

خود را «علی علوی تهرانی» و در زیر سیستم دوم «علی علوی» معرفی کند. بدین ترتیب مثلاً در مقایسهٔ دو زیرسیستم، این دو مقدار یکسان نیستند و این به معنی ناسازگاری داده است. هم‌چنین احتمال بروز افزونگی^۱ نیز وجود دارد. یعنی محتویات بعضی از فیلدها به‌صورت متعدد و تکراری ذخیره شوند. به‌طور مثال، مشخصات سکونت یک فرد هم در سیستم فایلینگ اول و هم در سیستم دوم ثبت گردد. نتیجهٔ افزونگی، اتلاف حافظه و مشکلات در انجام عملیات مبنایی است.



نکته
افزونگی، می‌تواند ناشی از طراحی نامناسب باشد که در سایر سیستم‌های ذخیره و بازیابی نظیر بانک اطلاعاتی نیز وجود دارد و منحصر به سیستم فایلینگ نیست.

برنامه‌سازی در این روش معمولاً با استفاده از یک زبان سطح بالا صورت می‌گیرد و معمولاً فایل‌های سیستم قابل استفاده به‌وسیلهٔ سایر سیستم‌های فایلینگ نیستند. سیستم فایلینگ هنوز کاربرد دارد و به‌طور کلی معایب روش فایلینگ را می‌توان به‌صورت زیر خلاصه کرد:

- ۱- احتمال بروز افزونگی (تکرار در ذخیره‌سازی)
- ۲- حجم زیاد کدنویسی
- ۳- احتمال بالای ناسازگاری داده‌ها
- ۴- سخت بودن اعمال استاندارد در سیستم‌ها

۵-۱- پایگاه داده‌ها

سیستم مدیریت پایگاه داده، یکی از سیستم‌های توسعه یافتهٔ ذخیره و بازیابی اطلاعات است. اگرچه این مفهوم بسیار شناخته شده و رایج است، اما واقعیت آن است که تعریف کاملاً واحدی در متون معتبر برای آن ذکر نشده است.

در نگرش بانک اطلاعاتی، طراحی به‌صورت یکپارچه انجام می‌گیرد. از آنجایی که یک محیط عملیاتی می‌تواند دارای زیرمحیط‌های مختلفی باشد، این نگرش باعث می‌شود برخی معایب روش فایلینگ نظیر ناسازگاری داده‌ها و اعمال نشدن استانداردها رفع گردد، مثلاً برای فیلد نام در تمام محیط، مقدار

یکسانی از حافظه اختصاص می‌یابد. حال آن‌که در روش فایلینگ به علت مجزا بودن فایل‌ها به سختی و با صرف هزینه بالا می‌توان نگرش بانک اطلاعاتی داشت.

یکپارچه‌سازی و مدیریت متمرکز باعث جلوگیری از تکرار در ذخیره‌سازی (افزونگی) خواهد شد و در نتیجه از حافظه، استفاده بهتری صورت می‌گیرد. در مجموع می‌توان پایگاه داده را به صورت زیر تعریف کرد:

«مجموعه‌ای از داده‌ها به صورت یکپارچه با حداقل افزونگی، تحت کنترل یک سیستم متمرکز و در چارچوب یک مدل داده‌ای»

با ترکیب دست‌آوردهای شبکه‌ای، پایگاه داده می‌تواند به وسیله چند کاربر و همزمان استفاده شود.

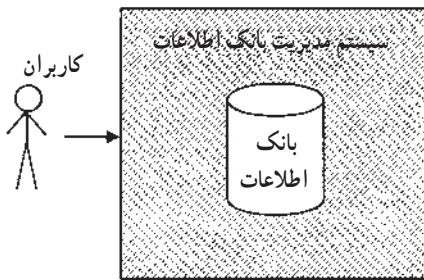


تعریف‌های مختلف پایگاه داده را بررسی کنید.

۶-۱- سیستم مدیریت پایگاه داده

اصلی‌ترین تفاوت روش پایگاه داده باروش‌های قبلی مثل سیستم فایلینگ، وجود حصارى به نام «سیستم مدیریت بانک اطلاعاتی» است. هرگونه دستیابی به داده‌ها باید از طریق آن انجام شود.

در DBMS، بانک اطلاعاتی و تمامی فایل‌های آن فقط و فقط در اختیار این نرم‌افزار قدرتمند هستند. کاربران درخواست خود را به این نرم‌افزار ارسال می‌کنند و در صورت تأیید، کار خواسته شده انجام می‌گیرد.



۱-۶-۱ معماری پایگاه داده: سؤال این است که چگونه می‌توان در یک سیستم بزرگ و پیچیده بانک اطلاعات، بین کاربران و داده‌های بسیار زیاد ذخیره شده روی رسانه ارتباط برقرار کرد و

چگونه سیستم مدیریت پایگاه داده درخواست‌های کاربران برای عملیات پایگاه داده‌ای (مانند: بازیافت، درج، به‌هنگام‌سازی، حذف و غیره) را اداره و کنترل می‌کند.

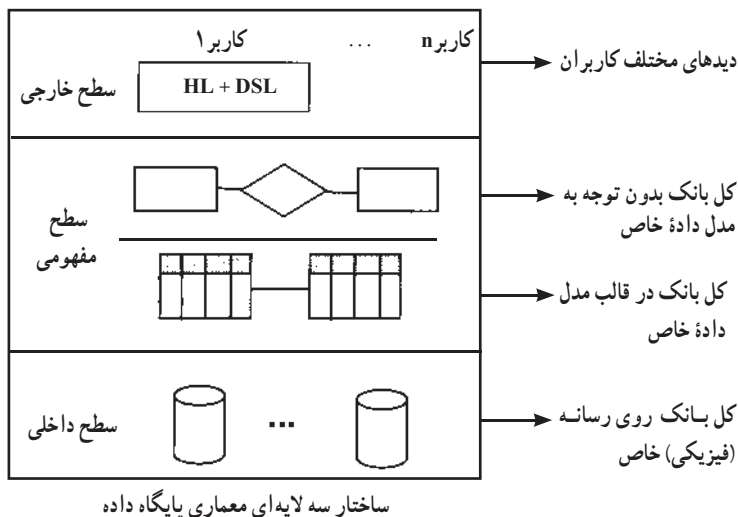
پاسخ این سؤال در معماری سه لایه‌ای است که به وسیله کمیته ANSI/SPARC^۱ عرضه شده است. در ایجاد این معماری سطوح به گونه‌ای از هم مجزا شده‌اند که برای رسیدن به هدف با هم به خوبی مراد و تعامل کنند و هم به اندازه کافی مجزا و مستقل باشند که تغییر در یک سطح به سطوح دیگر انتقال پیدا نکند. این سه لایه عبارت است از:

۱- سطح خارجی^۲ ۲- سطح مفهومی^۳ ۳- سطح داخلی^۴

● **سطح خارجی:** همان دید کاربری است. بالاترین سطح انتزاع است. به این ترتیب که فقط بخشی از پایگاه داده که به مسایل کاربر و یا برنامه کاربردی مربوط است را شامل می‌شود.

● **سطح مفهومی:** ساختار پایگاه داده است. تمام موجودیت‌ها^۵ و ارتباطات بین آن‌ها را شامل می‌شود.

● **سطح داخلی:** جزئیات ذخیره‌سازی است. پایین‌ترین سطح انتزاع است و به روش‌های فیزیکی ذخیره و بازیابی نزدیک است. چگونگی ذخیره‌سازی را نشان می‌دهد. ساختارهای داده را توضیح می‌دهد و هم چنین روش‌های دسترسی را نشان می‌دهد. سطح داخلی، سطح فایلینگ است.



۱- ANSI Standard Planning and Requirements Committee

۲- External Level

۳- Conceptual Level

۴- Internal Level

۵- Entity, مانند دانش آموز در محیط عملیاتی مدرسه



نکته
HL^۱: زبان برنامه‌نویسی غیربانکی است. مثل Visual Basic و...
DSL^۲: زبان برنامه‌نویسی بانکی یا به عبارتی زبان فرعی داده‌ای است. مثل SQL و...

۲-۶-۱- مدل داده‌ای: در تعریف پایگاه داده از «مدل داده‌ای» نام برده‌ایم. سیستم مدیریت بانک اطلاعاتی پنهان از دید کاربر، هرگونه دستیابی به داده‌ها را انجام می‌دهد. به این منظور در معماری پایگاه، مدل داده‌ای استفاده شده است. مدل‌های داده‌ای براساس ساختمان داده‌های مختلفی در بانک‌های اطلاعاتی استفاده می‌شوند که سه نوع متداول آن عبارت است از:

۱- رابطه‌ای Relational Data Structure

۲- سلسله مراتبی Hierarchical Data Structure^۲

۳- شبکه‌ای Network Data Structure^۳

در مورد مدل‌های داده‌ای دیگر پژوهش کنید.



همان‌طور که دیدیم، سیستم مدیریت پایگاه داده‌ها، نرم‌افزاری است پیچیده که واسط بین کاربران و محیط فیزیکی ذخیره داده‌ها است. از جمله سیستم‌های مدیریت پایگاه اطلاعات می‌توان از: Oracle، My Sql، SQL Server، Access و... نام برد.

برخی از این سیستم‌های نرم‌افزاری از دیگری پیچیده‌تر، کامل‌تر و بنابراین گران‌تر هستند و برای ذخیره‌سازی داده‌های بسیار زیاد استفاده می‌شوند. به برخی از آن‌ها نیز غیر از مدیریت داده‌ها، بخش‌های دیگری اضافه شده است که واسط کاربری را آرایه می‌کنند. مثلاً در Oracle، بخش Development 2000 جدای از بخش مدیریت بانک عرضه می‌شود که طراح به کمک آن می‌تواند فرم‌ها و گزارش‌ها را ایجاد کند و آسان‌تر با محیط بانک ارتباط برقرار کند. در نرم‌افزار اکسس هم بخش‌هایی چون فرم‌ها، گزارش‌ها، ماژول‌ها و... برای این منظور ایجاد شده است که در بخش‌های بعدی بیشتر با آن آشنا می‌شویم.

۱- Host Language

۲- Data Sub Language

۳ و ۴- درباره مدل‌های سلسله مراتبی و شبکه‌ای در مقاطع بالاتر بیشتر خواهید آموخت.

۱-۷-۱- مفاهیم پایه پایگاه داده رابطه‌ای

۱-۷-۱- محیط عملیاتی^۱ و تعریف موجودیت: محیط عملیاتی، محیطی است که می‌خواهیم یک سیستم ذخیره و بازیابی برای آن ایجاد کنیم. مانند: مدرسه یا دانشگاه، بانک، تعمیرگاه، فروشگاه و...

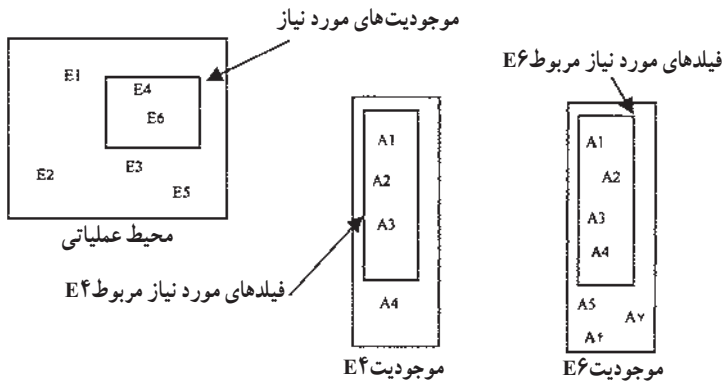
در هر محیط عملیاتی تعدادی موجودیت وجود دارد. موجودیت، هر مفهوم یا شیء در محیط عملیاتی است. مثلاً دانش‌آموز، معلم، کارمند مدرسه، کلاس و درس موجودیت‌های محیط عملیاتی مدرسه هستند. تمام موجودیت‌های یک محیط عملیاتی مهم نیستند بلکه موجودیت‌هایی برای ما مهم هستند که می‌خواهیم در مورد آن‌ها اطلاع داشته باشیم. به‌عنوان یک مثال ساده در محیط عملیاتی مدرسه موجودیت‌های دانش‌آموز، معلم و درس برای سیستم فرضی ثبت‌نام مهم و قابل توجه هستند. در هر محیط عملیاتی مجموعه‌ای از موجودیت‌های مناسب و مرتبط براساس مرز و محدوده سیستم مورد طراحی، انتخاب می‌شوند.

در محیط‌های عملیاتی بانک، خانواده و باشگاه ورزشی چه موجودیت‌هایی قابل تشخیص است؟

تمرین ۱-۱:

پرسش: در چه مواقعی از فیلدهای خود موجودیت برای تعریف کلید استفاده نمی‌شود؟

۱-۷-۲- ویژگی‌های هر موجودیت: صفات و ویژگی‌های هر موجودیت در واقع همان فیلدها هستند. در مورد انتخاب فیلدهای یک موجودیت نیز مانند انتخاب موجودیت‌های محیط عملیاتی فقط صفات مورد نیاز و مرتبط به‌عنوان فیلد در نظر گرفته می‌شوند و تمام صفات یک موجودیت در نظر گرفته نمی‌شوند. به‌عنوان مثال، در محیط عملیاتی مدرسه در مورد اطلاعات مربوط به دانش‌آموز ویژگی‌هایی نظیر قد، وزن، رنگ چشم در نظر گرفته نمی‌شود چرا که در محیط عملیاتی مورد نظر به آن‌ها نیاز نیست، ولی ممکن است در محیط عملیاتی دیگر مناسب و مفید باشند.



شکل ۱-۱- زیر مجموعه موجودیت‌ها و فیلدهای انتخاب شده برحسب نیاز



در نظر گرفتن محدوده محیط عملیاتی قبل از تشخیص موجودیت و ویژگی‌ها چه فایده‌هایی دارد؟

در سه محیط عملیاتی تمرین ۱-۱ موجودیت‌های مرتبط به همراه فیلدهای مورد نیاز را شناسایی کنید.

تمرین ۱-۲:



محیط عملیاتی مدرسه را در نظر بگیرید. موجودیت‌ها و ویژگی‌های هر موجودیت را تعریف کنید.

موجودیت‌ها: دانش آموز، معلم و درس است.

ویژگی‌های موجودیت دانش آموز: شماره دانش آموزی، نام، نام خانوادگی، تلفن، آدرس.

ویژگی‌های موجودیت معلم: نام، نام خانوادگی، تلفن، آدرس.

ویژگی‌های موجودیت درس: شماره درس، نام درس، تعداد واحد، ساعت، کلاس و نام کلاس.

* دقت کنید نام معلم ویژگی درس نیست زیرا تداخل اطلاعاتی رخ می‌دهد، بعدها خواهیم دید چرا مشخصه معلم به درس اضافه می‌شود.

مثال ۱-۱

۳-۷-۱- ارتباط^۱: پیش از این عنوان شد که در محیط عملیاتی، موجودیت‌هایی وجود دارند. معمولاً بین موجودیت‌ها ارتباط یا ارتباط‌هایی وجود دارد که قابل شناسایی است. در محیط عملیاتی مدرسه، بین موجودیت‌های درس، دانش‌آموز و معلم ارتباط‌های زیر قابل تعریف است:

- ۱- معلم درس ارائه می‌کند.
 - ۲- دانش‌آموز هر ترم چندین درس ثبت‌نام می‌کند.
 - ۳- دانش‌آموز برای هر یک از درس‌های ثبت‌نام شده نمره می‌گیرد.
همان‌طور که ملاحظه می‌شود:
- در هر ارتباط موجودیت‌هایی شرکت دارند.
 - هر ارتباط دارای یک مفهوم یا عملکرد است. مثلاً در ارتباط اول «ارائه کردن» عملکرد می‌باشد.
 - هر ارتباط دارای «ماهیتی» خاص است. مثلاً در ارتباط دوم ماهیت ارتباط می‌تواند یک به چند باشد. یعنی یک دانش‌آموز چند درس اخذ می‌کند.
 - برای ایجاد ارتباط، از کلید خارجی استفاده می‌شود. کلید خارجی فیلدی است مشترک بین دو رابطه که در یک رابطه اصلی است و در رابطه‌ای که کلید خارجی محسوب می‌شود، به تنهایی کلید اصلی نیست.
 - ارتباط مانند موجودیت می‌تواند ویژگی‌هایی داشته باشد.



آیا بین سه موجودیت نیز می‌توان ارتباط برقرار کرد؟ مثال بزنید.

۴-۷-۱- انواع ماهیت در ارتباط^۲: تناظر بین موجودیت‌ها، ماهیت ارتباط است و به یکی از سه نوع زیر وجود دارد:

- ۱- یک به یک ۱: ۱
- ۲- یک به چند ۱: N
- ۳- چند به چند M: N

در ارتباط یک به یک، یک نمونه از موجودیت اول فقط با یک نمونه از موجودیت دیگر ارتباط دارد. مثلاً، هر مدرسه یک مدیر دارد. در ارتباط یک به چند، یک نمونه موجودیت با چندین نمونه از موجودیت دیگر مرتبط است. مثلاً یک معلم چند درس ارائه می‌کند. در حالت سوم، یک نمونه از موجودیت اول با چندین نمونه از موجودیت دوم مرتبط است و برعکس. مثلاً یک دانش‌آموز چند درس اخذ می‌کند و یک درس به وسیله چند دانش‌آموز اخذ می‌شود.

مثالی از حالت چند به چند ارائه کنید.

تمرین ۳-۱:



حالتی را مثال بزنید که موجودیت با خودش مرتبط باشد.

در محیط عملیاتی مدرسه، ماهیت ارتباطها به صورت زیر است:

۱- یک معلم چند درس ارائه می‌کند و هر درس به وسیله یک معلم ارائه می‌شود.

(۱: n)

۲- دانش‌آموز (در هر ترم) چندین درس ثبت‌نام می‌کند و یک درس به وسیله چندین

دانش‌آموز انتخاب می‌شود. (m: n)

۳- یک دانش‌آموز برای هر درس ثبت‌نام شده، یک نمره می‌گیرد. یک نمره به

یک درس داده می‌شود. (۱: n)

مثال ۱-۲

۵-۷-۱- نمودار ارتباط - موجودیت/ER: برای نشان دادن ارتباط بین موجودیتها

و بیان عملکرد ارتباط، از نمودار ER استفاده می‌شود. به عبارت دیگر، یک مدل‌سازی معنایی از داده‌هاست.

اجزای نمودار ER عبارت است از:

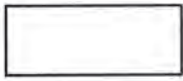
۱- موجودیتها

۲- عملکرد ارتباط

۳- ماهیت ارتباط

۴- ویژگی‌های موجودیت

برای ترسیم نمودار، از شکل‌هایی به صورت زیر استفاده می‌شود^۱:



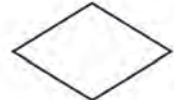
موجودیت



ویژگی موجودیت



ارتباط بین موجودیت‌ها



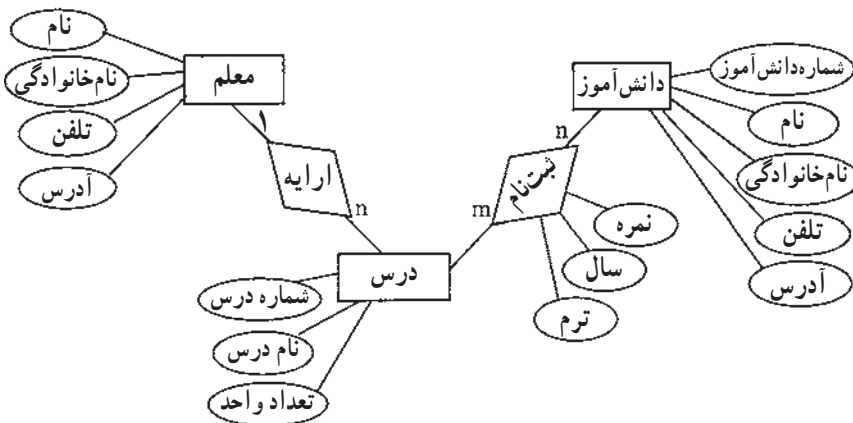
عملکرد

موجودیت‌ها از جنس «اسم» هستند مانند دانش‌آموز و درس. عملکرد ارتباط از جنس «فعل» یا «مصدر» است مانند اخذ کردن. ماهیت ارتباط تناظر بین موجودیت‌ها را نشان می‌دهد، ویژگی‌های موجودیت همان صفات قابل شناسایی هر موجودیت است.



ماهیت ارتباط روی خطوط متصل بین موجودیت‌ها در دو طرف شکل لوزی ذکر می‌گردد.

نمودار ER محیط عملیاتی مدرسه به صورت زیر است:



مثال ۱-۳

۱- شکل‌ها به صورت قراردادی است و در منابع مختلف متفاوت است.

دقت: یک عملکرد خود می تواند مشخصه‌هایی داشته باشد. در عملکرد ثبت نام مشخصه‌های سال، ترم و نمره درس وجود دارد.

۶-۷-۱- رابطه / جدول : مدل رابطه‌ای بر مبنای یک مفهوم ریاضی به نام «رابطه»^۱ تعریف شده است. استحکام نظریه مجموعه‌ها و سهولت پیاده‌سازی آن باعث رواج این مدل داده‌ای شده است. بر این اساس هر بانک اطلاعاتی از تعدادی رابطه یا همان جدول تشکیل می‌شود.

هر جدول می‌تواند نشان‌دهنده یک موجودیت و یا ارتباط بین موجودیت‌ها باشد. ER مثال ۱-۱ را دوباره ملاحظه نمایید. طبق مدل داده‌ای رابطه‌ای جدول‌های زیر را می‌توان در نظر گرفت :

- جدول دانش‌آموز (Student)
- جدول درس (Course)
- جدول معلم (Teacher)

Course		
شماره درس	نام درس	تعداد واحد

Student				
شماره دانش‌آموزی	نام	نام خانوادگی	تلفن	نشانی

Teacher			
نام	نام خانوادگی	تلفن	آدرس

هر جدول با نام خود و مجموعه نام ستون‌هایش مشخص می‌شود. دقت کنید بهتر است در نرم‌افزار اکسس نام جدول و نام ستون‌ها به صورت لاتین باشد ولی در این جا فارسی آورده شده است.

۱- رابطه، زیر مجموعه‌ای از حاصل ضرب دکارتی مجموعه‌ها است. مفاهیم مرتبط در این نظریه به ترتیب عضو، مجموعه حاصل ضرب دکارتی، رابطه و تابع است.

محاسن مدل داده‌ای رابطه‌ای :

- ۱- از نظر نمایش، ساده و قابل فهم است و تنها از یک مفهوم اصلی به نام رابطه یا جدول استفاده می‌کند.
- ۲- انواع ماهیت‌های ارتباط را پشتیبانی می‌کند.
- ۳- مبنای ریاضی دارد.

هنگامی که طراح، موجودیت‌های مورد نیاز و ویژگی‌های آن‌ها را به همراه ارتباط بین موجودیت‌ها شناسایی کرد، کافی است نام و نوع آن‌ها را در حالت جدولی به پایگاه داده معرفی نماید.

۷-۷-۱- تبدیل ER به جدول

- هر موجودیت مستقل به یک جدول تبدیل می‌شود.
- صفات موجودیت‌ها به فیلدهای جدول مربوطه تبدیل می‌شوند.
- موجودیت‌هایی که ارتباط ۱: ۱ دارند به یک جدول تبدیل می‌شوند. [اگرچه ممکن است داخل اطلاعاتی رخ دهد]. در مواردی نیز مانند $n: 1$ یکی از جدول‌ها به دیگری منتقل می‌شود.
- در ارتباط $n: 1$ ، کلید اصلی جدول طرف ۱ به جدول طرف n اضافه می‌شود.
- ارتباط $n: m$ ، تبدیل به یک جدول می‌شود که کلید اصلی دو موجودیت را شامل می‌شود.

۸-۱- قدم‌های طراحی بانک اطلاعاتی

در طراحی هر سیستم اطلاعاتی، صرف‌نظر از روش و متدولوژی طراحی باید ابتدا محیط سیستم شناخته و مطالعه شود، سپس نیازسنجی صورت می‌گیرد یعنی تعیین نیازهای اطلاعاتی - پردازشی، تشخیص محدودیت‌ها و قواعد حاکم بر محیط.

روش‌های مختلفی برای طراحی وجود دارد. با بیان ساده می‌توان گفت که در طراحی بانک داده مدل‌سازی معنایی داده صورت می‌گیرد یعنی ابتدا طراحی مفهومی انجام می‌شود و نمودار ER رسم می‌گردد و در طراحی سیستم اطلاعاتی، معمولاً ابتدا تحلیل فرآیندی صورت می‌گیرد و پردازش‌ها و گردش اطلاعات شناسایی می‌شوند. با مشخص شدن پردازش‌ها، داده‌هایی که از طریق فرم‌ها رد و بدل می‌شوند و تغییراتی که روی آن‌ها انجام می‌شود، مشخص می‌گردند و به این ترتیب ذخایر داده‌ای شکل می‌گیرند و برنامه کاربردی طراحی می‌شود یعنی واسط‌های کاربری مثل فرم‌ها و گزارش‌ها و کنترل‌های

مورد نیاز تعیین می‌شود. در این روش‌ها، پس از انجام طراحی تفصیلی، جدول‌ها، فیلدها و کلید اصلی آن‌ها از نمودار ER و یا ذخایر داده‌ای تعیین می‌شوند. روش تبدیل موجودیت‌ها و رابطه‌ها (نمودار ER) را به جدول‌ها، فیلدها و کلید قبلاً توضیح داده‌ایم.

برای مطالعه بیشتر

در هر روش و متدولوژی طراحی، مراحل انجام طراحی، اصول و مفروضات مورد استفاده، ابزار و نکات کاربردی و محصولات و نتایج حاصل در هر مرحله و همچنین ارتباط بین مراحل، چگونگی عبور از هر مرحله، روش کنترل مستندات حاصل در هر مرحله، و در برخی متدولوژی‌ها ممیزی، کنترل و حساب‌رسی کیفیت مراحل و محصولات هر مرحله مشخص است. به‌طور کلی سه روش وجود دارد:

فرایندگرا (Process Oriented)، داده‌گرا (Data Oriented) و شیء‌گرا (Object Oriented). روش فرایندگرا مانند Urdon, Jackson, SSADM و غیره. داده‌گرا مانند روش Information Engineering (IE) و روش شیء‌گرا مانند RUP (UML).

کنجکاو

یک روش و متدولوژی طراحی نام برده و خصوصیات آن را به‌طور خلاصه بیان کنید.

بحث شناخت سیستم و طراحی فرم‌ها و گزارش‌ها به مبحث تجزیه و تحلیل سیستم‌های اطلاعاتی مرتبط است که خود محدوده وسیعی دارد و از بحث ما خارج است.

با این سرآغاز طراحی یک بانک داده را به قدم‌های زیر ساده می‌کنیم:

- ۱- ابتدا هدف از طراحی پایگاه داده را مشخص کنید.

مشخص کنید چه نوع اطلاعاتی باید از بانک دریافت شود. با افرادی که از بانک استفاده خواهند کرد صحبت کنید و در مورد فرم‌ها و گزارش‌های مورد نیاز، اطلاعات جمع‌آوری کنید.

- ۲- نمودار ER را طراحی کنید.

موجودیت‌ها، ویژگی‌ها و ارتباط بین آن‌ها را شناسایی کنید و نمودار ارتباط - موجودیت‌ها

(ER) را ترسیم کنید.

برای هر موجودیت کلید اصلی را تعیین یا ایجاد کنید.



بمخاطر بسپارید که یک جدول نباید اطلاعات افزونه داشته باشد یعنی هر جدول باید اطلاعات یک موضوع را نگهداری کند و از اختلاط اطلاعاتی پرهیز شود.

۳- جدول‌ها و فیلدهای آن‌ها را مشخص کنید.

نمودار ER ترسیم شده را به جدول‌ها، فیلدهای مربوطه تبدیل کنید. بعدها خواهیم دید که جدول ایجاد شده از نظر سطوح نرمال بررسی شده و ممکن است خود به جدول‌های جدید تجزیه شوند.

۴- فرم‌های مورد نیاز را تعریف کنید.

فرم‌های ورود، ویرایش و حذف داده را رسم کنید و از وجود فیلدهای فرم در جدول‌های بانک مطمئن شوید.

۵- گزارش‌های مورد نیاز را تعریف کنید.

گزارش‌های مورد نیاز کاربران بانک را معین کرده، قالب نمایش را تعیین کنید. وجود داده‌های لازم و مورد نیاز نمایش گزارش در جدول‌های بانک را بررسی نمایید.

۶- طراحی خود را اصلاح و بازنگری کنید.

یافته‌های خود را با افرادی که از بانک استفاده می‌کنند مطرح کنید و بار دیگر نیازها را بررسی و طراحی خود را اصلاح و بازنگری نمایید.



تفاوت‌های طرح خود را با طراحی مثال بعد بررسی و تحلیل کنید.



دقت کنید که ممکن است طراحی شما با جدول‌های زیر متفاوت باشد. این مسأله نشان‌دهنده غلط بودن طراحی نیست. بلکه برحسب مفروضات، سطح نرمال بودن جدول‌ها و محدوده سیستم کاملاً طبیعی است.

۱- با نرمال‌سازی در فصل ۷ آشنا خواهید شد.

محیط عملیاتی مدرسه را طراحی کنید.

نمودار ER رسم شده است.

۱- نمودار ER به جدول‌های زیر تبدیل می‌شود:

شکل [] نشان‌دهنده کلید اصلی و شکل () نشان‌دهنده کلید خارجی اضافه شده از جدول دارای ارتباط است.

جدول معلم Tbl-Teacher

Teacher Code	Teacher FName	Teacher LName	Teacher Tel	TeacherAddress
[کد معلم]	نام	نام خانوادگی	تلفن	آدرس

جدول دانش‌آموز Tbl-Student

Student Code	Student FName	Student LName	Student Tel	StudentAddress	Year
[شماره دانش‌آموزی]	نام	نام خانوادگی	تلفن	آدرس	سال

* در جدول درس، فیلد کد معلم را که کلید اصلی در جدول معلم است به عنوان کلید خارجی اضافه می‌کنیم.

جدول درس Tbl-Course

Course Code	Course FName	Course Unit	Teacher Code
[شماره درس]	نام درس	تعداد واحد	(کد معلم)

* ارتباط ۱: n بین درس و معلم، کلید جدول معلم (کد معلم) به جدول درس اضافه می‌شود. دقت کنید که کد معلم در جدول درس بخشی از کلید اصلی نیست و کلید خارجی به حساب می‌آید.

جدول ثبت نام Tbl-Register

Year	Term	Course Code	Student Code	CourseGrade
سال	ترم	شماره درس	شماره دانش آموز	نمره درس

* ارتباط $n:m$ بین دانش آموز و درس به جدول ثبت نام تبدیل می شود و کلید اصلی جدول درس (شماره درس) و کلید اصلی جدول دانش آموز (شماره دانش آموز) به آن اضافه می شود.

چون ارتباط $n:m$ مفروض خود دارای ویژگی هایی است پس این دو کلید به تنهایی کلید اصلی آن جدول نیستند و ترکیب آن دو با فیلد سال و ترم جدول کلید اصلی شده است.

با توجه به اصول و مفروضات طراحی که در بخش نرمال سازی خواهیم دید، جدول ثبت نام به دو جدول زیر تبدیل می شود:

جدول ثبت نام Tbl-Register

Register Year	Term	Student Code	Register Code
سال	ترم	شماره دانش آموز	سریال ثبت نام

* ما شماره سریال ثبت نام را به عنوان کلید اصلی اضافه می کنیم. می توانستیم سال و ترم را هم کلید بگیریم ولی بهتر است از کلید ترکیبی به علت پیچیدگی هایی که ایجاد می کنند استفاده نکنیم.

جدول دروس ثبت نام شده Tbl-Register

Register Code	Course Code	Grade
سریال ثبت نام	شماره درس	نمره درس

* دقت کنید که در جدول دروس ثبت نام شده شماره درس بخشی از کلید اصلی است.

۲- فرم‌های مورد نیاز :

فرم تعریف معلم

کد معلم :
 نام :
 نام خانوادگی :
 تلفن :
 آدرس :

فرم تعریف دانش‌آموزان

شماره دانش‌آموز :
 نام :
 نام خانوادگی :
 تلفن :
 آدرس :
 سال :

فرم تعریف درس

شماره درس :
 نام درس :
 تعداد واحد :
 کد معلم :

فرم ثبت نام

سال : ترم : نام دانش‌آموز :

شماره درس	نام درس	تعداد واحد

فرم ورود نمره

سال : ترم : نام دانش‌آموز :

شماره درس	نام درس	تعداد واحد	نمره نهایی

۳- گزارش‌های مورد نیاز :

گزارش ثبت‌نام

سال : _____
ترم : _____
نام دانش‌آموز : _____ شماره دانش‌آموز : _____
دروس ثبت‌نام شده

شماره درس	نام درس	تعداد واحد	نام معلم

گزارش لیست معلمان مدرسه

کد معلم	نام معلم	تلفن	آدرس

گزارش لیست حضور و غیاب

سال : _____
ترم : _____
نام معلم : _____ درس : _____
تعداد واحد : _____ ساعت تشکیل کلاس : _____

شماره دانش‌آموز	نام دانش‌آموز	حضور	غیاب

گزارش کارنامه دانش‌آموزان

نام و نام خانوادگی : _____
تاریخ گزارش : _____
رشته : _____ سال : _____ ترم : _____

شماره درس	نام درس	تعداد واحد	نمره

جمع واحد : _____
معدل : _____

خلاصه فصل

سیستم ذخیره و بازیابی از مفهوم ساده فایلینگ توسعه پیدا کرده است. فیلد کوچک‌ترین واحد ذخیره داده است. رکورد مجموعه فیلدهای مرتبط است و مجموعه رکوردها فایل را تشکیل می‌دهند. این مفاهیم پایه‌ای در هر سیستم اطلاعاتی دیگر نیز مطرح‌اند. فیلد یا زیرمجموعه‌ای از فیلدها که باعث یکتایی رکورد شوند را کلید گویند. در سیستم‌های ذخیره و بازیابی، عملیات مشخصی روی رکوردها انجام می‌گیرد که عبارت است از بازیابی، درج، حذف، به‌هنگام‌سازی و تغییر ساختار. این عملیات را در محیط Access بیشتر بررسی می‌کنیم.

برای جلوگیری از درگیر شدن کاربر با مسایل برنامه‌نویسی و انجام سریع‌تر و دقیق‌تر تغییرات و در مجموع برای برقرار کردن ارتباط بین حجم بالای داده‌های ذخیره شده روی رسانه و کاربران معماری سه لایه‌ای مطرح شده است: سطح خارجی، سطح مفهومی، سطح داخلی.

نرم‌افزاری که این ارتباط را برقرار، مدیریت و اداره می‌کند، سیستم مدیریت بانک اطلاعاتی نام دارد که برای مدیریت دستیابی به داده‌ها و پنهان کردن آن از دید کاربران از یک مدل مفهومی به نام مدل داده‌ای استفاده می‌کند. مدل رابطه‌ای از جمله مدل‌های داده‌ای مهم است که بر پایه مفهوم ریاضی مجموعه‌ها بنا شده است و بیشتر سیستم‌های مدیریت بانک اطلاعاتی رایج از آن پشتیبانی می‌کنند.

در این فصل مفاهیم پایه مدل داده رابطه‌ای مانند: رابطه/جدول، ارتباط، ماهیت ارتباط، نمودار ارتباط - موجودیت (ER) و روش تبدیل این نمودار به جدول‌های پایگاه داده مطرح شده است.

۱- دلایل متغیر شدن طول رکورد چیست؟

۲- معایب روش فایلینگ چیست؟

۳- کدام یک از اجزای محیط ذخیره و بازیابی بیشتر اهمیت دارد؟

چرا؟

۴- هدف از ایجاد سیستم ذخیره و بازیابی چیست؟

۵- کلید را تعریف کنید و انواع آن را با ذکر مثال توضیح دهید.

۶- معماری پایگاه داده را توضیح دهید.

۷- مدل داده‌ای را تعریف کنید و توضیح دهید.

۸- ماهیت ارتباط را تعریف کنید و در یک محیط عملیاتی مثال

بزنید.

۹- مراحل طراحی پایگاه داده را توضیح دهید.

۱۰- لزوم طراحی قبل از پیاده‌سازی و شروع کار با نرم‌افزار Access

را بیان کنید و مزایا و معایب احتمالی آن را بررسی کنید.

فصل ۲

مقدمات بانک اطلاعاتی اکسس ۲۰۰۷

در فصل اول، با مفاهیم پایه‌ای بانک‌های اطلاعاتی و تعاریف آن آشنا شدیم. در این فصل می‌خواهیم با محیط بانک اطلاعاتی اکسس، ابزارها و اجزای آن آشنا شویم.

هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

- یک بانک نمونه را طراحی نماید.
- نرم‌افزار اکسس را اجرا کند.
- از ابزارها، پنجره‌ها و منوهای محیط کار استفاده کند.
- فایل پایگاه داده را ایجاد کند.
- اجزای بانک را تعریف و هدف کاربردی هر یک را توضیح دهد.

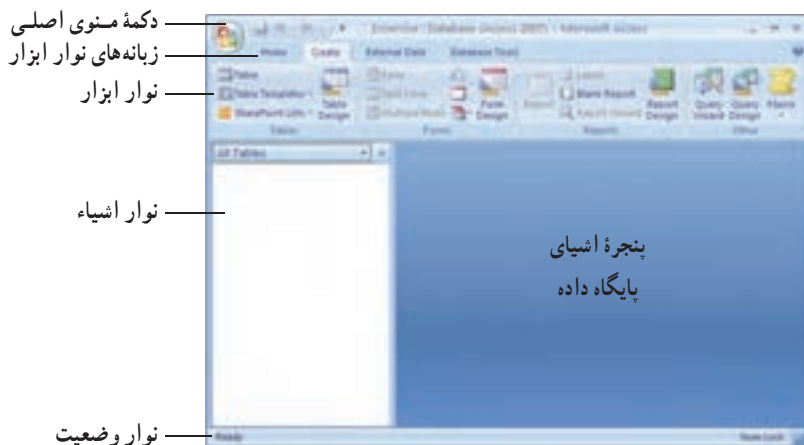
۲-۱- راه‌اندازی نرم‌افزار اکسس

برای استفاده از این نرم‌افزار مانند سایر برنامه‌ها، باید آن را در سیستم کامپیوتر مورد نظر نصب کرده باشیم. نرم‌افزار اکسس به همراه بسته نرم‌افزار Office 2007 نصب می‌شود که در بسته‌های نرم‌افزاری ۱ و ۲ با نصب آن آشنا شده‌اید.

۲-۲- محیط کار، منوها و ابزارهای محیط

این نرم‌افزار نیز مانند سایر برنامه‌های تحت ویندوز شامل منوها و انواع ابزارهای کار است که

در یک محیط کار قرار گرفته‌اند. تمامی قابلیت‌های این نرم‌افزار به وسیله این محیط کار قابل اجرا و دسترسی است (شکل ۲-۱).



شکل ۲-۱ - محیط کار سیستم



نوار منوی محیط کار را بررسی کنید. بسیاری از گزینه‌های آن مانند سایر نرم‌افزارهای تحت ویندوز است. هر یک را که می‌شناسید مختصر توضیح دهید.

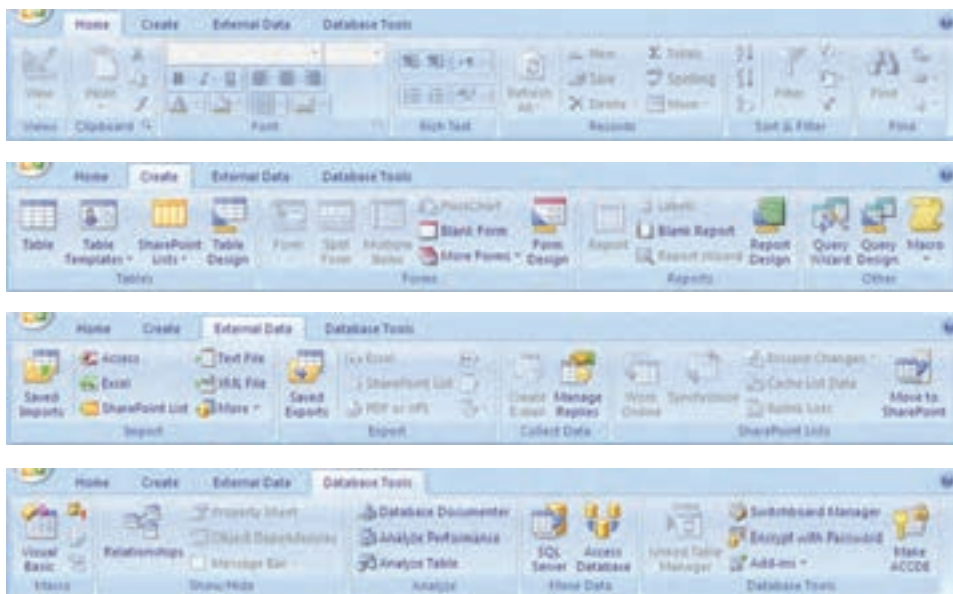
عملکرد نوار منو، نوار ابزار و نوار وضعیت مشابه آن‌ها در سایر نرم‌افزارهای Office 2007 است و از توضیح مجدد آن‌ها خودداری می‌کنیم^۱.

نوار اشیاء: شامل بخش‌های مختلف بانک اطلاعاتی است. جدول‌ها، فرم‌ها، پرس‌وجوها

و...

پنجره اشیاء پایگاه داده: با توجه به نوع انتخاب شده در نوار اشیاء، عملیات و کنترل‌های مربوط به آن در این قسمت نمایش داده می‌شود.

۱- برای یادآوری به کتاب‌های بسته‌های نرم‌افزاری ۱ و ۲ مراجعه شود.



شکل ۲-۲- نوار ابزار Access

بسته به نوع شیء انتخاب شده در نوار ابزار زبانه‌های متفاوتی با عملکرد مختلف نمایش داده می‌شود.

۳-۲- انواع اشیاء و اجزای پایگاه داده

پایگاه داده Access از اشیاء مختلفی تشکیل شده است، که عبارتند از جدول‌ها، پرس‌وجوها، فرم‌ها، گزارش‌ها، ماکروها، ماجول‌ها و صفحات دسترسی به داده. کارهای متفاوتی را با هر یک از این انواع انجام می‌دهیم، مانند: ذخیره‌سازی، نمایش و چاپ داده‌ها، نگهداری برنامه و... در ابتدا ممکن است از تعداد کمی از آن‌ها استفاده کنید. مثلاً از جدول^۱ شروع می‌کنید که برای ذخیره داده‌هاست، از فرم‌ها^۲ برای ویرایش داده‌های روی صفحه و از گزارش‌ها^۳ برای چاپ داده‌ها و همچنین از پرس‌وجوها^۴ برای انتخاب و ترکیب داده‌ها استفاده می‌کنید. بعد از آن، ممکن است از ماکروها^۵ و ماجول‌ها^۶ برای نگهداری برنامه‌ها استفاده کنید و یا در پروژه‌هایی که برای ویرایش و دسترسی داده‌ها از مرورگر وب^۷ استفاده می‌شود، از صفحات دسترسی به داده^۸ استفاده کنید.

۱- Table
۵- Macro

۲- Forms
۶- Module

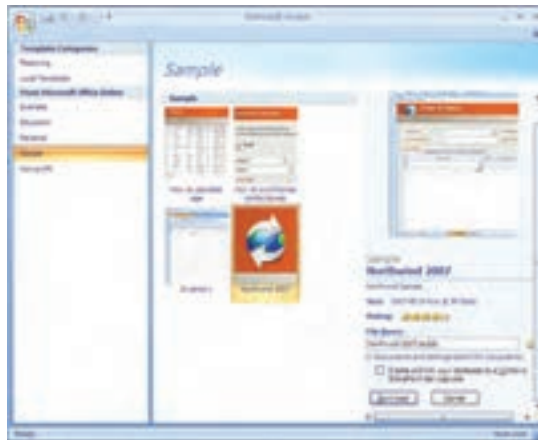
۳- Reports
۷- Web Browser

۴- Query
۸- Data Access Page

در این بخش این انواع اشیاء را با کمک بانک اطلاعاتی نمونه با نام Northwind.accdb که در Access وجود دارد، بررسی می‌کنیم. برای آن که بتوانیم از این بانک استفاده کنیم باید فایل مربوطه را در محیط Access باز کنیم. یکی از این روش‌ها عبارت است از:

برای استفاده از بانک اطلاعاتی نمونه می‌بایست با انتخاب گزینه New از منوی اصلی برنامه، در قسمت Sample از پنجره جدید باز شده، Northwind 2007 را انتخاب کرده و مطابق شکل ۲-۳ نام و محل ذخیره‌سازی آن را مشخص نمایید.

در صورتی که به اینترنت متصل باشید فایل بانک مورد نظر به کامپیوتر شما منتقل شده و از این به بعد می‌توانید از آن استفاده کنید.



شکل ۲-۳- دریافت فایل مثال Northwind

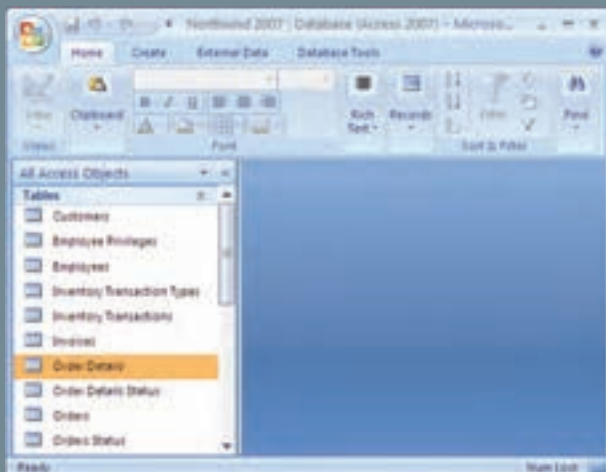
۱-۳-۲- جدول‌ها برای ذخیره داده‌ها: جدول، جایی است که می‌توان داده‌ها را ذخیره کرد.

یک جدول از رکوردهایی با قالب مشابه تشکیل شده است. یک بانک اطلاعاتی می‌تواند جدول‌های متعددی داشته باشد. مثلاً بانک اطلاعاتی یک فروشگاه کتاب می‌تواند شامل جدول کتاب: (عنوان، انتشارات، نویسنده، قیمت و سایر اطلاعات یک کتاب)، جدول فروشندگان: (نام شرکت، نشانی، تخفیف‌ها و سایر اطلاعات هر فروشنده) و جدول مشتریان فروشگاه: (نام، آدرس و دیگر اطلاعات یک فرد) تشکیل شده باشد.

وقتی جدول‌ها ایجاد شد و داده‌ها در آن ثبت شدند، می‌توانید داده‌ها را مرتب کنید، شرط‌های خاص بر آن اعمال و رکوردهای دارای آن شروط را مشاهده یا چاپ کنید، داده/داده‌های با شرط خاص را جستجو کرده، آن‌ها را پیدا کنید.

در بانک Northwind از نوار اشیاء، گزینه Tables را انتخاب کنید.
جدول‌های ایجاد شده مطابق شکل زیر است:

تمرین ۱-۲: ✓

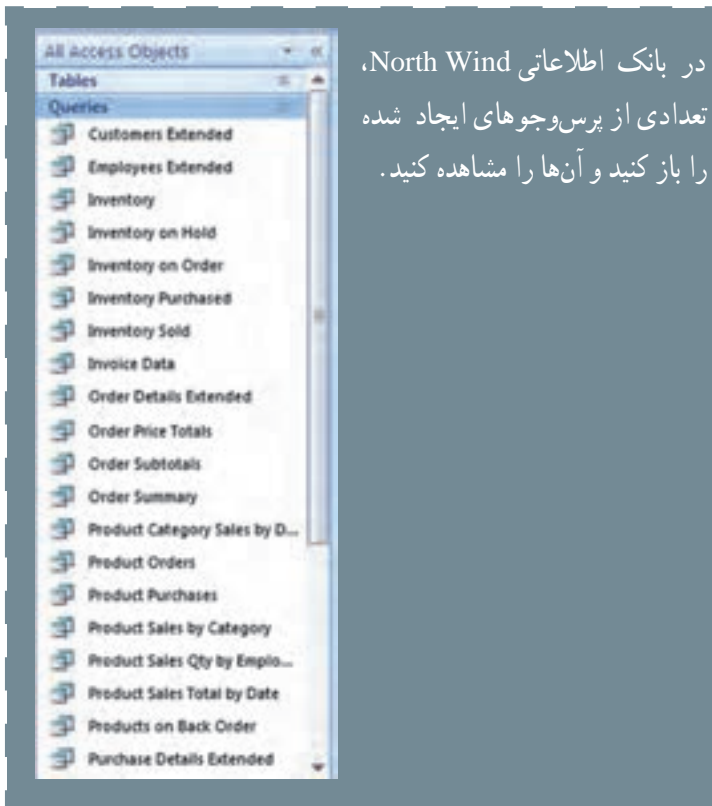


کنجکاو

روی هر جدول دو بار کلیک کنید. به نظر شما هر یک چه داده‌هایی را
نگه می‌دارد؟

در فصل اول در خصوص یک طراحی خوب صحبت کردیم و این‌که چگونه از موجودیت‌ها و ارتباطات آن‌ها به تعریف جدول‌ها می‌رسیم. در فصل‌های بعد با توجه به محیط عملیاتی مدرسه‌ای که طراحی کرده‌ایم، کار با نرم‌افزار را دنبال می‌کنیم.

۲-۳-۲ پرس‌وجوها برای انتخاب داده‌ها: یک پرس‌وجو برای جستجو، مشاهده و ویرایش (به‌هنگام‌سازی) داده‌های موجود در جدول به‌کار می‌رود. انواع مختلفی از پرس‌وجو وجود دارد. ساده‌ترین و معمول‌ترین نوع پرس‌وجو انتخاب داده‌ها (Select) از جدول است، مثلاً رکوردهایی که می‌خواهید در یک گزارش نمایش داده شوند. با توجه به مثال فروشگاه کتاب، می‌توان پرس‌وجویی ایجاد کرد که مشخصات تمام مشتریانی که در شهر مفروض X زندگی می‌کنند و شماره تلفنی از آن‌ها ندارید را نشان دهد.



در بانک اطلاعاتی North Wind، تعدادی از پرس وجوهای ایجاد شده را باز کنید و آن‌ها را مشاهده کنید.

تمرین ۲-۲:

می‌توان پرس وجوهایی ایجاد کرد که از ترکیب اطلاعات چند جدول استفاده کند. مثلاً در مثال فروشگاه کتاب که اطلاعات مشتریان در جدول مشتریان و اطلاعات تأمین‌کنندگان در جدول فروشندگان است، می‌توان پرس وجویی ایجاد کرد که اطلاعات افراد بانک را نمایش دهد، این پرس وجو، اطلاعات هر دو جدول را هم‌زمان نشان می‌دهد. علاوه بر آن می‌توان پرس وجویی ایجاد کرد که فیلدهای محاسباتی^۱ مانند تعداد کل، جمع کل، متوسط مقادیر و... را نمایش دهد. مثلاً تعداد کل مشتریانی که به فروشگاه مراجعه کرده‌اند. از انواع دیگر پرس وجوها، پرس وجوی عملیاتی^۲ است یعنی ایجاد رکوردهای جدید یا ویرایش و حذف رکوردها را انجام دهد. مثلاً پرس وجویی که رکوردهای یک جدول را انتخاب کند و در جدول دیگری کپی نماید، یا تغییری را بر تمام رکوردهای انتخاب شده اعمال کند، یا رکوردهایی را حذف نماید. مثلاً مشتریانی که طی دو سال خریدی نداشته‌اند را حذف نماید.

پرس وجوها ایزاری مناسب برای خارج کردن اطلاعات مفید از جدول‌ها هستند و ممکن است

^۱ _ Calculated Field

^۲ _ Action Query

در یک بانک اطلاعاتی، تعداد زیادی پرس و جو ایجاد کنید.

در بانک Northwind :

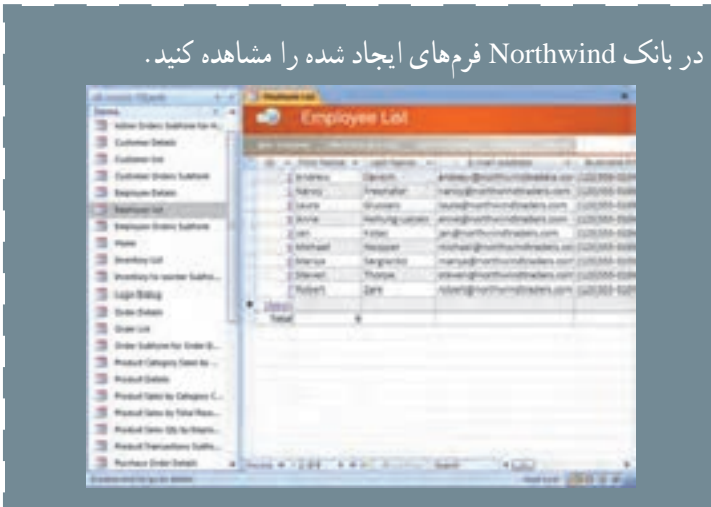
۱- هریک از این پرس و جوها چه کاری انجام می دهند؟ چه نوعی هستند (انتخاب داده یا عملیاتی)؟

تمرین ۲-۳

۲- محیط عملیاتی مدرسه را در نظر بگیرید. به نظر شما چه پرس و جوهای لازم است؟

۳-۲- فرم ها برای ویرایش و نمایش داده ها : دیدیم با پرس و جوی عملیاتی می توانیم اطلاعاتی را در جدول ها وارد کنیم. در محیط اکسس پس از باز کردن جدول می توانیم داده ها را به آن وارد کنیم. راه دیگر ورود اطلاعات به جدول ها، استفاده از فرم است. با کمک فرم می توان ورود اطلاعات را آسان تر کرد. داده های یک یا چند جدول در صفحه فرم نمایش داده می شود و می توان با امکانات فرم، اطلاعات نمایش یافته را ویرایش یا حذف کرد و یا رکورد جدیدی را ایجاد نمود. طرح های مختلفی برای نمایش اطلاعات در فرم وجود دارد که در زمان ایجاد یک فرم خواهیم دید. مثلاً قالب جدولی^۱ نمایش اطلاعات در فرم، شبیه نمای جدولی است. می توانیم با اشکال گرافیکی مثل مستطیل و خطوط، اطلاعات را روی فرم گروه بندی کنیم. هم چنین می توانیم از لیست های بازشونده^۲، دکمه های انتخاب^۳ و دیگر انواع کنترل های روی صفحه برای ورود و ویرایش آسان تر داده ها استفاده کرد.

تمرین ۲-۴ در بانک Northwind فرم های ایجاد شده را مشاهده کنید.



۱- Table Layout

۲- Pull-Down Lists

۳- Radio/Option Buttons



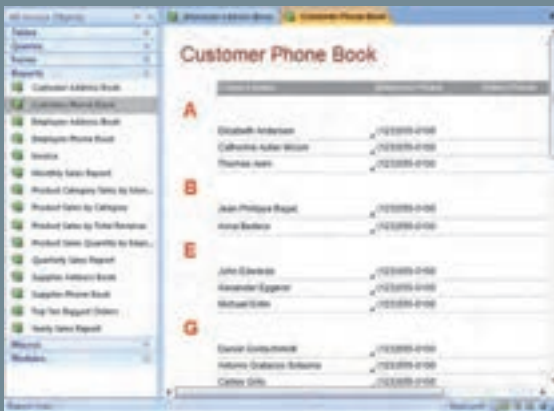
پژوهش کنید هریک از فرم‌های بالا به چه منظور ایجاد شده‌اند؟ از چه کنترل‌هایی در فرم‌ها برای ورود و ویرایش آسان‌تر داده‌ها استفاده شده است؟

راهنمایی: با انواع کنترل‌های برنامه‌نویسی در درس برنامه‌سازی آشنا شده‌اید مثل انواع List ها، انواع Button ها، انواع OptionBox ها، ChekBox ها و

۴-۳-۲- گزارش‌ها برای چاپ داده‌ها: فرم‌ها عموماً برای ورود اطلاعات و گزارش‌ها برای چاپ اطلاعات استفاده می‌شوند. گزارش‌ها مانند فرم‌ها اطلاعات را نشان می‌دهند و می‌توان قالب مشخصی را برای نمایش اطلاعات انتخاب کرد. معمولاً گزارش‌ها براساس پرس‌وجوها ایجاد می‌شوند و پرس‌وجو اطلاعات موردنظر برای نمایش گزارش را انتخاب می‌کند. در طراحی گزارش (Report Design) می‌توان ترتیب نمایش رکوردها، محلی که فیلدها نمایش داده می‌شوند، نوع قلم نمایش و اندازه آن، خطوط، فضای خالی و غیره را مشخص کرد. علاوه بر آن می‌توان تعیین کرد رکوردها بر روی کاغذ چاپ شوند، یا روی پاکت نامه، یا به‌صورت برجسب و یا انواع دیگر چاپی. اکسس ابزار جادویی (Wizard) دارد که می‌توان گزارش‌های ساده را به‌آسانی و با سرعت ایجاد کرد و یا نمودارها (Chart) را براساس داده‌های موردنظر به‌سادگی ایجاد و چاپ کرد.

در بانک Northwind گزارش‌های ایجاد شده را مشاهده کنید.

تمرین ۵-۲:





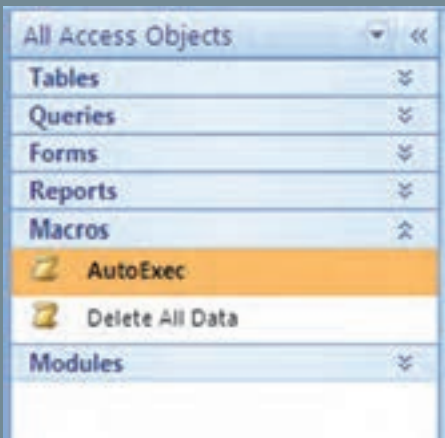
- ۱- پژوهش کنید چه شباهت‌هایی در این گزارشات مشاهده می‌شود؟
- ۲- امکانات در دسترس در گزارشات در زمان اجرا و نمایش چیست؟

راهنمایی: امکاناتی مثل چاپ، جستجو، حرکت بین صفحات و...

۵-۳-۲- ماکروها برای کنترل کلیدها : اکسس شامل دو امکان برنامه‌نویسی مجزاست : ماکروها (macro) و محیط شبیه VisualBasic (VBA). ماکروها، کد برنامه فرمان‌های مورد استفاده در محیط اکسس هستند. می‌توان با ماکرو به‌جای انجام کارها به‌صورت دستی و با کمک کلیدها، عملیات را خودکار کرد. به‌طور مثال، می‌توانید ماکرویی ایجاد کنید زمانی که جدولی باز شود نشانگر در آخرین رکورد قرار گیرد بدون آن‌که پس از باز شدن جدول، از کلیدها یا دکمه‌هایی استفاده کنید. زمانی که یک ماکرو ایجاد شد، برای اجرا در زمان مناسب می‌توانید دکمه‌ای بر روی فرم قرار دهید تا با کلیک روی آن‌ها، آن ماکرو اجرا شود. همچنین می‌توانید تعیین کنید زمانی که نشانگر یا ماوس روی یک فیلد قرار گرفت، به‌طور اتوماتیک ماکرویی اجرا شود. نیازی نیست برنامه‌نویس باشید تا ماکرویی را ایجاد کنید چرا که محیط کار کمک می‌کند تا به‌سادگی ماکرو ایجاد شود.

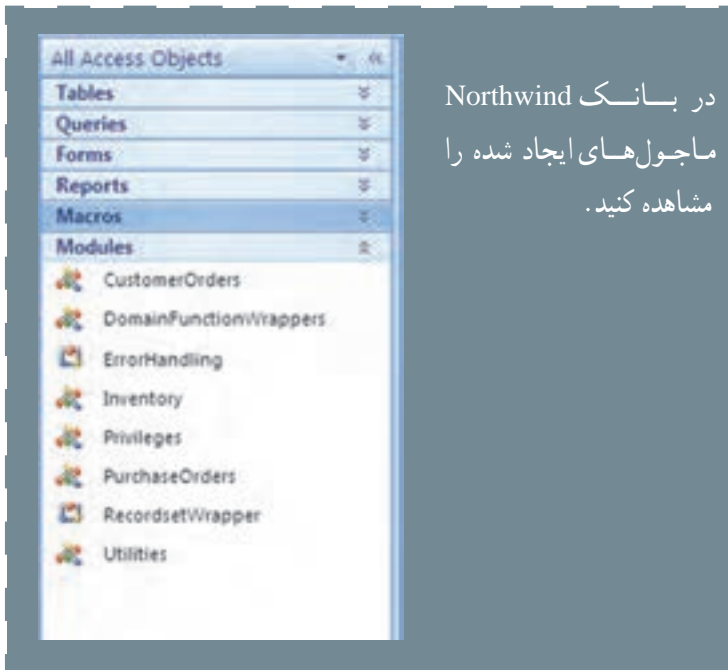
تمرین ۶-۲:

در بانک Northwind ماکروهای ایجاد شده را مشاهده کنید.
 * دقت کنید که این ماکروها پیچیده به‌نظر می‌رسند و البته کار بسیاری هم انجام می‌دهند.



۶-۳-۲- ماجول‌ها برای نوشتن برنامه‌های مورد نظر : ماجول، بخش اصلی برنامه‌نویسی

است. ماجول اصطلاح دیگری برای برنامه‌های VBA^۱ است. VBA زبان برنامه‌نویسی است که براساس زبان Basic ایجاد شده است. ماکروها برای ذخیرهٔ کلیدهای فرمان و یا پاک کردن داده‌های وارد شده در یک فیلد مناسب هستند ولی زمانی که عملیات و دستورات پیچیده‌تری نیاز باشد، باید برنامه‌نویسی کرد. مثلاً فرض کنید سفارشات کتاب را به‌وسیلهٔ پست الکترونیک دریافت کرده‌اید، آن‌گاه می‌توانید برنامه‌ای بنویسید که پیام‌ها را از صندوق پست خوانده، رکوردهایی را به جدول خاصی در بانک وارد نماید. با این کار دیگر نیازی به کپی اطلاعات به‌وسیلهٔ یک اپراتور نیست. هم‌چنین می‌توان برنامه‌ای نوشت که با کلیک روی دکمهٔ ارسال، لیست اقلام ارسالی و برچسب نامه را چاپ کرده، موجودی انبار را نیز کسر نماید و یک پیغام نیز به‌وسیلهٔ پست الکترونیک به مشتری ارسال کند.



تمرین ۲-۷:

۷-۳-۲- صفحات برای ویرایش داده‌ها در مرورگر وب : فرم‌ها برای ورود و ویرایش

داده به‌کار می‌روند ولی برای نمایش آن‌ها باید محیط اکسس اجرا شود. بدین معنی که باید اکسس روی کامپیوتر نصب شده باشد. اگر نخواهید از محیط اکسس استفاده کنید باید از صفحه‌ها (Page) استفاده

۱- Visual Basic for Application، در کتاب برنامه‌سازی ۳ با VBA آشنا می‌شوید.

کنید که فرم‌ها را دربر می‌گیرند. هر فردی می‌تواند با مرورگر وب مثل IE (Internet Explorer) صفحه شامل فرم را باز کند و مانند فرم، رکوردهای جدول را ویرایش، اضافه یا حذف نماید. اگر در اینترنت برای جستجو یا عضویت، اطلاعاتی را وارد کرده باشید، صفحه شامل فرم را دیده‌اید^۱. این گونه صفحه‌ها عموماً به بانک اطلاعاتی متصل شده‌اند و اکسس می‌تواند یکی از آن‌ها باشد.

۴-۲- ایجاد یک پایگاه داده در نرم‌افزار اکسس

برای آن‌که بتوانیم اشیاء بانک اطلاعاتی را ایجاد کنیم و در واقع سیستم بانک اطلاعاتی طراحی شده خود را پیاده‌سازی نماییم باید یک بانک پایگاه داده با پسوند accdb داشته باشیم^۲. این فایل واقعیت فیزیکی جدول‌ها، فرم‌ها، پرس‌وجوها، گزارش‌ها و سایر اشیاء بانک را دربر می‌گیرد. برای ایجاد فایل بانک می‌توانیم از ویزارد استفاده کنیم. با ویزارد می‌توانیم براساس یکی از بانک‌های اطلاعاتی از پیش تعریف شده، بانک اطلاعاتی دلخواه خود را همراه با جدول‌ها، گزارش‌ها و فرم‌های موردنیاز بسازیم. اما از آنجایی که ممکن است بانک اطلاعاتی ما با هیچ‌یک از بانک اطلاعاتی از پیش ساخته شده موجود در اکسس سازگار نباشد، این روش مناسب نیست. بنابراین ابتدا یک بانک اطلاعاتی خالی ایجاد می‌کنیم و سپس جدول‌ها، فرم‌ها، گزارش‌ها، پرس‌وجوها را به آن اضافه می‌کنیم.

– بانک اطلاعاتی DBProj.accdb را بدون کمک ویزارد ایجاد کنید.

✓ تمرین ۸-۲:

مراحل ایجاد به‌صورت زیر است:

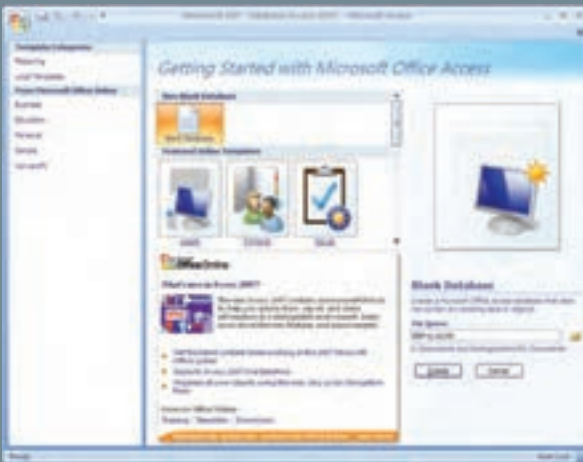
– پس از ورود به اکسس از منوی اصلی گزینه New را انتخاب کنید.

– روی گزینه Blank database از پانل ایجاد مطابق شکل صفحه بعد کلیک کنید.

– Access نام و محل ذخیره‌سازی فایل را می‌پرسد. در قسمت File Name، نام بانک اطلاعاتی را DBProj تایپ کرده، سپس دکمه Create را کلیک نمایید. همیشه محل ذخیره مناسبی را برای فایل در نظر بگیرید.

۱- در کتاب بسته‌های نرم‌افزاری ۳، جلد دوم، با چگونگی ایجاد فرم و کار کردن با آن در صفحات وب آشنا می‌شوید.
۲- در نسخه‌های قبلی نرم‌افزار Access پسوند فایل‌ها .mdb بوده است، با توجه به اینکه بیشتر عملیات موردنظر در این کتاب بر پایه نسخه ۲۰۰۷ می‌باشند از توضیح روش ذخیره‌سازی با پسوند .mdb خودداری شده است.

پس از ایجاد بانک، پنجره بانک نشان داده می‌شود که در ابتدای فصل توضیح داده شده است.



۲-۵- بستن پایگاه داده

بانک اطلاعاتی ایجاد شده از طریق منوی اصلی گزینه Close Database بسته می‌شود ولی توجه کنید که هنوز محیط کار Access فعال است، اگر چه یک سری از قابلیت‌های محیط کار Access در دسترس هستند.



چه قابلیت‌هایی در محیط کار Access در صورت فعال نبودن یک بانک داده در دسترس است؟

۲-۶- باز کردن پایگاه داده موجود

مانند باز کردن فایل در سایر نرم‌افزارهای تحت ویندوز است. مجدداً یادآوری می‌کنیم که از منوی اصلی، گزینه Open را انتخاب کنید. از پنجره باز شده، نام و محل فایل ذخیره شده را انتخاب نمایید و دکمه Open را کلیک کنید. Access بانک اطلاعاتی موردنظر را باز می‌کند و پنجره بانک نشان داده می‌شود.

خلاصه فصل

بانک اطلاعاتی اکسس یکی از ابزارهای مدیریت داده است. با افزایش حجم داده‌ها باید بتوانیم آن‌ها را تجزیه و تحلیل کنیم و اطلاعات موردنیاز را بازیافت نماییم. اکسس، مانند سایر نرم‌افزارهای تحت ویندوز دارای یک محیط کار است که ابزارها، منوها و امکانات متعددی را دربر دارد. پنجره اصلی آن علاوه بر امکاناتی که از آن‌ها نام برده‌ایم، شامل اشیاء بانک اطلاعاتی است. مهم‌ترین اشیاء بانک عبارتند از: جدول‌ها، پرس‌وجوها، گزارش‌ها و فرم‌ها.

جدول مهم‌ترین جزء بانک است که داده‌ها را نگه می‌دارد. با پرس‌وجو می‌توان داده‌های با شرط خاص را بازیافت کرد. از فرم‌ها برای نمایش، ورود و ویرایش ساده‌تر داده‌ها استفاده می‌شود؛ از گزارش‌ها برای نمایش داده‌ها و چاپ آن‌ها در طرح مناسب استفاده می‌شود.

قبل از ایجاد بانک و اشیاء آن باید، بانک اطلاعاتی طراحی گردد. هدف از طراحی، شناخت محیط عملیاتی، نیاز کاربران و در نظر گرفتن تمام امکانات موردنیاز بانک اطلاعاتی قبل از پیاده‌سازی است. باید جدول‌ها، فرم‌ها، گزارش‌ها و غیره طراحی شوند با انجام طراحی، در زمان پیاده‌سازی در محیط اکسس، با اشکالات اساسی مواجه نمی‌شویم که در غیر این صورت کار اصلاح اشکالات ایجاد شده به مراتب بسیار دشوارتر و وقت‌گیرتر خواهد بود.

- ۱- محیط کار اکسس شامل چه امکاناتی است؟
- ۲- امکانات پنجرهٔ اشیاء بانک را نام برده و مختصر توضیح دهید.
- ۳- انواع اصلی اشیاء بانک اطلاعاتی را نام ببرید و توضیح دهید.
- ۴- نام پیش فرض و پیوند فایل‌هایی که در نرم افزار اکسس ایجاد می شود را بنویسید.
- ۵- تفاوت گزارش و فرم از نظر کاربرد را بنویسید.
- ۶- ماکروها و برنامه‌هایی نوشته شده با VBA در محیط اکسس چه تفاوتی از نظر امکانات دارند؟

فصل ۳

ایجاد و کار با جدول‌ها

در فصل دوم با مقدمات بانک اطلاعاتی و از آن جمله جدول آشنا شدید. در این فصل ایجاد جدول و عملیات روی آن را بررسی می‌کنیم.

- هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:
- به وسیله روش‌های گوناگون جدول‌ها را ایجاد کند.
- فیلدهای جدول را متناسب با طراحی انتخاب کند و خصوصیات فیلدها را تغییر دهد.
- به کمک Import Table و Link Table جدول را از بانک‌های دیگر موجود به بانک خود منتقل کند.
- ثبت، درج، ویرایش و حذف داده‌ها در جدول را انجام دهد.
- طراحی جدول را تغییر دهد.
- داده‌ها را جستجو کند، با اعمال شرایطی داده‌ها را فیلتر نماید و یا مرتب کند.
- ارتباط بین جدول‌ها را در بانک ایجاد کند.

در نرم افزار اکسس، سه روش اصلی برای ایجاد جدول وجود دارد که عبارتند از :

– ایجاد جدول در نمای Design

– ایجاد جدول به وسیله Table Templates (Wizard)

– ایجاد جدول با ورود اطلاعات (در نمای Datasheet)

علاوه بر این سه روش می توان جدول را از جدول های موجود در پایگاه داده دیگر نیز ایجاد کرد که خود به دو صورت انجام می گیرد: Import Table و Link Table.

در فصل های قبل سیستم ثبت نام محیط عملیاتی مدرسه را طراحی کردیم و جدول ها، فرم ها و گزارش های مورد نیاز را به دست آوردیم. می خواهیم در این فصل جدول هایی با مشخصات زیر را در بانک اطلاعاتی ایجاد شده DBProj، وارد نماییم.

(محیط عملیاتی = مدرسه)
(سیستم = ثبت نام)

1. دروس Course-Tbl

Name	Type	Size
→ CourseCode	Text	5
CourseName	Text	20
CourseUnit	Byte	-
TeacherCode	Integer	-
ProgramHour	Text	50
ProgramClass	Text	50

2. رشته تحصیلی Major-Tbl

Name	Type	Size
→ MajorCode	Integer	-
MajorName	Text	20

۱- Table Wizard در نسخه ۲۰۰۷ به بعد عملاً حذف شده است و بجای آن گزینه بسیار محدودی به نام Table Template فرار داده شده که می تواند در شرایط خاص کاربرد داشته باشد.

3. Tbl-Register ثبت نام

Name	Type	Size
RegisterCode	Integer	-
StudentCode	Integer	-
RegisterDate	Date/Time	-

4. Tbl-RegisterItem درس های ثبت نام شده

Name	Type	Size
RegisterCode	Integer	-
CourseCode	Text	5
Grade	Decimal	-

(Precision = 5, Scale = 2, Decimalplace = 2)

5. Tbl-Student دانش آموز

Name	Type	Size
StudentCode	Integer	-
StudentFName	Text	20
StudentLName	Text	30
StudentTel	Long Integer	-
StudentAddress	Memo	-
MajorCode	Integer	-

6. Tbl-Teach معلم

Name	Type	Size
TeacherCode	Integer	-
TeacherFName	Text	20
TeacherLName	Text	30
TeacherTel	Long Integer	-
TeacherAddress	Memo	-

۳-۱- ایجاد جدول درنمای Design

– در لیست اشیاء روی گزینه Tables کلیک کنید.

– دکمه New را انتخاب نمایید.

– از زبانه Create نوار ابزار، گزینه Table Design را انتخاب نمایید.

در پنجره طراحی باز شده جدول شکل ۳-۱، اطلاعات لازم را وارد کنید که در ادامه به تفصیل

توضیح داده خواهد شد.



شکل ۱-۳- پنجره طراحی جدول برای جدول Books

۱-۱-۳ اجزای پنجره طراحی جدول : با توجه به پنجره طراحی جدول شکل ۱-۳، دو ستون نام فیلد و نوع فیلد برای تعریف فیلد ضروری هستند. هر فیلد در یک ردیف تعریف می‌شود. هر فیلد غیر از نام و نوع، مشخصات دیگری نیز دارد که در پنجره پانل سایر مشخصات فیلد تنظیم می‌شوند.

● **نام فیلد** : نام فیلد بهتر است به گونه‌ای انتخاب شود که نشان دهنده محتوای آن باشد. مثلاً عنوان کتاب، Title نامیده شود. بهتر است از حروف لاتین استفاده کنید هر چند مجاز هستید از کاراکتر Space در نام استفاده کنید ولی بهتر است از آن استفاده نکنید. اندازه نام نباید بیش از ۶۴ کاراکتر باشد.

● **نوع فیلد** : نوع داده‌ای است که در فیلد وارد می‌شود. عبارت است از موارد زیر :

نوع فیلد	توضیحات
Text	نوع پیش فرض است. این نوع هر ترکیبی از حروف و ارقام است. حداکثر تا ۲۵۵ کاراکتر در این نوع فیلد امکان پذیر است.
Memo	از نوع متن است که تا ۶۴k کاراکتر را ذخیره می‌کند.
Number	فقط اعداد را ذخیره می‌کند. نوع صحیح در اندازه ۱۶ بیتی و با ۳۲ بیتی، نوع ممیز اعشار یا سایر انواع عددی دیگر که در قسمت General در پانل پایین پنجره طراحی جدول (نمای Design) تعیین می‌شوند. پانل پایین را در ادامه توضیح می‌دهیم.

Date/Time	تاریخ، زمان و یا ترکیبی از آن‌ها را نگه می‌دارد. به خاطر داشته باشید که این نوع به تاریخ لاتین حساس است و در صورتی که برای نگهداری تاریخ شمسی استفاده شود در برخی تاریخ‌ها مثلاً «۱۳۸۵/۰۲/۳۱» پیغام عدم اعتبار خواهد داد. بنابراین برای نگهداری تاریخ شمسی بهتر است از این نوع استفاده نشود بلکه از Text استفاده کنید.
Currency	مقادیر پولی را نگه می‌دارد و می‌تواند به‌طور خودکار علامت \$ داشته و محل‌های علامت جداساز هزارتایی «کاما» را نگه دارد.
Auto Number	در هنگام ایجاد رکورد جدید، اکسس به‌طور خودکار عدد صحیح منحصر به فرد را برای آن جدول، تولید و به این فیلد منسوب می‌کند. در قسمت General در پانل سایر مشخصات فیلد در ردیف New Values لیست بازشونده‌ای شامل دو مقدار وجود دارد: Increment که عدد منحصر به فرد به‌طور سریال تولید می‌شود و Random که هر عددی به‌طور تصادفی و منحصر به فرد ایجاد می‌شود. از آن جایی که هر جدول باید یک فیلد کلید اصلی داشته باشد در صورتی که از فیلدهای جدول یعنی صفات موجودیت مرتبط توانیم کلید را مشخص کنیم، ایجاد فیلد جدیدی از این نوع به عنوان فیلد کلید اصلی، می‌تواند مفید باشد.
Yes/No	این نوع می‌تواند مقادیر دو ارزشی True/False، Yes/No، On/Off و یا دیگر مقادیر را دربر گیرد.
\OLE Object	این نوع می‌تواند صدا، تصویر و یا دیگر اشیاء را دربرگیرد. مثل فایل Word و یا Excel که به وسیله برنامه‌های دیگر تولید می‌شوند.
Hyperlink	نوعی است که به آدرس وبسایت یا محل بانک دیگر وصل می‌شود. داده‌های این نوع شامل حداکثر چهار بخش است که به وسیله علامت # جدا می‌شوند. Displaytext#Address#Subaddress#Screentip به خاطر بسپارید که قسمت Address تنها بخشی است که حتماً باید وارد شود نمونه‌هایی از این نوع داده را می‌بینید: ● لینک اینترنتی: HomePage#http://www.microsoft.com # صفحه اول سایت مایکروسافت که با عنوان "HomePage" روی مرورگر باز می‌شود. ● لینک به پایگاه داده Students##Form Student#Click to view and edit students information فرم Students که در پایگاه داده جاری قرار دارد و پیغام Click... را نمایش می‌دهد. ● لینک به پایگاه داده دیگر: #c:\Microsoft Office\Northwind.mdb#Form Customers فرم Customers که در پایگاه داده Northwind قرار دارد را نشان می‌دهد. ● لینک به برنامه دیگر: # c:\windows\personal\student.ppt#13 اسلاید ۱۳ ارائه Power Point در آدرس C:\... را نشان می‌دهد. #c:\windows\personal\student.xls#Sheet1! c:\... سلول A2 را در Sheet1 صفحه گسترده Excel در آدرس C:\... را نشان می‌دهد.

۲-۱-۳ پانل سایر مشخصات فیلد **Field Properties** : یک فیلد علاوه بر نام و نوع داده‌ای، مشخصه‌های دیگری نیز دارد که در این قسمت در اختیار قرار می‌گیرند. این مشخصه‌ها نحوه ذخیره‌سازی، دستکاری و نمایش مقدار داده ذخیره شده در آن فیلد را کنترل و تنظیم می‌کنند. با انتساب این مشخصه‌ها زمان زیادی در مرحله ساخت فرم، گزارش و اشیاء دیگر پایگاه داده صرفه‌جویی می‌شود یعنی زمان کمتری در ایجاد آن‌ها صرف خواهد شد. پنجره مشخصه‌های فیلد، بسته به نوع داده‌ای که برای آن فیلد در نظر گرفته‌اید تغییر می‌کند.

برخی از مشخصه‌های این پنجره عبارت است از :

مشخصه	توضیحات
Field Size	اندازه نوع فیلد text را به تعداد مشخصی کارا کمتر محدود می‌کند و یا نوع فیلد Number را به یک دامنه اعداد محدود می‌کند.
Format	قالب نمایش داده در نمای Datasheet است.
Decimal places	تعداد ارقام سمت راست نقطه اعشار را تعریف می‌کند. این مشخصه در نوع داده Number، در صورتی که Format مقدار خالی و یا مقدار General باشد اثری ندارد و در سایر موارد می‌تواند Format تعیین شده را تغییر دهد.
Input Mask	قالبی است که تعیین می‌کند داده در زمان ورود اطلاعات به چه شکلی دریافت شود. مثلاً برای تلفن موبایل تعریف #####\0#### نمونه داده : 0912-1234567 را دریافت می‌کند. یعنی ابتدای آن صفر و پس از چهارمین رقم، علامت - قرار گیرد و کلاً ۱۱ رقم باشد.
Caption	برچسبی است که به جای نام فیلد، در جدول‌ها، فرم‌ها و گزارش‌ها نمایش داده می‌شود.
Default Value	مقدار پیش‌فرضی است که در زمان ایجاد یک رکورد جدید اگر مقداری به فیلد مربوطه اختصاص داده نشده باشد، در آن ذخیره می‌شود.
Validation Rule	شرطی است که داده‌های وارد شده را محدود می‌کند. مثلاً این که داده وارد شده بیش از ۱۰۰۰۰ نباشد.
Validation Text	متنی است که در زمان عدم احراز شرایط Validation Rule به عنوان پیغام خطا یا اخطار نمایش داده می‌شود.
Required	مشخص می‌کند در زمان ورود داده، آن فیلد نمی‌تواند خالی رها شود بلکه حتماً باید مقداری به آن نسبت داده شود.
Allow Zero Length	مشخص می‌کند آیا می‌توان مقدار خالی یعنی "" را به فیلد نسبت داد یا خیر.
Indexed	باز یافت داده از فیلد را تسریع می‌کند. تمام فیلدهای کلیدی اصلی به‌طور پیش‌فرض Indexed هستند. زمانی که مقدار آن yes انتخاب شود، می‌توانیم تعیین کنیم مقدار تکراری در آن فیلد قرار گیرد یا خیر.



نحوه شرکت مقدار عددی در جملات محاسباتی با نوع Field Size متفاوت چگونه است؟

راهنمایی: نتیجه تقسیم ۱۳ بر ۲ (۱۳/۲) می‌تواند به دو جواب برسد. برحسب آن که مقدارها Integer فرض شوند ۶ و با فرض مقدارها به نوع اعشاری ۶/۵ خواهد بود.

مقادیر مشخصه Format نوع داده اعداد و نوع داده تاریخ / زمان

● **انتسابات قالب عددی (Format Number):** خصوصیت Format نحوه نمایش داده را در صفحه نمایش Datasheet مشخص می‌کند. این مشخصه بر نحوه ذخیره شدن داده در جدول و یا بر نحوه شرکت مقدار عددی ذخیره شده آن در محاسبات تأثیری ندارد.

مشخصه Format برای نوع داده Number

عدد را به همان صورتی که وارد شده است، نمایش می‌دهد. قالب پیش فرض است.	General
از کاما به عنوان جداساز هزارتایی استفاده می‌کند و علامت ارز را نیز نشان می‌دهد. اعداد منفی در پرانتز نمایش می‌یابند و مقدار پیش فرض $\text{Decimal Place}=2$ است.	Currency
حداقل یک رقم را نشان می‌دهد و مقدار پیش فرض $\text{Decimal Place}=2$ است. اعداد حول تعداد ارقام راست نقطه اعشار گردا می‌شوند. بانک از کاما برای نمایش جداساز هزارتایی استفاده می‌کند.	Fixed
عدد وارد شده را در 100° ضرب می‌کند و علامت درصد را به انتهای آن اضافه می‌کند. مقدار پیش فرض $\text{Decimal Place}=2$ است.	Percent
اعداد را به صورت علمی توان 10° و با نماد E نشان می‌دهد.	Scientific

● **انتسابات تاریخ / زمان (Date/Time Format) :** قالب نمایش تاریخ و یا زمان را تغییر

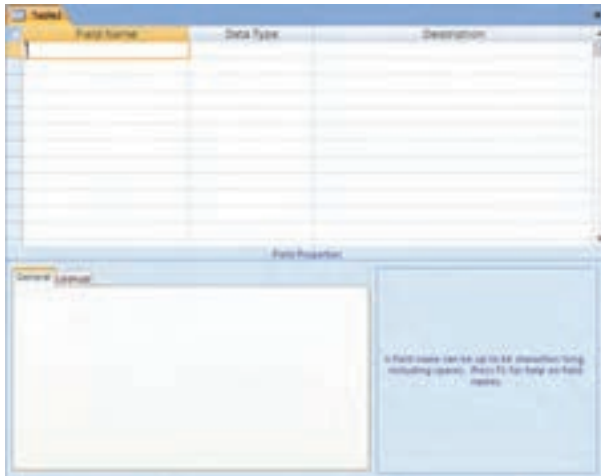
می دهد .

مشخصه Format برای نوع داده Date/Time

اگر مقدار فقط تاریخ بدون زمان باشد و یا اگر مقدار زمان بدون تاریخ باشد، از آن استفاده می شود.	General Date
اسامی روز و ماه به صورت متنی نوشته می شود. مثل : Tuesday, July 4, 1995	Long Date
نام ماه به صورت مخفف نشان داده می شود و نام روز حذف می شود. مثل : 04-Jul-95	Medium Date
تاریخ به صورت رقم که با علامت "/" جدا شده اند نمایش داده می شود. مثل : 7/4/95.	Short Date
زمان به صورت ساعت، دقیقه و ثانیه که با علامت ":" جدا شده اند، نمایش می یابد به همراه علامت AM برای صبح و PM برای بعد از ظهر. مثل : 6:30:15PM	Long Time
مشابه قالب Long Time است جز آن که ثانیه نمایش نمی یابد مثل : 06:30PM	Medium Time
زمان به صورت ۲۴ ساعته بدون ثانیه نمایش داده می شود. مثل : 18:30	Short Time

مراحل ایجاد جدول Tbl-Course به وسیله Design

- از زبانه Create نوار ابزار گزینه Table Design را انتخاب نمایید.
- پنجره طراحی جدول خالی نمایش می یابد. مطابق شکل زیر :
- در سلول نام فیلد ردیف اول CourseCode را تایپ کنید.

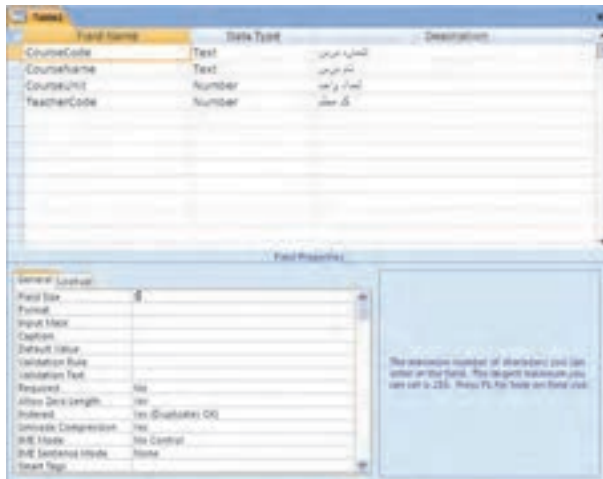


مثال ۳-۱

– با فشردن کلید [Tab] در ستون دوم نوع فیلد (Data type)، نوع داده را انتخاب کنید که به طور پیش فرض text است. [با کلیدهای جهتی (Arrow Key) صفحه کلید، انواع دیگر داده را می‌توانید انتخاب کنید]. همان‌طور که در فرض مثال می‌بینید نوع داده کد درس متنی و از نوع text انتخاب شده است.

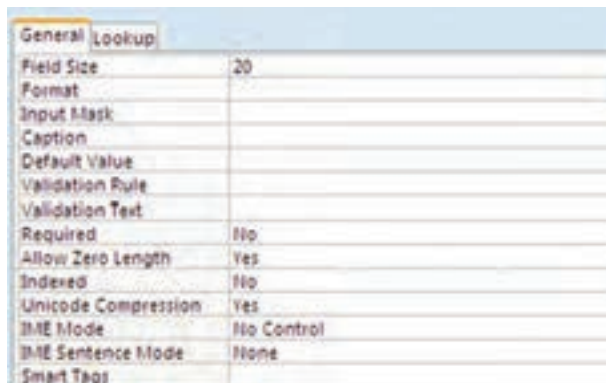
– با کلید [Tab] در ستون توضیحات قرار بگیرید. هرچند الزامی به تکمیل آن نیست ولی بهتر است توضیحاتی در مورد آن فیلد به صورت فارسی در آن تایپ شود.

– با کلید [Tab] در سطر بعد قرار بگیرید. سایر فیلدها را به همین روش تعریف کنید.

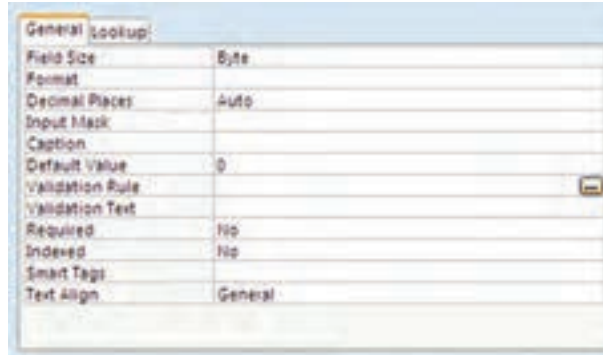


– سایر مشخصات فیلدهای CourseName و CourseUnit، TeacherCode را مطابق شکل‌های زیر تنظیم کنید.

CourseName:

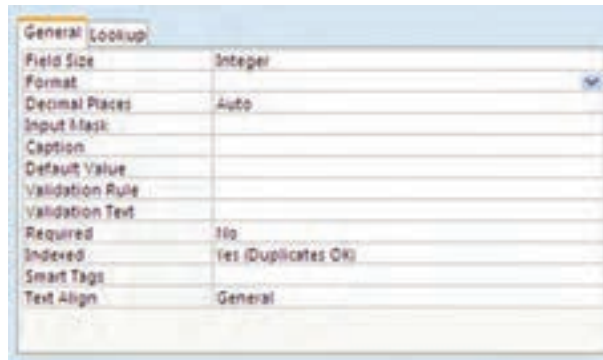


CourseUnit:



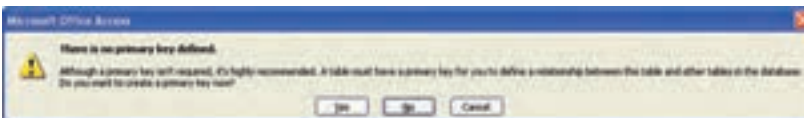
General Lookup	
Field Size	Byte
Format	
Decimal Places	Auto
Input Mask	
Caption	
Default Value	0
Validation Rule	
Validation Text	
Required	No
Indexed	No
Smart Tags	
Text Align	General

TeacherCode:




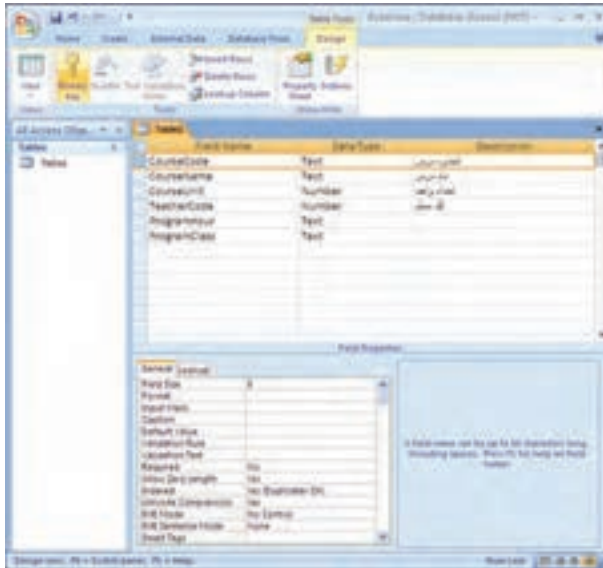
General Lookup	
Field Size	Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (Duplicates OK)
Smart Tags	
Text Align	General

– پس از تعریف تمام فیلدها، دکمه Save از نوار ابزار یا گزینه Save از منوی اصلی را انتخاب کنید تا پنجره ذخیره‌سازی جدول ظاهر شود.
– نام آن را Tbl_Course قرار دهید و OK را کلیک کنید. چون هنوز فیلد کلید اصلی را مشخص نکرده‌ایم، پیغام زیر نمایش داده می‌شود.



– در صورتی که جدول، فیلد کلید اصلی تعریف شده‌ای نداشته باشد می‌توان از قابلیت Auto Number نرم‌افزار Access استفاده کرد. در این صورت با انتخاب دکمه Yes فیلد جدیدی به نام ID و از نوع Auto Number (خودشمارنده) ایجاد می‌شود. در اینجا کلید جدول ما در طراحی مشخص شده است بنابراین دکمه Cancel را انتخاب و کلید را به صورت زیر تعریف می‌کنیم.

– مطابق شکل زیر، زمانی که در حالت Design هستیم، فیلد مورد نظر برای کلید را در پنجره طراحی جدول انتخاب کرده، از نوار ابزار دکمه علامت کلید  را کلیک کنید. در کنار نام فیلد علامت کلید ظاهر می شود.

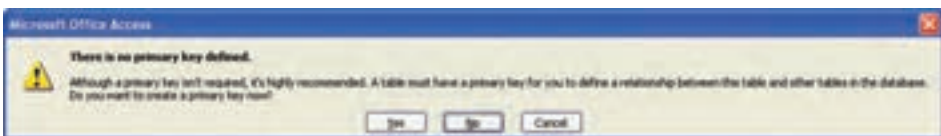


– در این حال به هنگام ذخیره سازی پنجره نام جدول مجدداً ظاهر می شود. نام جدول را Tbl-Course قرار دهید و آن را ذخیره نمایید. توجه کنید که پیغام قبلی ظاهر نمی گردد و جدول ذخیره می شود.



اگر کلید شامل بیش از یک فیلد باشد، چگونه در اکسس مشخص می شود؟

گزینه No در پیغام زیر چگونه عمل می کند؟



تمرین ۱-۳: جدول Tbl_RegisterItem را به این روش ایجاد کنید.



۳-۲- ایجاد جدول به وسیله قالب‌های آماده

کاربر می‌تواند با استفاده از جدول‌ها و فیلدهای پایگاه داده از پیش تعریف شده و با پاسخ دادن به یک سری سؤالات درباره داده‌ای که می‌خواهد ذخیره شود، جدول را ایجاد کند. استفاده از قالب‌های آماده مزایا و معایبی دارد که از آن جمله عبارت است از: جدول‌های پایگاه داده بدون درک طراحی و شناخت نوع فیلد، ایجاد می‌شوند، ولی کنترل کمی به وسیله کاربر می‌تواند در آن لحاظ شود.



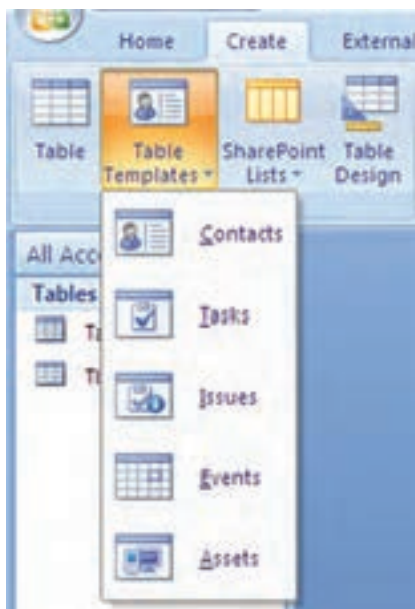
مزایا و معایب استفاده از قالب‌های آماده را بررسی کنید.

مراحل ایجاد جدول Tbl-Teacher به وسیله قالب‌های آماده

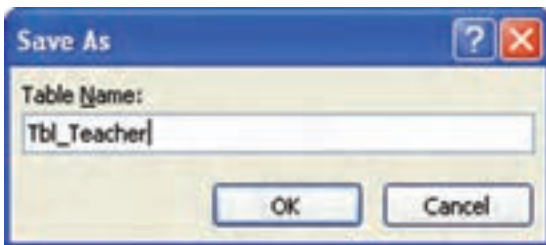
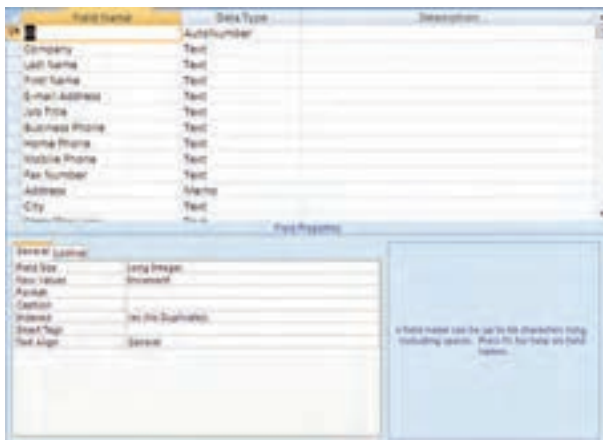
– از زبانه Create نوار ابزار، روی گزینه

Table Templates کلیک کرده و مطابق شکل،

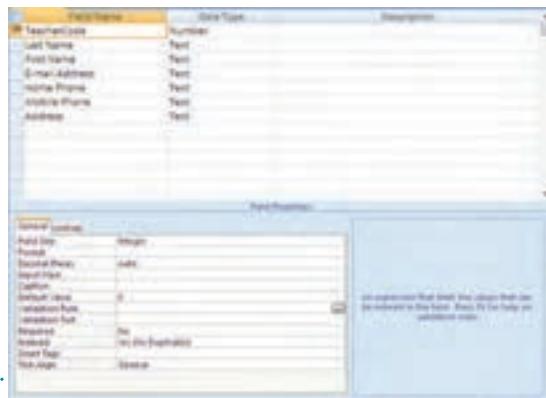
جدول Contacts را انتخاب نمایید.



– مطابق شکل از زبانه Home نوار ابزار، نمای Design را انتخاب نمایید.



– با استفاده از دکمه موجود در کنار منوی اصلی برنامه و یا کلید میانبر Ctrl+S جدول ساخته شده را به نام Tbl_Teacher ذخیره کنید.



– ساختار جدول را براساس شکل روبه‌رو تغییر دهید، همانطور که ملاحظه می‌کنید فیلد ID به TeacherCode تغییر نام پیدا کرده و نوع داده آن نیز عوض شده است. برای تغییر سایر فیلدها به ترتیب

- زیر عمل نمایید.
- فیلدهای BusinessPhone، JobTitle، Company و FaxNumber را حذف نمایید.
 - نوع داده فیلد Address را از Meno به Text تغییر دهید.
 - تمام فیلدهای بعد از Address را حذف نمایید.

✓ **تمرین ۲-۳:** جدول Tbl_Student را به این روش ایجاد کنید.

۳-۳- ایجاد جدول به وسیله نمای Datasheet

در این حالت اطلاعات در سلولها وارد می‌شوند بدون آن که فیلدها از قبل تعریف شده باشند. با ذخیره اطلاعات وارد شده، Access برحسب مقادیر داده وارد شده در هر ستون، نوع فیلد را حدس زده و نوع داده فیلد را تعیین می‌کند. بدین ترتیب جدول ایجاد می‌شود.



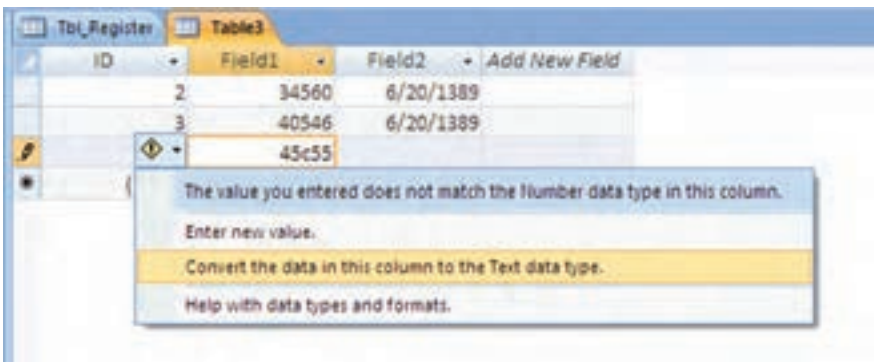
مراحل ایجاد جدول
Tbl-Register به وسیله
Datasheet

– از زبانه Create نوار
ابزار، گزینه Table را انتخاب
نمایید.

– مطابق شکل، در پنجره باز شده، اطلاعات را وارد نمایید.

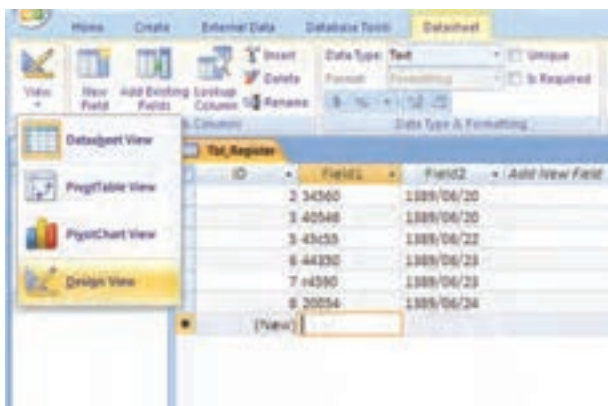
ID	Field1	Field2	Field3	Add New Field
2	34560	1389/06/20		
3	40546	1389/06/20		
5	45c55	1389/06/22		
6	44350	1389/06/23		
7	r4590	1389/06/23		
8	20054	1389/06/24		
(New)				

– با ورود داده‌های رکورد سوم پیامی مبنی بر تغییر نوع داده ورودی نمایان می‌شود که با انتخاب گزینه مشخص شده در تصویر زیر در واقع تغییر نوع داده فیلد مربوطه را تأیید می‌کنیم.

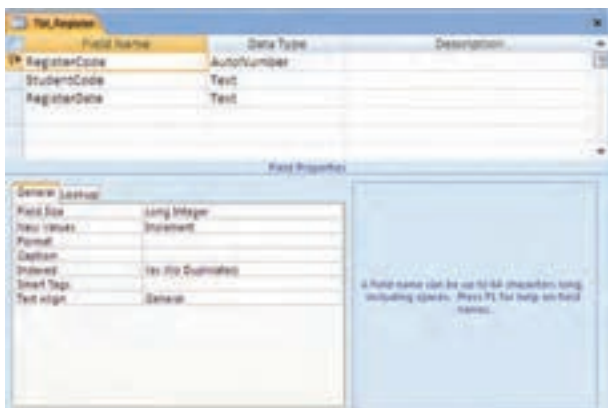
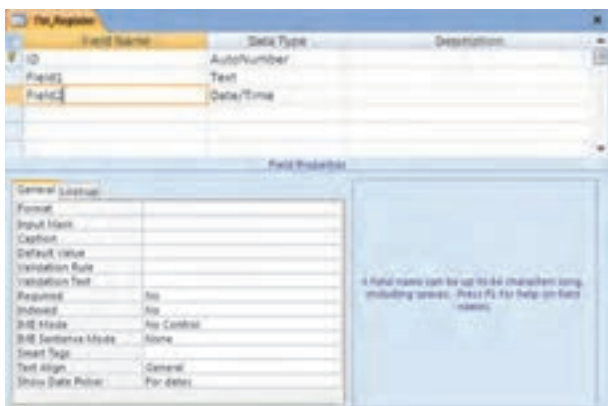


– پس از آن، دکمه Save از نوار ابزار یا گزینه Save از منوی اصلی را انتخاب کنید تا پنجره ذخیره‌سازی جدول ظاهر شود.

– نام آن را Tbl-Register قرار دهید و OK را کلیک کنید.
 – پنجره ورود داده را ببندید. Tbl-Register در اشیاء جدول بانک ثبت شده است.



– از نوار ابزار
 گزینه Design View
 را انتخاب کنید. تا پنجره
 طراحی جدول (شکل
 روبه‌رو) نمایان شود.



– نام فیلد و نوع
 فیلد را مطابق شکل
 روبه‌رو تصحیح نمایید.

– به دلیل آن که اندازهٔ فیلد StudentCode را از ۵۰ به ۵ تغییر داده‌ایم، پیغام زیر نشان داده می‌شود.



– بدون هیچ ترسی دکمهٔ Yes را انتخاب کنید! اندازهٔ داده‌های وارد شده در این فیلد از ۵ کاراکتر بیشتر نیست و داده‌ای حذف نخواهد شد. در سایر مواقع کاراکترهای بیش از ۵ از بین می‌روند.

– جدول اصلاح شده ذخیره می‌شود.

۳-۴- ایجاد جدول به وسیلهٔ Link Table و Import Table

با انتخاب گزینه‌های Import یا Link در پنجرهٔ New Table، می‌توان از جدول‌های پایگاه دادهٔ موجود، جدول‌هایی مشابه ایجاد کرد.

در حالت دریافت Import، هیچ‌گونه ارتباطی بین پایگاه دادهٔ فعلی و بانک قبلی وجود نخواهد داشت و طراحی جدول به همراه داده‌های موجود در آن به بانک جدید کپی می‌شود. ولی در حالت Link، جدول مبدأ در بانک مربوطه‌اش قرار دارد و جدول افزوده شده به آن متصل شده است و در صورتی که محل فیزیکی بانک مبدأ تغییر کند، لینک جدول افزوده شده بی‌اثر خواهد شد.

داده‌هایی که در بانک اضافه می‌شوند، در جدول لینک شده نمایش می‌یابند و داده‌های وارد شده به جدول لینک شده، در جدول مبدأ اضافه می‌شود. در این حالت، طراحی جدول لینک شده را در بانک جدید نمی‌توان تغییر داد، بلکه باید در بانک مبدأ تغییرات اعمال شود.

با انتخاب هر یک از این گزینه‌ها، ابتدا نام و مسیر بانک موجود و موردنظر را انتخاب می‌کنیم، سپس از پنجرهٔ باز شده جدول یا جدول‌های موردنظر را انتخاب و Add می‌کنیم.

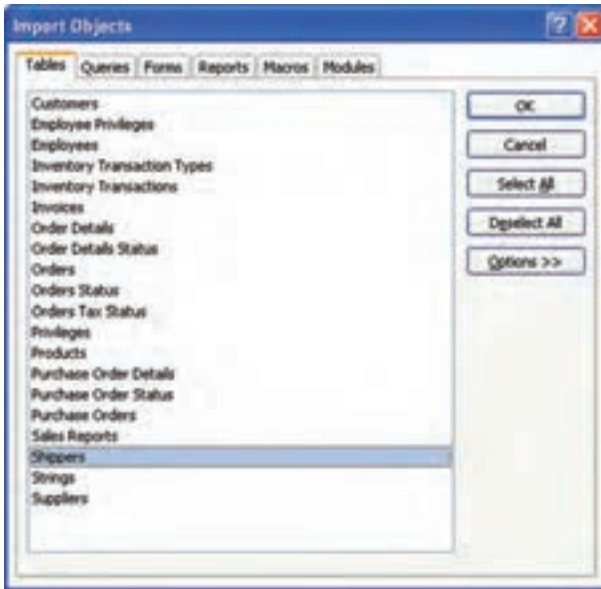
- ۱- آیا می‌توان با روش **Import و Link** سایر اشیای بانک داده را نیز منتقل کرد؟ این کار چگونه صورت می‌گیرد؟
- ۲- کدام اشیاء قابل انتقال نیستند؟
- ۳- آیا شیء خاصی وجود دارد که فقط به یکی از این دو روش منتقل شود؟

مراحل ایجاد جدول رشته‌های تحصیلی (Tbl-Major) به وسیله **Import** – از زبانه **External Data** نوار ابزار، گزینه **Import Access Database** را انتخاب نمایید. مطابق شکل زیر:



– در پنجره باز شده مسیر بانک **Northwind** را انتخاب کنید و کلیک نمایید.



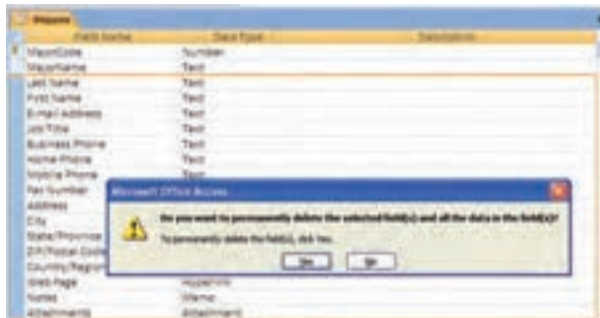


– از پنجره باز شده مطابق شکل رو به رو، Shippers را انتخاب و کلید OK را انتخاب کنید.

– در صورت نیاز می‌توانید مراحل وارد کردن اشیاء بانک خارجی را در پنجره بعدی با انتخاب گزینه Save import Steps ذخیره کنید.

– جدول Shippers به لیست اشیاء جدولی بانک اضافه می‌شود. آن را انتخاب و گزینه Design را از نوار ابزار پنجره بانک انتخاب کنید و تغییرات را مشابه مشخصات بانک اعمال کنید. یعنی تغییر نام ID به MajorCode و تغییر مشخصات فیلد آن به Number و نوع Integer. تغییر نام Company به MajorName و تغییر اندازه آن به عدد ۲۰. همچنین ردیف فیلد Phone و سایر فیلدهای اضافه را انتخاب و کلید Delete را فشار دهید تا حذف شوند.

– با فشردن دکمه Delete در ردیف Phone، پیغام زیر ظاهر می‌شود. Yes را انتخاب کنید.

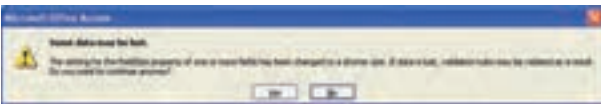


– ممکن است حذف بعضی از فیلدها باعث حذف امکانی از جدول شود که به آن Index گفته می‌شود در صورت برخورد می‌توانید فرم مربوطه را تأیید کنید.

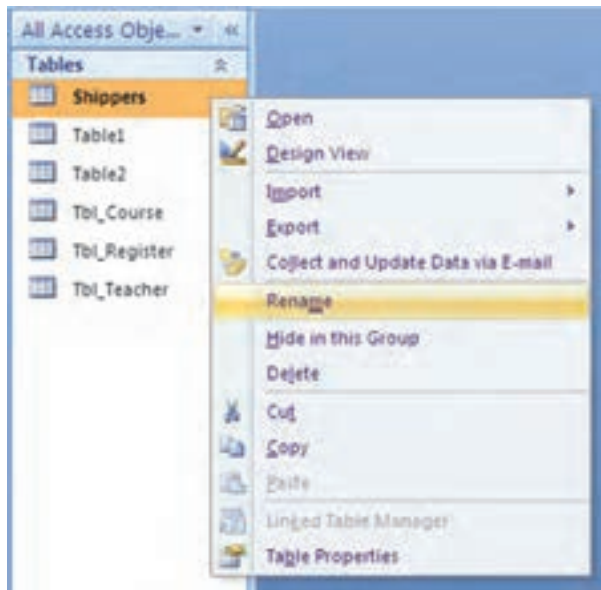


– داده‌های ستون Description را پاک کنید یا متناسب با تعریف فیلد جدید اصلاح نمایید.

– به دلیل کاهش اندازه فیلد، هنگام ذخیره جدول، پیغام زیر نمایش داده می‌شود، دکمه Yes را انتخاب کنید.



– جدول Shippers به لیست اشیاء جدول اضافه شده است. آن را به Tbl_Major تغییر نام دهید. این کار با روش‌های مشابه تغییر نام فایل در محیط ویندوز قابل انجام است.



۱- شاخص‌گذاری (Indexing) یکی از ابزارهای سودمند پایگاه داده جهت سهولت در بازیابی و جستجوی اطلاعات می‌باشد که تشریح عملکرد و کاربرد آن از حوصله این کتاب خارج است.

آیا روش ایجاد Link مشابه روش Import است؟ دلایل استفاده از روش Link را توضیح دهید و در این محیط عملیاتی برای کدام جدول / جدول‌ها، این اتصال معنی‌دار است؟

۵-۳- ورود داده‌ها به جدول‌ها

برای وارد کردن مقادیر داده‌ها در جدول، ابتدا جدول را از پنجره اشیاء بانک، انتخاب کنید. سپس روی نام جدول موردنظر دوبار کلیک کنید و یا از دکمه‌های فرمان، Open را کلیک نمایید. نمایی مشابه DataSheet و به صورت زیر (شکل ۲-۳) نمایش داده می‌شود. جدول انتخاب شده Tbl-Major است.



شکل ۲-۳- نمای DataSheet

● افزودن رکوردها: با تایپ مقادیر فیلدها در ردیف آخر رکوردها (علامت *) یک رکورد جدید ایجاد می‌شود. همچنین می‌توانید دکمه New Record را در پایین Datasheet کلیک کنید و مقادیر را وارد نمایید. Access برای ذخیره مقادیر داده پیغامی نمایش نمی‌دهد و همزمان با تایپ کردن داده‌ها، ذخیره‌سازی انجام می‌شود.

تمرین ۳-۳: مقدار Test a Major را با کد درس ۱۰ ایجاد کنید.

✓ **تمرین ۴-۳:** اگر مقدار تکراری در فیلد کلید اضافه شود چه اتفاقی می افتد؟ اگر در فیلد با نوع عددی، کاراکتر الفبایی تایپ شود چه اتفاقی می افتد؟

● **ویرایش رکوردها:** برای ویرایش رکورد به سادگی نشانگر را روی رکورد ببرید و تغییرات موردنظر را اعمال کنید. با استفاده از کلیدهای جهتی می توانید از یک فیلد به دیگری جابه جا شوید. همچنین می توانید از دکمه های قبلی (Prev)، بعدی (Next)، اولین (First) و آخرین (Last) در پایین Datasheet نیز برای این کار استفاده کنید. مشابه افزودن داده، برای ویرایش نیز پیغامی نمایش نمی دهد و هم زمان با تایپ، تمام تغییرات لحاظ می شود. گرچه می توان با کمک Undo Typing از منوی Edit، تغییرات را به حالت قبل برگرداند ولی با بستن پنجره جدول، تغییرات قطعی می شود.

✓ **تمرین ۵-۳:** در جدول Tbl-Major، مقدار Test a Major را به مقدار «ریاضی فیزیک» تغییر دهید.

● **حذف رکوردها:** با انتخاب رکورد موردنظر و انتخاب Delete Record از منوی Edit و یا کلیک گزینه حذف از نوار ابزار محیط کار، می توان آن رکورد را حذف کرد. ابتدا پیغامی برای تأیید حذف، نمایش می یابد و در صورت تأیید، رکورد حذف می شود.

✓ **تمرین ۶-۳:** در جدول Tab-Major، رکوردهای Shipping Company A، Shipping Company B و Shipping Company C را حذف کنید.

● **تغییر اندازه سطرها و ستون ها:** ارتفاع سطرها در نمای Datasheet با کشیدن خط نوار خاکستری بین ردیف ها که با جابه جایی ماوس به شکل صلیب (↔) درمی آید قابل انجام است. با تغییر ارتفاع یک ردیف تمام ردیف ها در نمای Datasheet تغییر می کند. عرض ستون ها به شکلی مشابه ردیف قابل تغییر است. با دوبار کلیک روی خط نیز به طور خودکار اندازه، مناسب با بزرگ ترین مقدار وارد شده در ستون تنظیم می شود. برخلاف ردیف ها، هر ستون می تواند اندازه متفاوتی داشته باشد.

مقادیر دقیق تر به وسیله انتخاب Row Hieght و Column Width از منوی Format و وارد کردن

مقدار طول و عرض قابل تنظیم است.

● **ثابت کردن ستون‌ها:** مشابه حالت Freezing در Excel می‌توان، ستون‌های جدول را در Access ثابت کرد. در این صورت ستون، در ابتدایی‌ترین قسمت نما قرار می‌گیرد.

برای ثابت‌سازی، ستون موردنظر را انتخاب و Freeze Columns را از منوی Format انتخاب می‌کنیم.

برای برگشت ثابت‌سازی، ستونی را انتخاب و UnFreeze All Columns را از منوی Format انتخاب می‌کنیم.

● **پنهان‌سازی ستون‌ها:** ستون‌ها می‌توانند -گرچه در بانک وجود دارند- از دید کاربر پنهان باشند.

برای پنهان‌سازی، ستون موردنظر و یا ستون‌های هم‌جوار موردنظر را انتخاب و از زبانه Home، گزینه Records، گزینه More...، Hide Columns را انتخاب می‌کنیم.

برای نمایش مجدد ستون، از منوی Unhide Columns، Format را انتخاب می‌کنیم. پنجره‌ای باز می‌شود که تمام فیلدها و در کنار هر یک، یک Check box دیده می‌شود. با انتخاب Check box کنار فیلدهای موردنظر و بستن پنجره، ستون‌ها بر حسب انتخاب یا عدم انتخاب Check box، ظاهر و یا پنهان می‌شوند.

● **جستجو و یافتن داده در جدول:** می‌توان داده/اطلاع خاصی را در جدول جستجو کرد. برای این کار به صورت زیر عمل کنید:

۱- باز کردن جدول در نمای Datasheet

۲- قرار دادن نشانگر در ستونی که می‌خواهید جستجو شود و انتخاب... Find از زبانه Home.

۳- پنجره محاوره‌ای Find and Replace نمایش داده خواهد شد. مقدار شرط را در جعبه متن: Find What وارد کنید.

۴- در لیست بازشونده: Look In، ناحیه موردجستجو را تعیین کنید. (کل جدول یا فیلد مشخص شده.)

۵- معیار تطابق را در لیست بازشونده Match اختیار کنید. سه انتخاب موجود است:

a. Any Part of Field: زیرمجموعه کاراکترهای هر فیلد را مطابقت می‌دهد.

b. Whole Field: کل فیلد را مطابقت می‌دهد.

c. Start of Field: شروع فیلد را مطابقت می‌دهد.

۶- از لیست بازشونده Search، می‌توانید All را برای تمام ستون‌ها، Up را برای جستجو به سمت ابتدای قبل از محل رکوردی که در آن هستیم و Down را برای جستجو به سمت انتهای بعد از محل قرارگرفتن رکورد انتخاب کنید.

۷- پس از تنظیم کل معیارها، روی دکمه Find Next کلیک نمایید. اگر بیش از یک رکورد در آن شرط صدق کند، کلیک روی دکمه Find Next را تا رسیدن سایر رکوردهای دارای شرط ادامه دهید.

آیا رکوردی با مقدار «ریاضی فیزیک» در جدول Tbl-Major وجود دارد؟

تمرین ۲-۳:

● جایگزین کردن داده‌ها: با تابع Replace، می‌توان مقدار خاص داده را با مقدار جدید جایگزین کرد.

۱- زبانه Replace را در پنجره محاوره‌ای Find and Replace انتخاب کنید.

۲- برای جستجو و انتخاب داده‌های موردنظر، به روش جستجوی فوق عمل کنید و معیارها را تنظیم نمایید. مقدار جدید را در جعبه متن: Replace With تایپ کنید.



۳- با کلیک Find Next و سپس کلیک دکمه Replace، مقدار یافته شده با مقدار جدید جایگزین می‌شود.

۴- با کلیک Replace All، تمام مقادیر دارای شرط به یک باره جایگزین می‌شوند.

۳-۶- مرتب‌سازی و فیلتر کردن داده‌ها


با امکان مرتب‌سازی می‌توان ترتیب نمایش رکوردها را تعریف کرد و با فیلتر کردن می‌توان فقط رکوردهای مطلوب و منطبق با معیار را مشاهده کرد.^۱

— مرتب‌سازی: فیلد موردنظر برای مرتب‌سازی را انتخاب کنید، از زبانه Home گزینه

Sort Ascending یا Sort Descending را انتخاب کنید (دکمه  یا ) را در نوار ابزار کلیک کنید). برای مرتب‌سازی با بیش از یک فیلد، ابتدا ستون‌های موردنظر را انتخاب کنید، سپس به صورت

۱- با مفهوم فیلتر در کتاب بسته‌های نرم‌افزاری ۲ بخش صفحه گسترده آشنا شده‌اید.

قبل عمل کنید.

— **فیلتر کردن** : رکوردهایی که دارای شرایط موردنظر باشند را نمایش می دهد. فیلد موردنظر را انتخاب کنید، روی رکورد حاوی مقدار مطلوب قرار بگیرید، از زبانه Home، گزینه Filter By Selection را انتخاب کنید یا روی نوار ابزار دکمه  را انتخاب کنید.

— **حذف فیلتر** : دکمه Remove Filter را در نوار ابزار کلیک کنید. تمام رکوردهای جدول دوباره نمایش داده می شود.

مقادیر زیر را در جدول Tbl_Major اضافه کنید.



MajorCode	MajorName
1	ریاضی و فیزیک
2	علوم تجربی
3	علوم انسانی
4	فنی و حرفه ای
5	کار و دانش
*	

مثال ۳-۵

۳-۷- حذف و اضافه ستون ها

بهترین راه افزودن و یا حذف فیلد از جدول، استفاده از نمای Table است چرا که انتخاب های متعددی در اختیار هست. در نمای Datasheet هم می توان این کار را انجام داد که مشابه حذف و اضافه ستون در نرم افزار Excel است.

یادآوری

ستونی را که می خواهید، ستون جدید در سمت چپ آن افزوده شود انتخاب کنید سپس با انتخاب Insert Column از منوی باز شده با کلیک راست ستون جدید افزوده می شود.

برای حذف می‌توان ستون مورد نظر را انتخاب و با گزینه Delete Column از منوی باز شده با کلیک راست آن را حذف کرد. زمانی که ستونی حذف یا اضافه شود و جدول ذخیره گردد تغییرات قطعی می‌شوند و قابل برگشت به وسیله Undo نخواهند بود.

در جدول Tbl-Course، فیلدهای ساعت تشکیل کلاس ProgramHour

و شماره کلاس ClassName را اضافه کنید. آن را ذخیره نمایید. **تمرین ۸-۳:**

در نمای Design این فیلدهای جدید چه خصوصیتی دارند؟

۸-۳- ایجاد ارتباط بین جدول‌ها

انواع ارتباط‌ها بین جدول‌ها را دیده‌ایم. در نظریه بانک‌های اطلاعاتی رابطه‌ای، می‌توانیم یکپارچگی ارتباط‌ها را تعریف کنیم که به Referential Integrity معروف است. در اکسس می‌توانیم ارتباط‌های بین جدول‌ها را معرفی کنیم بدون آن که کنترل یکپارچگی به وسیله بانک اعمال گردد، ولی حسن تعریف ارتباط آن است که اکسس کنترل‌هایی را روی عملیات داده‌ها اعمال می‌کند و با تعریف مجموعه قواعد هم از صحت ارتباط بین رکوردهای دو جدول مرتبط شده اطمینان می‌یابیم، هم اکسس سازگاری و یکپارچگی اطلاعات بانک را حفظ می‌کند. به‌طور مثال، هنگام حذف رکورد از جدولی که ارتباط یک - به - چند با جدولی دیگر دارد، باید رکوردهای جدول جزئیات نیز حذف گردد چرا که در غیر این صورت چون اطلاعات اصلی (master) حذف شده است، رکوردهای جدول دوم (detail) بلا تکلیف می‌ماند. برای جلوگیری از این گونه ناسازگاری‌های داده، زمان معرفی ارتباط در اکسس، یکپارچگی را نیز تعریف می‌کنیم.

*** برای ایجاد یکپارچگی شرایط زیر باید وجود داشته باشد:**

– فیلد برقرارکننده ارتباط در جدول پدر (master) کلید اصلی باشد و یا دارای اندیس منحصر به فرد (Unique) باشد.

– فیلدهای برقرارکننده در هر دو جدول، نوع داده (Data Type) یکسانی داشته باشند^۱.

– هر دو جدول در یک بانک اطلاعاتی قرار داشته باشند.

۱- برای مطالعه بیشتر به مفهوم میدان در کتاب‌های بانک اطلاعاتی مراجعه شود.

* هنگامی که یکپارچگی ارتباط فعال گردد باید موارد زیر را مدنظر قرار دهیم :
– نمی‌توانیم در فیلد کلید خارجی (فیلد مشترک در جدول فرزند (detail)) مقداری را وارد نماییم که در جدول پدر موجود نباشد.

– نمی‌توانیم از جدول پدر رکوردی را حذف کنیم که رکوردهای متناظر در جدول فرزند وجود داشته باشند. مگر آن که در تعریف یکپارچگی رابطه به اکسس مجوز برای حذف رکوردهای متناظر در جدول فرزند داده باشیم. در این صورت خود به خود رکوردهای متناظر در جدول فرزند حذف می‌شوند.

– نمی‌توانیم مقدار فیلد کلید اصلی را در جدول پدر تغییر دهیم و ویرایش نماییم، اگر رکوردهای متناظری در جدول فرزند وجود داشته باشند.



- ۱- با تعریف چه نوع ارتباطی، با حذف رکوردی در جدول پدر، رکوردهای متناظر در جدول فرزند خود به خود حذف می‌گردند؟
- ۲- چگونه با تغییر مقدار فیلد اصلی در جدول پدر، فیلد متناظر در جدول فرزند نیز ویرایش و به هنگام می‌شود؟


برای آن که کنترل یکپارچگی و صحت روابط بین جدول‌ها انجام گیرد، زمان معرفی ارتباط باید گزینه Enforce Referential Integrity را انتخاب کرده، در پنجره ظاهر شده شرایط موردنظر را تنظیم کنیم. در تمرین زیر چگونگی ایجاد رابطه و اعمال یکپارچگی را مشاهده خواهید کرد.



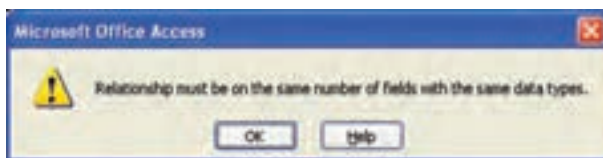
ارتباط‌های زیر را در بانک خود اضافه کنید.

مثال ۳-۶

مراحل انجام کار

- از زبانه Database Tools نوار ابزار، گزینه Relationships را انتخاب نمایید. در پنجره باز شده، جدول‌های مورد نظر را با کلیک روی  اضافه نمایید.
- مانند شکل از فیلد مشخص شده یک جدول به فیلد متناظر در جدول دیگر درگ کنید تا ارتباط ایجاد شود.
- در پنجره باز شده بدون هیچ تغییری، Create را انتخاب کنید. دقت کنید که یکپارچگی روابط در همین پنجره تعیین می‌شوند. گزینه Enforce Referential Integrity که ما آن را بدون تغییر گذاشته‌ایم.

همان‌طور در تبدیل موجودیت و ارتباطات به جدول‌ها در فصل اول دیدید، فیلد کلید اصلی از یک جدول در دیگری اضافه شده است. این ارتباطات در زمان طراحی بانک مشخص می‌شوند. در نوع ارتباط یک – به – چند، فیلد کلید اصلی جدول طرف «یک» در جدول طرف «چند» اضافه می‌شود و در جدول دوم اصطلاحاً کلید خارجی نام دارد. نام فیلد کلید خارجی می‌تواند متفاوت از نامی باشد که در جدول اول استفاده شده است ولی نوع آن‌ها الزاماً یکی است، در غیر این صورت در ایجاد یکپارچگی رابطه Referential Integrity خطای زیر (شکل ۳-۳) رخ می‌دهد.



شکل ۳-۳. پیغام خطای نوع فیلد

کنجکاوی

زمان ایجاد رابطه در محیط Relationships، درگ کردن از فیلد کلید اصلی جدول پدر به فیلد کلید خارجی جدول فرزند یا عکس این عمل یعنی درگ کردن از جدول فرزند به جدول پدر، آیا تفاوتی ایجاد می‌کند؟

خلاصه فصل

شیء جدول، داده‌های بانک را نگه می‌دارد. جدول‌های ترسیم شده روی کاغذ باید در بانک اطلاعاتی تعریف و ایجاد شوند. سه روش اصلی برای ایجاد جدول وجود دارد: Design View، Table Templates، Datasheet View و همچنین یک روش انتقال به وسیله Import Table و یا Link Table.

پس از ایجاد جدول و تعریف خصوصیات فیلدهای متناسب با طراحی انجام شده، می‌توان داده‌ها را در آن ثبت کرد. یکی از راه‌های ورود داده‌ها به جدول به کمک نمای Datasheet است که با امکانات موجود در این نما می‌توان رکوردها را اضافه کرد، داده‌های آن را ویرایش کرد، حذف کرد و یا طرح نمایش را با تغییر اندازه سطر و ستون‌ها و یا تغییر رنگ، نوع قلم، ثابت سازی ستون، پنهان‌سازی ستون و ... تغییر داد. می‌توان در این نما داده‌های موردنظر را جستجو کرد، مقدار داده را با یک مقدار جدید جایگزین نمود و یا ترتیب نمایش رکوردها را با مرتب‌سازی تغییر داد و یا رکوردها را برحسب شرایط خاص فیلتر کرد.

دیدیم عموماً جدول‌های بانک اطلاعاتی با یکدیگر ارتباط دارند. ایجاد رابطه (Relation) که در طراحی مشخص شده است با انتقال کلید اصلی به جدول دیگر به‌عنوان کلید خارجی صورت می‌گیرد. در Access ارتباط موجود می‌تواند به وسیله قواعد سازگاری کنترل شود تا ناسازگاری داده‌ای در بانک به‌وجود نیاید. این کار با تعریف Referential Integrity در محیط Relationships ایجاد می‌شود.

- ۱- روش‌های ایجاد جدول را با هم مقایسه کنید.
- ۲- مشخصه‌های انواع داده‌فیلدها را بیان کنید.
- ۳- شرایط ایجاد یکپارچگی ارتباط در جدول چیست؟
- ۴- قواعد سازگاری چه مواردی را کنترل می‌کند؟
- ۵- تفاوت انتقال جدول از بانک دیگر به کمک Import و Link چیست؟

چیست؟

فصل ۴

ایجاد و کار با فرم‌ها

فرم، واسطی است بین کاربر و پایگاه داده. در این فصل ابتدا با مفاهیم اولیه فرم و اجزای آن که به آن‌ها کنترل (Control) می‌گوییم آشنا شده، سپس ایجاد فرم و عملیات روی آن را بررسی می‌کنیم. فرم‌ها اطلاعات بانک را در قالبی جذاب نشان می‌دهند. با ایجاد فرم، کاربر، داده‌ها را آسان‌تر وارد می‌کند و امکان اشتباه نیز کاهش می‌یابد.

هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

- کنترل‌های روی فرم را ایجاد کند.
- بتواند طراحی کنترل‌های روی فرم را تغییر دهد.
- به وسیله گزینه Form Wizard فرمی را ایجاد و از گزینه‌های این ابزار استفاده کند.
- در نمای Design View کنترل‌ها را به انواع دیگر تبدیل کند و مشخصه‌های آن‌ها را تغییر دهد.
- بتواند در نمای Form View فرم را ایجاد و رکوردهایی را ثبت، ویرایش یا حذف نماید.

۴-۱- مفاهیم اولیه فرم و اشیای آن

۴-۱-۱- انواع «نوع کنترل» در ایجاد فرم‌ها

● مقید^۱: داده‌ها از منبع رکوردهای متصل شده به فرم نمایش می‌یابند، می‌توان رکوردها را ویرایش کرد، حذف کرد و یا اضافه نمود.

● نامقید^۲: به رکوردها متصل نیستند و محتوای آن‌ها با جابه‌جایی از رکوردی به دیگری تغییر نمی‌کند.

۴-۱-۲- انواع کنترل‌ها: نمونه‌هایی از این کنترل‌ها را در شکل‌های ۴-۱ و ۴-۲

می‌بینید.

جدول ۴-۱- انواع کنترل‌ها

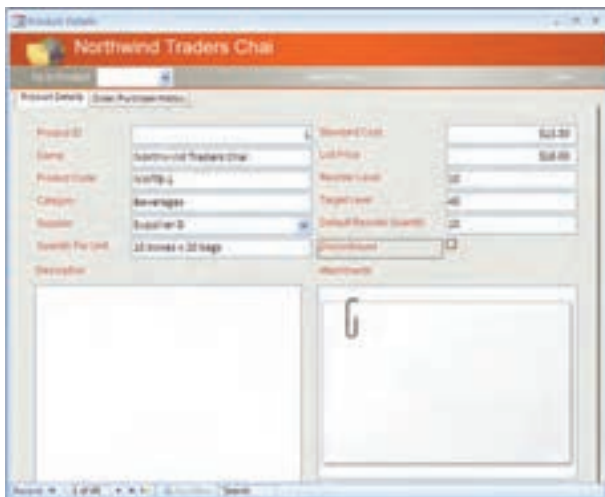
کنترل	نوع	مفهوم
Label	Unbound	توضیحی درخصوص متن هم‌جوارش ارائه می‌کند.
Text Box	Bound	داده‌ها را از منبع رکورد (Record Source) نمایش می‌دهد، می‌توان آن را ویرایش نمود و یا داده جدیدی را وارد کرد.
List Box	Bound	لیست ورودی‌های ممکن را نمایش می‌دهد.
Combo Box	Bound	لیست ورودی‌ها را نشان می‌دهد، همچنین محل متنی برای ورود متن داده به وسیله صفحه کلید نیز فراهم می‌کند. ترکیبی است از List box و Text box.
Tab Control	Unbound	جنبه سه بُعدی به فرم می‌افزاید.
Check Box	Bound	انتخاب بله/خیر را برای یک فیلد فراهم می‌کند. اگر علامت <input checked="" type="checkbox"/> داشته باشد، بله است.
Toggle Button	Bound	انتخاب بله/خیر را برای یک فیلد فراهم می‌کند. اگر فشرده شده باشد، بله است.
Option Button	Bound	انتخاب را برای یک فیلد فراهم می‌کند.
Option Group	Bound	انتخاب‌های یک فیلد را دسته‌بندی می‌کند و نمایش می‌دهد.
Bound Object Frame	Bound	داده ^۳ OLE مثل تصاویر را نمایش می‌دهد.

۱- Bound

۲- Unbound

۳- Object linking and embedding

نمایش تصویر و یا یک فیلم کوتاه که به رکوردی متصل نباشد.	Unbound	UnboundObject Frame
با آن می‌توان خطوط و مربع یا مستطیل‌ها را در فرم رسم کرد.	Unbound	Line and Rectangle
ساده‌ترین راه اجرای یک فرمان یا یک ماکرو است. با کلیک روی آن برنامه نوشته شده اجرا می‌شود.	Unbound	Command Button



شکل ۴-۱- نمونه‌ای از کنترل‌های یک فرم



شکل ۴-۲- نمونه‌ای از کنترل‌های یک فرم

۳-۱-۴- روش تغییر اندازه و جابه‌جایی کنترل‌ها : با جابه‌جایی ماوس روی کنترل، اشکال زیر ظاهر می‌شوند که هر یک عمل متفاوتی انجام می‌دهد :

جدول ۲-۴- روش تغییر اندازه و جابه‌جایی کنترل

عمل مورد انتظار	چه زمانی ظاهر می‌شود؟	شکل
با کلیک می‌توان کنترل موردنظر را انتخاب کرد.	زمانی که به کنترل انتخاب نشده اشاره کنید، نمایش داده می‌شود (همان وضعیت پیش فرض ماوس است)	
با درگ، می‌توان کنترل انتخاب شده را جابه‌جا کرد.	زمانی که به لبه کنترل انتخاب شده اشاره کنید، ظاهر می‌شود. (البته نه در زمان اشاره به مربعات تغییر اندازه)	
با درگ، می‌توان آن کنترل را جابه‌جا کرد نه به همراه سایر کنترل‌های انتخاب شده.	زمانی که به مربع بزرگ تغییر اندازه سمت چپ - بالای کنترل انتخاب شده اشاره کنید، نمایش داده می‌شود.	
با درگ می‌توان اندازه کنترل را تغییر داد.	زمانی که به مربع تغییر اندازه اشاره کنید. (البته نه مربع چپ - بالا)	

۴-۱-۴- انتخاب بیش از یک کنترل

- کلیک روی اولین کنترل (انتخاب یک کنترل) و انتخاب سایرین با $\text{Shift}+\text{click}$ روی هر یک از آن‌ها.
- درگ selection box (کلیک یک نقطه و درگ روی کنترل، به طوری که داخل مربع نقطه‌چین نمایش یافته قرار گیرد).

۴-۲- ایجاد فرم

راه‌های مختلفی برای ایجاد فرم وجود دارد. دو راه ایجاد فرم که در این جا بررسی می‌کنیم عبارت است از : Design View و Form Wizard.

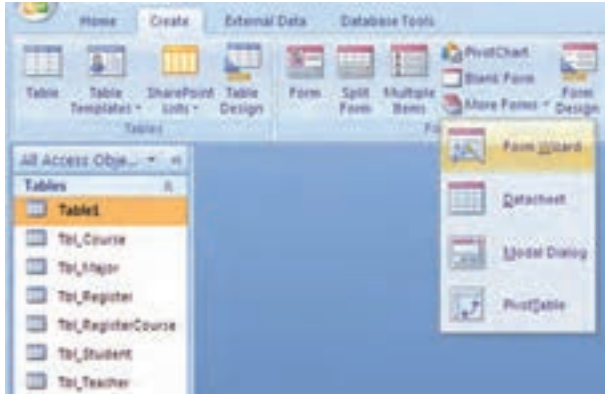
۱-۲-۴- ایجاد فرم به وسیله Form Wizard

ایجاد فرم Form Course

● کلیک روی Forms در نوار اشیاء.

● انتخاب گزینهٔ more Forms از زبانهٔ Create نوار ابزار و انتخاب گزینهٔ Form Wizard

(شکل ۳-۴).

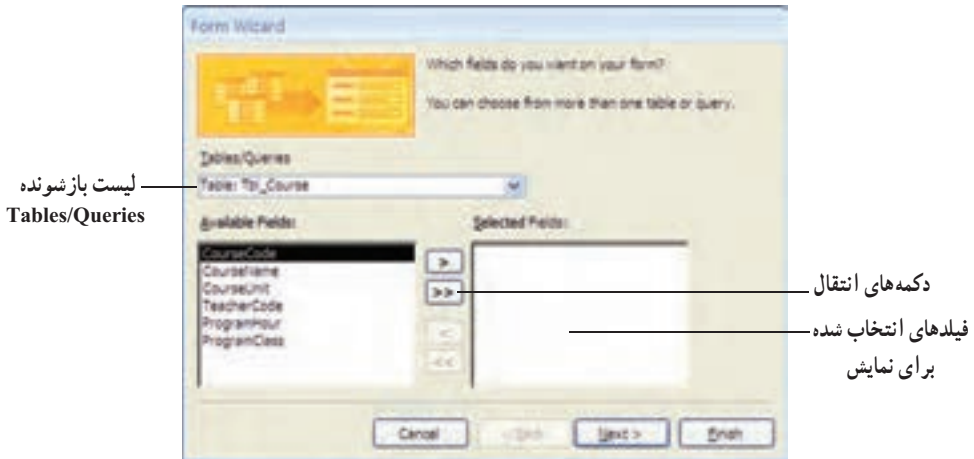


شکل ۳-۴- ایجاد فرم به وسیلهٔ Wizard

● جدول یا پرس‌وجوی موردنظر را از لیست بازشونده Tables/Queries انتخاب کنید.

جدول یا پرس‌وجو باید تمام اطلاعات موردنیاز نمایش روی فرم را دربر داشته باشد. در اینجا انتخاب

Tbl-Course (شکل ۴-۴).



شکل ۴-۴- اضافه کردن فیلدها به فرم

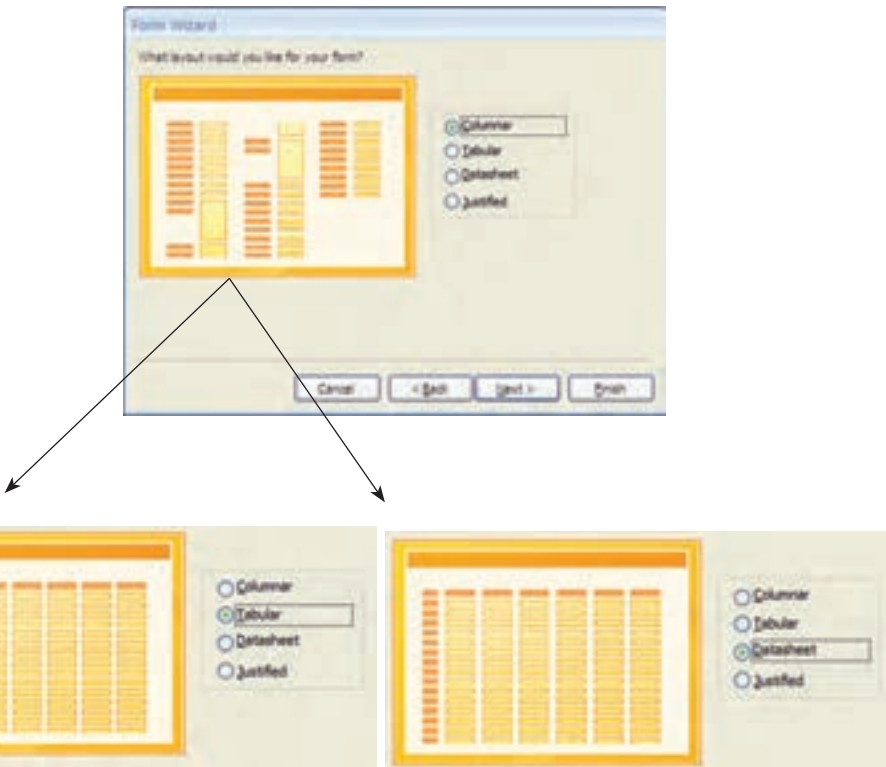
● بر روی دکمه >> کلیک کنید تا تمام فیلدهای جدول به لیست سمت راست با عنوان : Selected Fields افزوده شود. یا با کلیک روی دکمه > فیلدهای مورد نظر خود را یکی یکی اضافه کنید. در صورتی که فیلدی به اشتباه در لیست راست اضافه شده باشد با دکمه < آن را از لیست انتخابی خود حذف کنید.

● با کلیک دکمه Next پنجره انتخاب قالب فرم نمایش داده می شود که پیش فرض آن Columnar است (شکل ۵-۴).

— Columnar : هر رکورد با برچسبها و فیلدهای فرم نمایش داده می شود. شبیه یک فرم معمولی است.

— Tabular : چندین رکورد در صفحه لیست می شوند. فیلدها به صورت ستونی و رکوردها در سطرها نمایش می یابند.

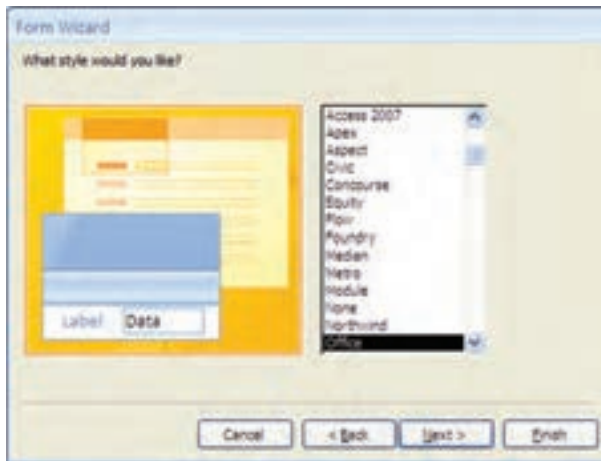
— Datasheet : رکوردها به صورت نمای datasheet جدول نشان داده می شوند.



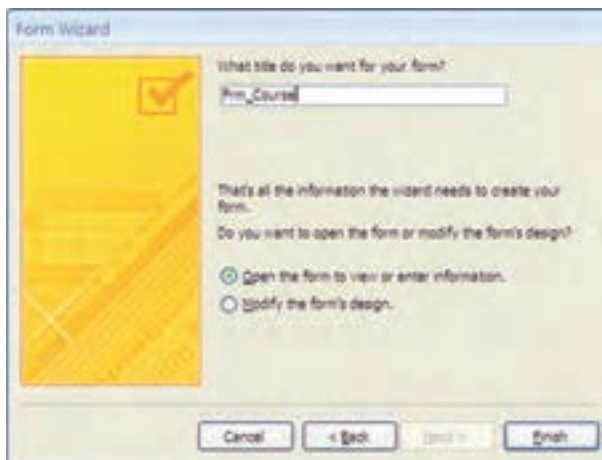
شکل ۵-۴- پنجره انتخاب قالب فرم

تفاوت قالب فرم Columnar و Justifield چیست؟

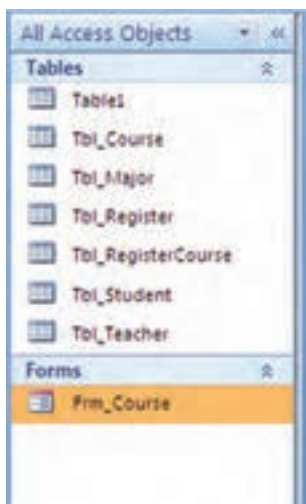
- در پنجره‌های متوالی بعدی، پیش فرض Standard را قبول کنید (شکل ۴-۶) و در پنجره آخر (شکل ۴-۷)، نام موردنظر برای فرم را وارد نمایید و دکمه Finish را کلیک کنید. فرم ساخته شده در نمای نمایشی نشان داده می‌شود.



شکل ۴-۶- پنجره انتخاب طرح



شکل ۴-۷- پنجره آخر طراحی فرم به روش Wizard



● با دکمه Finish فرم را ببندید. در این صورت فرم ایجاد شده را در پنجره اشیاء تحت Forms مشاهده خواهید کرد (شکل ۸-۴).

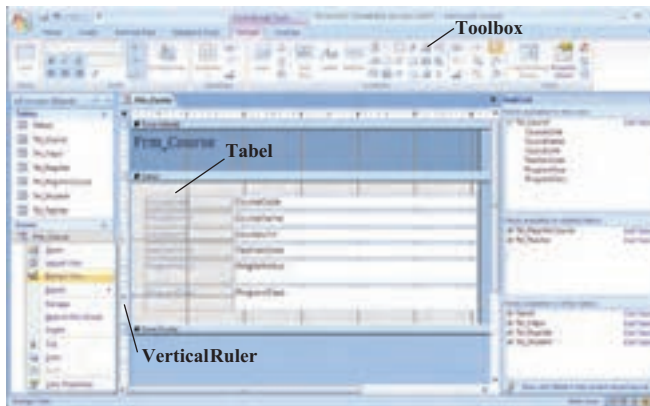
شکل ۸-۴ پنجره اشیاء فرم

به وسیله ویزارد برای جدول Tbl_Teacher فرم Frm_Teacher را ایجاد نمایید. همچنین به کمک ویزارد، فرمی برای جدول Tbl_Student ایجاد کنید. نام آن را Frm_Student قرار دهید و کنترل‌های آن را مانند زیر تنظیم نمایید.


تمرین ۱-۴:

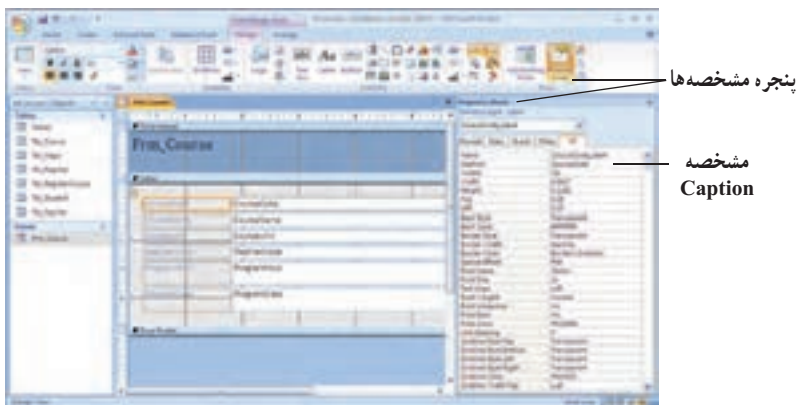
۲-۴-۲- ایجاد فرم به وسیله **Design View** : در این حالت باید با استفاده از ابزارهای

طراحی فرم که در شکل ۹-۴ می بینید، فرم را ایجاد کنید. کنترل متناظر و مناسب برای هر فیلد جدول در فرم افزوده و خصوصیات آن را تغییر می دهید. در صورت نیاز برنامه نویسی نیز لازم است.



شکل ۹-۴- ابزارهای طراحی فرم

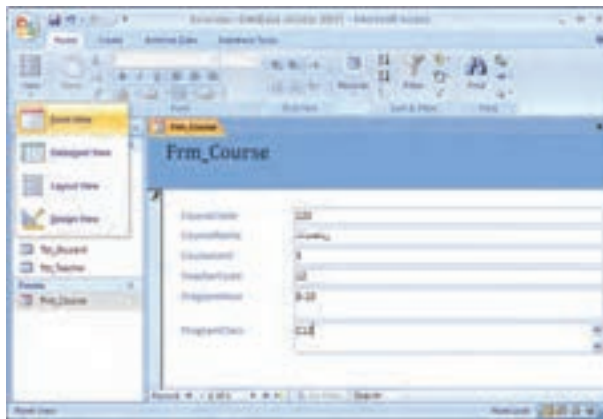
در بخش تغییر طراحی فرم (بخش ۵-۴)، اندکی بیشتر با مبحث طراحی فرم آشنا می شویم. تغییر مشخصات کنترلها: با پنجره مشخصه ها می توان مثلاً متن نمایشی در یک Label فرم را - مشخصه **Caption** - همان طور که در شکل ۱۰-۴ ملاحظه می شود، تغییر داد. در صورتی که پنجره مشخصه ها ظاهر نباشد آن را به وسیله دکمه  در نوار ابزار مشاهده کنید.



شکل ۱۰-۴- پنجره مشخصه ها

۳-۴- افزودن رکورد در فرم

پس از ایجاد فرم با نمای Form View، می‌توان اطلاعات را در جدول متناظر فرم ثبت کرد. با تکمیل اطلاعات بر روی کنترل‌های فرم، اطلاعات به صورت یک رکورد به جدول اضافه می‌شود. با کلید [Tab] از کنترلی به دیگری جابه‌جا می‌شوید. زمانی که آخرین رکورد نمایش داده می‌شود، در صورتی که روی آخرین کنترل فرم قرار داشته باشید، با کلید [Tab] کنترل‌های فرم خالی شده و اطلاعات قبلی در یک رکورد ذخیره می‌شود. با کلیک دکمه New Record در پایین فرم نیز کنترل‌های فرم خالی شده و منتظر تکمیل می‌ماند و با تکمیل فیلدها، رکورد جدید اضافه می‌شود (شکل ۴-۱۱).

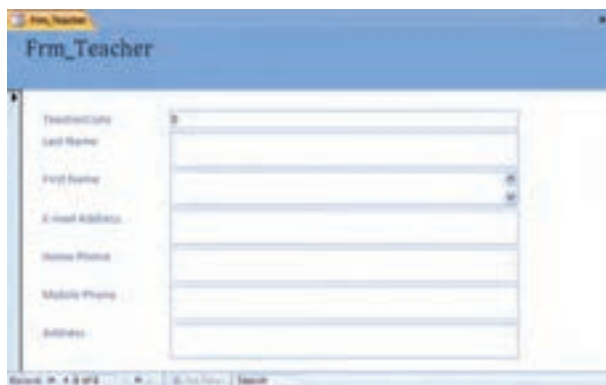



شکل ۴-۱۱- نمونه‌ای از فرم

به وسیله‌ی نمای ورود اطلاعات فرم Frm-Teacher، اطلاعات زیر را در جدول زیر ثبت می‌کنیم.

Teacher	Last Name	First Name	Email Address	Home Phone	Mobile Phone	Address
1043	محمدرضا	محمد			912-1418982	
1046	مهدی	مهدی				
1047	محمدرضا	محمدرضا		021-7795542		
1048	محمدرضا	محمدرضا			912-5554348	
1049	محمدرضا	محمدرضا				

برای این منظور روی فرم Frm-Teacher راست کلیک نموده و گزینه‌ی Layout View را انتخاب کنید.



- یک ردیف از اطلاعات جدول بالا را وارد کنید. دکمه  را کلیک کنید.
- با این کار اطلاعات ثبت می‌گردد و کنترل‌های فرم خالی شود.
- ردیف‌های دیگر را وارد و همین گونه ادامه دهید تا تمام داده‌ها ثبت گردند.

۴-۴- ویرایش فرم‌ها

همان‌طور که در بسته‌های نرم‌افزاری ۱ و ۲ دیدید در این نرم‌افزار نیز باید بتوانیم کنترل‌های روی فرم را جابه‌جا نماییم، تغییر اندازه دهیم، برحسب متناظر به کنترل را اصلاح کنیم، ظاهر نمایشی فرم را تنظیم کنیم، سربرگ و پانوشت برای فرم‌ها ایجاد کنیم و برای سهولت اعمال تغییرات محیط ویرایش فرم را تنظیم نماییم. در محیط Access به راحتی می‌توانیم انواع کنترل را به یک‌دیگر تبدیل کنیم که در محیط‌های برنامه‌نویسی دیگر به این سهولت قابل انجام نیست.

۴-۴-۱- تنظیمات محیط ویرایشی فرم

– **تنظیم شبکه نقطه‌چین Grid lines:** به‌طور پیش‌فرض، در حالت نمای Design روی فرم یک سری نقاط و خطوط به عنوان راهنما نشان داده می‌شود تا محل قرار گرفتن عناصر روی فرم به آسانی تنظیم شود. برای نمایش یا عدم نمایش این خصوصیت بر روی فرم از زبانه Arrange قسمت Show /Hide، گزینه Grid را انتخاب کنید و برای تنظیم فاصله نمایش شبکه نقطه‌چین در حالت نمای Design فرم، روی مربع انتخاب‌گر فرم دوبار کلیک نمایید تا پنجره مشخصه‌های فرم نمایان شود و یا با روش قبلی این پنجره را آشکار کنید. در زبانه Format گزینه‌های GridX و GridY را تنظیم کنید.

گزینه GridX و GridY چه مقیاسی دارند و هر یک کدام بعد مربع یعنی فاصله عمودی یا افقی نمایش شبکه نقطه چین را تغییر می دهند؟

گزینه Snap to Grid: با انتخاب این گزینه، کنترل های روی فرم حتماً باید با گوشه های شبکه نقطه چین تنظیم شوند. در این صورت زمان جابه جایی کنترل، حرکت آن به صورت پرشی خواهد شد. با عدم انتخاب این گزینه، آزادانه می توانید اشیاء فرم را در کوچک ترین فاصله و بین فاصله مربع های شبکه نقطه چین جابه جا نمایید.

۲-۴-۴- روش های ویرایش طراحی فرم

به صورت پیش فرض در نمای Design، کنترل های فرم به صورت ثابت و به هم چسبیده قرار دارند، برای اینکه بتوان روی هر کنترل به تنهایی تنظیمات نمایشی مانند تغییر اندازه و یا جابه جایی اعمال کرد می بایست از زبانه Arrange نوار ابزار، گزینه Remove را برای کنترل یا کنترل های مورد نظر انتخاب نمود.

تغییر اندازه کنترل های روی فرم: با درگ کردن به وسیله ماوس در گوشه و یا ضلع های یک شیء می توان اندازه آن را تغییر داد.

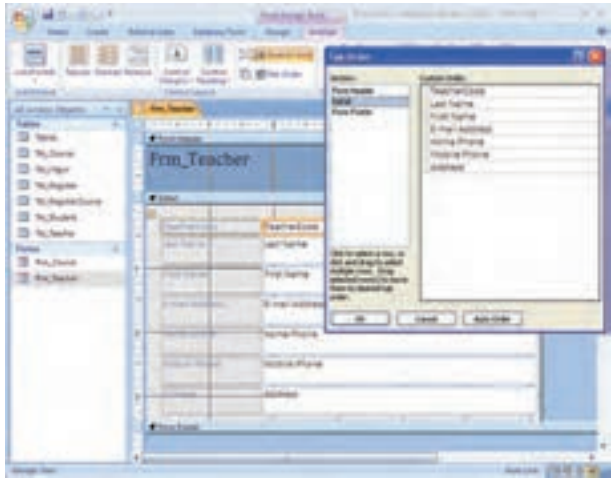
تغییر نوع کنترل: به آسانی می توان نوع کنترل های فرم را بدون نیاز به حذف و ایجاد مجدد، تغییر داد. مثلاً از Text box به Combo box تبدیل کرد. روی شیء کلیک راست کرده و گزینه Change To را انتخاب کنید و از لیست انواع مختلف کنترل ها، نوع دیگری از کنترل را انتخاب کنید.

جابه جایی برچسب و کنترل و تغییر اندازه برچسب: در Access هر شیء فرم و برچسب متناظر آن به هم متصل هستند و با جابه جایی هر یک، دیگری نیز جابه جا می شود. برای تغییر محل شیء و برچسب آن به صورت مجزا، روی علامت مربع بزرگ در گوشه بالای چپ کنترل برچسب کلیک و به محل مورد نظر درگ کنید.



گزینه‌های Stacked، Tabular Arrange در زبانۀ کاری انجام می‌دهند؟

— ترتیب حرکت روی کنترل‌ها در زمان نمایش فرم **Tab Order** : با انتخاب گزینه Tab Order در زبانۀ Arrange می‌توانید، ترتیب جابه‌جایی روی کنترل‌ها به وسیله کلید [TAB] را در نمای ورود اطلاعات فرم مشخص کنید. در پنجره ظاهر شده با درگ کردن لیست نام کنترل و جابه‌جایی آن‌ها ترتیب مناسب را تنظیم کنید (شکل ۱۲-۴).



شکل ۱۲-۴— تغییر ترتیب جابه‌جایی روی کنترل‌ها به وسیله کلید [TAB]

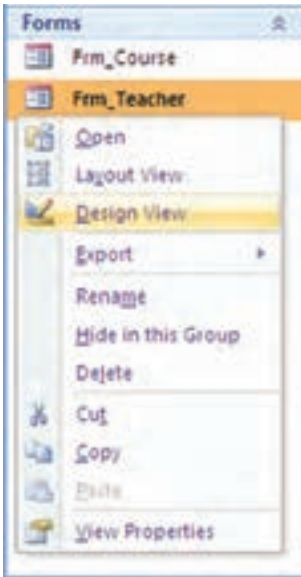
— **ظاهر فرم^۱** : رنگ زمینه فرم با مشخصه Back color در پنجره مشخصه‌ها قابل تنظیم است. رنگ هر یک از کنترل‌های فرم به‌طور جداگانه به‌وسیله انتخاب رنگ موردنظر از جعبه رنگ Fore Color در پنجره مشخصه‌ها قابل تغییر است. نوع قلم و اندازه آن، سایر موارد مربوط به نوع قلم و خطوط اطراف شیء به کمک گزینه‌های موجود در قسمت Controls زبانۀ Design قابل تغییر و تنظیم است.

— **سربرگ و پانویس فرم^۲** : سربرگ و پانویس فقط در زمان چاپ فرم ظاهر می‌شوند. با

۱_ Form Appearance

۲_ Page Header and Footer

کلیک راست روی فرم و انتخاب گزینه Page Header/Footer می‌توانید به این بخش دسترسی داشته باشید. نمایش شماره صفحه نیز می‌تواند به این بخش‌ها اضافه شود، این کار به وسیله قسمت Controls زبانه Design و گزینه Page Numbers ... انجام می‌شود. تاریخ و ساعت نیز از همین منو و به وسیله گزینه Date and Time اضافه می‌شوند. برای پنهان ساختن این بخش‌ها در نمای Design با راست کلیک روی فرم گزینه Page Header/Footer را مجدداً انتخاب کنید.

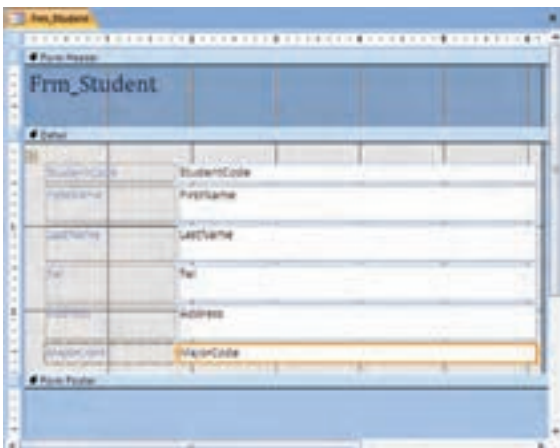


شکل ۴-۱۳- نوار ابزار نماها

۴-۵- تغییر طراحی فرم (Design)

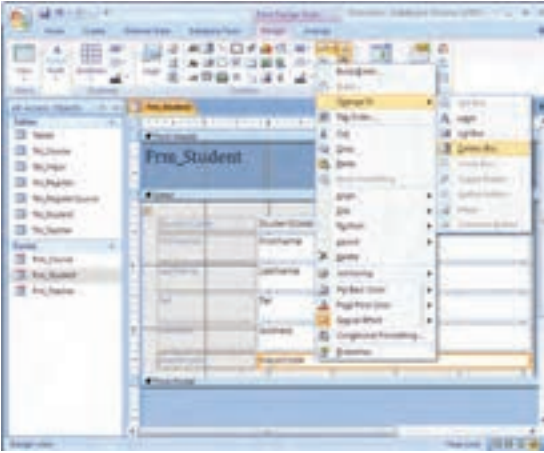
برای تغییر طراحی فرم، ابتدا فرم موردنظر را انتخاب کرده و سپس از نوار ابزار نماها (شکل ۴-۱۳)، گزینه Design View را انتخاب کنید. پنجره نمای طراحی نمایش داده می‌شود.


در ادامه با یک مثال طراحی فرم را تغییر می‌دهیم.

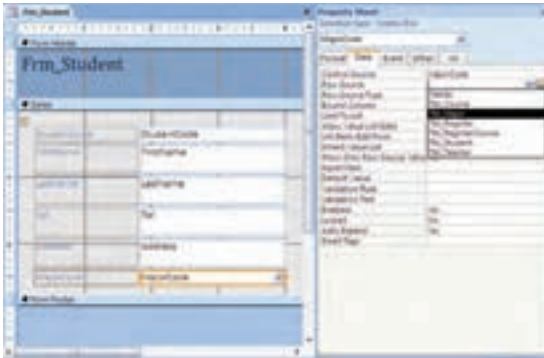


فرم Frm-Student ایجاد شده به وسیله ویزارد را در نظر بگیرید. می‌خواهیم طراحی آن را بهبود دهیم. می‌خواهیم Text Box مربوط به MajorCode را به Combo Box تبدیل کنیم.

– ابتدا جعبه متنی مربوط به MajorCode را انتخاب، کلیک سمت راست ماوس را فشار دهید و گزینه Combo Box را از منوی Change To انتخاب کنید (شکل روبه‌رو).

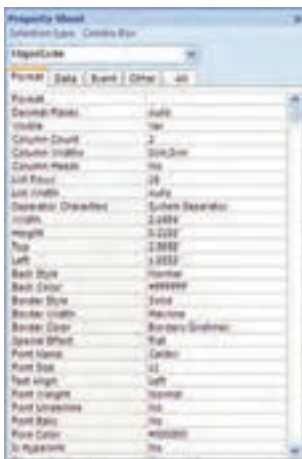


– حال دوباره روی کنترل کلیک سمت راست کرده، گزینه Properties را انتخاب کنید و یا از نوار ابزار گزینه  را کلیک کنید تا پنجره مشخصه‌ها باز شود.



– از زبانه Data در پنجره مشخصه‌ها، مقدار Row Source را Tbl-Major قرار دهید (شکل روبه‌رو). برای این کار، می‌توانید از علامت فلش سمت راست استفاده کرده و از لیست نمایش یافته جدول موردنظر را انتخاب نمایید.

– در زبانه Format، مشخصه Column Count را ۲ و خصوصیت Column Width را مقدار 0cm;3cm قرار دهید. در زمان اجرا برای انتخاب فقط مقادیر موجود در این جدول‌ها خصوصیت Limit to list را در زبانه Data، Yes قرار دهید و فرم را ذخیره نمایید.



به کمک فرم ایجاد شده اطلاعات زیر را در جدول ذخیره نمایید.
البته MajorCode به جای MajorName در جدول ثبت می‌شود.

تمرین ۲-۴:



Tbl-Student:

StudentCode	StudentName	StudentTel	StudentAddress	MajorName
1001	علی	02123456789	محل دانشگاه تهران - تهران	مهندسی و برق
1002	علی	02123456789	محل دانشگاه تهران - تهران	مهندسی و برق
1003	علی	02123456789	محل دانشگاه تهران - تهران	مهندسی و برق
1004	علی	02123456789	محل دانشگاه تهران - تهران	مهندسی و برق

به کمک ویزارد فرم درس Frm-Course را ایجاد کنید و به روش مشابه بالا، جعبه متن TeacherCode را به Combo Box تبدیل کنید.

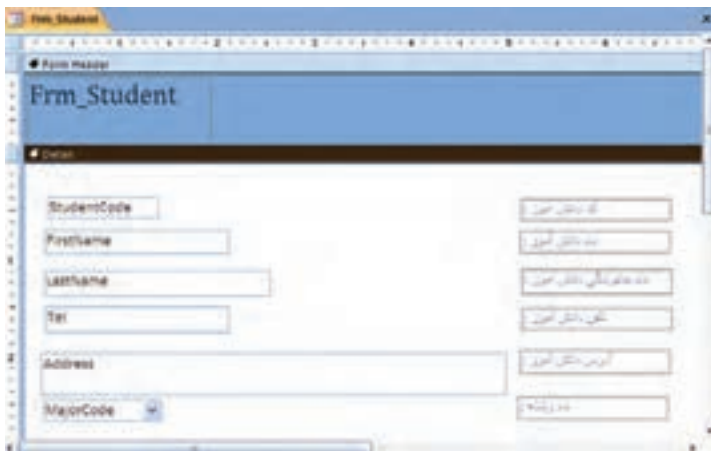
تمرین ۳-۴:



نمای طراحی فرم دانش‌آموز ایجاد شده را به صورت فارسی تبدیل کنید.

راهنمایی:


- عنوان برجسب‌ها را به فارسی تبدیل کنید. ابتدا برجسب موردنظر را انتخاب کنید و دوبار روی آن کلیک نمایید تا در حالت تغییر متن قرار گیرد و بتوانید عنوان فارسی را تایپ کنید و یا از گزینه Caption در زبانه Format پنجره مشخصه‌ها استفاده کنید.
- با کمک مربع بزرگ جابه‌جایی برجسب، آن‌ها را به سمت راست فرم منتقل کنید.

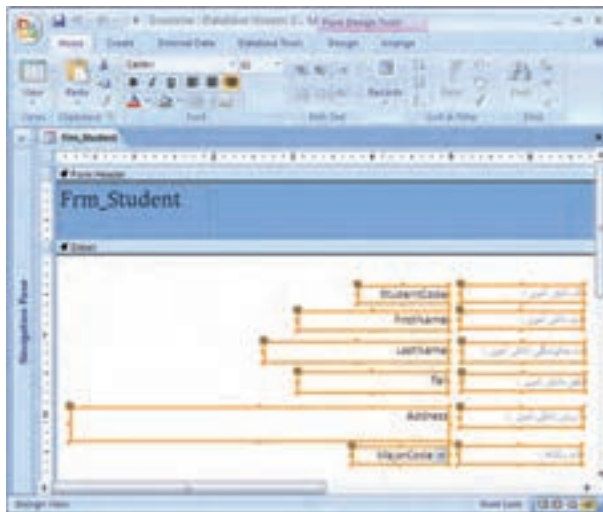


مثال ۳-۴

– با کمک مربع بزرگ جابه‌جایی متن، مناسب با محل برچسب آن‌ها را تنظیم نمایید.



– چیدمان متن را با کمک دکمه‌های  در زبانه Home نوار ابزار راست‌چین قرار دهید. یا از گزینه Text Align در زبانه Format پنجره مشخصه‌ها این کار را انجام دهید.

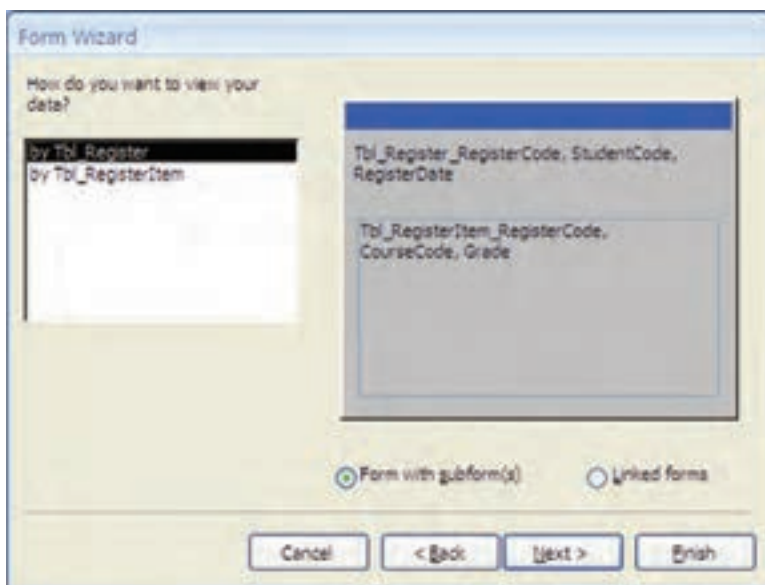


– کنترل‌های فرم را با استفاده از گزینه Select All قسمت Controls زبانه Design انتخاب و با کمک کلیدهای جهتی، آن‌ها را به سمت راست فرم جابه‌جا نمایید و فاصله ایجاد شده را تنظیم کنید.

– اندازه فرم را تنظیم کنید.

۴-۶- ایجاد فرم جزئیات در فرم اصلی (Sub Form)

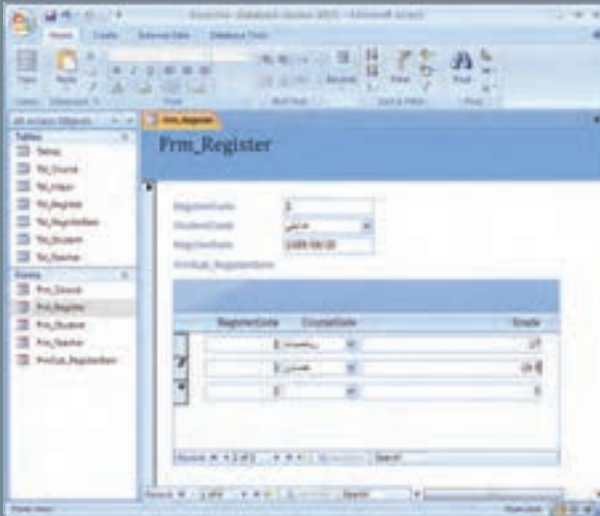
می‌خواهیم فرم Frm_RegisterItem و Frm_Register را به کمک Wizard ایجاد کنیم. برای این کار در پنجره Wizard، ابتدا در قسمت Tables/Queries، جدول Tbl_Register را انتخاب کرده، با کلیک دکمه >> تمام فیلدها را به لیست انتخابی منتقل کنید. سپس در قسمت Tables/Queries جدول Tbl_RegisterItem را انتخاب و مجدداً با دکمه >> تمام فیلدهای آن را انتخاب کنید. دکمه Next را بزنید. مجدداً Next و از پنجره باز شده Tabular را انتخاب کنید. در آخرین مرحله نام دو فرم را مقادیر Frm_RegisterItem و Frm_Register قرار دهید. با دکمه Finish شکل زیر ظاهر می‌شود.



شکل ۴-۱۴

همان‌طور که مشاهده می‌شود، این نوع فرم شامل زیر فرم است.

تمرین ۴-۴: ✓ مانند تمرین‌های قبل Text Box مربوط به StudentCode و CourseCode را به Combo Box تبدیل کنید (شکل زیر).



و با اجرای فرم داده‌های زیر را ثبت نمایید.

RegisterCode	StudentName	RegisterDate	CourseName	Grade
1	رضا موزایی	۱۳۸۵/۹/۱۹	مکتبہ	۱۹
۲	سپاس هدایی	۱۳۸۵/۹/۱۰	زبانسی	۱۹
۳	سپاس هدایی	۱۳۸۵/۹/۱۰	مضامین	۱۹
۴	سپاس هدایی	۱۳۸۵/۹/۱۱	مطرحات دینی	۱۹

با کمک همین امکانات ذکر شده، می‌توانید فرم‌های موردنظر خود را ایجاد کنید. به‌خاطر داشته باشید که برای ایجاد فرم‌های پیچیده‌تر و هوشمندتر و واردشده‌کاربر را کنترل کنند، داده‌ها را برای ذخیره آماده سازند و ... باید برنامه‌نویسی کنید. در این صورت زبان برنامه‌نویسی Visual Basic به شما کمک خواهد کرد، چرا که بسیاری از دستورات آن در محیط برنامه‌نویسی Access نیز قابل اجرا و استفاده است.

خلاصه فصل

فرم یکی از ابزارهای سودمند برای ثبت داده‌ها در جدول است. فرم‌ها پنجره‌هایی شامل کنترل‌ها هستند که کاربر می‌تواند در آن‌ها مقدار فیلد را تایپ کند یا از لیست نمایش یافته موردی را انتخاب کند و یا از کنترل‌های دیگر به آسانی داده‌ها را وارد نماید تا به صورت یک رکورد در جدول متناظر فرم ثبت شود.

با کمک ابزار جادویی Form Wizard می‌توان یک فرم ایجاد کرد. طراحی این فرم در نمای Design View قابل تغییر است. اگر با زبان برنامه‌نویسی Visual Basic و محیط آن آشنا باشید، ایجاد یک فرم در این دو محیط مشابه است. کنترل‌های دو محیط مشخصه‌های متناظری دارند که ما از ذکر آن‌ها در این‌جا خودداری می‌کنیم.

پس از ایجاد فرم با کمک امکانات نمای Form View می‌توانیم رکوردهایی را در جدول متناظر ثبت، ویرایش و یا حذف نماییم.

خودآزمایی

- ۱- کنترل مقید و نامقید چه تفاوتی دارند و چگونه تعریف می‌شوند؟
- ۲- مراحل ایجاد فرم به وسیله Form Wizard را توضیح دهید.
- ۳- امکانات نمایی Design View (نوار ابزار، جعبه ابزار، لیست فیلدها و ...) را به اختصار توضیح دهید.
- ۴- نکات مقید در ویرایش فرم‌ها را به اختصار بیان کنید.
- ۵- در صورتی که نخواهید از Form Wizard استفاده کنید، آیا می‌توانید Combo Box ای روی فرم ایجاد کنید که مقادیر جدول Tbl-Degree را نشان دهد؟

(راهنمایی: مشابه همان تغییراتی که در تبدیل نوع کنترل از Text Box

به Combo Box دیدید.)

فصل ۵

ایجاد و کار با پرس وجوها

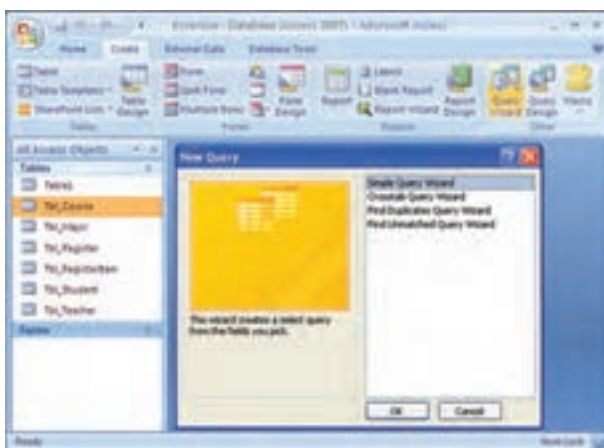
در اکسس جدول، قسمت اصلی نکه‌دارنده داده‌ها است و پرس‌وجو برای بازیافت اطلاعات از این جدول‌ها دارای اهمیت است. پرس‌وجوها انواع مختلف دارند ولی معمول‌ترین آن، پرس‌وجویی است که برای یافتن زیرمجموعه‌ای از رکوردهای دارای شرط یا شرایط خاص، استفاده می‌شود. به این نوع پرس‌وجو Select گفته می‌شود. پرس‌وجوهای دیگر، برای انجام عملیات روی مقادیر داده‌ها استفاده می‌شوند. همان‌طور که پیش‌تر گفتیم، برای نمایش داده‌ها در فرم‌ها و گزارش‌ها معمولاً از پرس‌وجوهایی از نوع Select استفاده می‌شود.

اهداف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

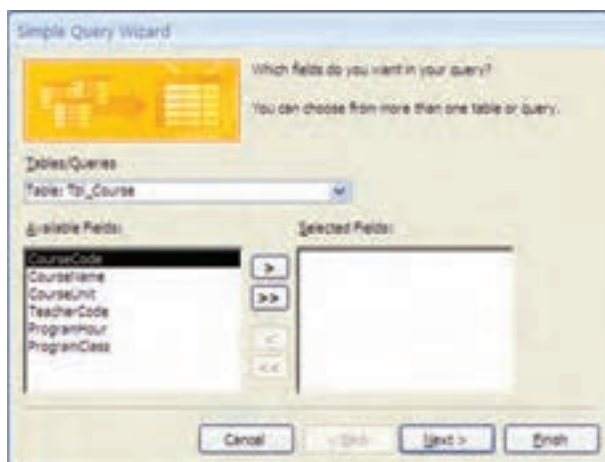
- به‌وسیله گزینه Query Wizard یک پرس‌وجوی بازیافت اطلاعات Select ایجاد کند.
- به‌وسیله Design View، پرس‌وجو ایجاد کند و یا پرس‌وجوی موجود را تغییر دهد.
- نتیجه پرس‌وجو را مرتب کند.
- پرس‌وجوی شرطی ایجاد کند. برای محدود کردن رکوردهای نتیجه شرط اعمال کند.
- فیلدهای غیرجدولی (محاسباتی) به پرس‌وجو اضافه کند و از Expression Builder برای این منظور استفاده کند.
- با پرس‌وجوهای عملیاتی آشنا شده، آن‌ها را ایجاد کند.
- با زبان ساخت یافته SQL آشنا شود.

۵-۱- ایجاد پرس و جو به وسیله Simple Query Wizard

● گزینه Query Wizard را از زبانه Create نوار ابزار انتخاب نمایید (شکل ۵-۱).
Simple Query Wizard را انتخاب کنید (شکل ۵-۱). این پنجره مشابه پنجره انتخاب فیلد فرم است. در لیست Tables/Queries جدول مورد نظر را انتخاب کنید و از لیست فیلدهای در دسترس در : Available Fields، فیلد مورد نظر خود برای ایجاد پرس و جو را با دکمه‌های >> یا > به فیلدهای انتخاب شده : Selected Fields اضافه کنید. می‌توانید جدول دیگری را انتخاب و فیلدهای مربوط از آن را نیز به لیست فیلدهای انتخابی اضافه کنید. پس از آن با کلیک Next، نام پرس و جو را وارد کرده، Finish نمایید.



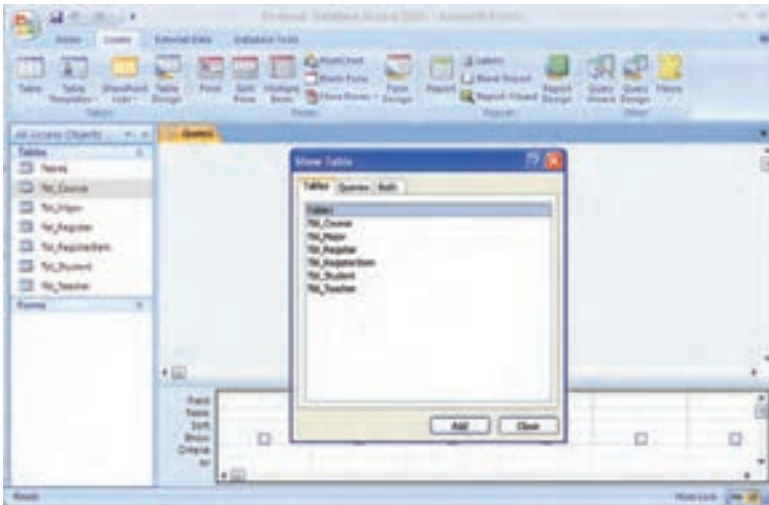
شکل ۵-۱- پنجره انتخاب ایجاد پرس و جو



شکل ۵-۲- انتخاب فیلدهای مورد نیاز برای نمایش در پرس و جو

۵-۲- ایجاد پرس و جو به وسیله Design View

● گزینه Query Design را از زبانه Create نوار ابزار انتخاب نمایید. پنجره زیر باز می شود (شکل ۵-۳). با انجام مراحل بعدی می توانید جدول ها، فیلدها و شرایط مورد نظر پرس و جو را ایجاد نمایید.

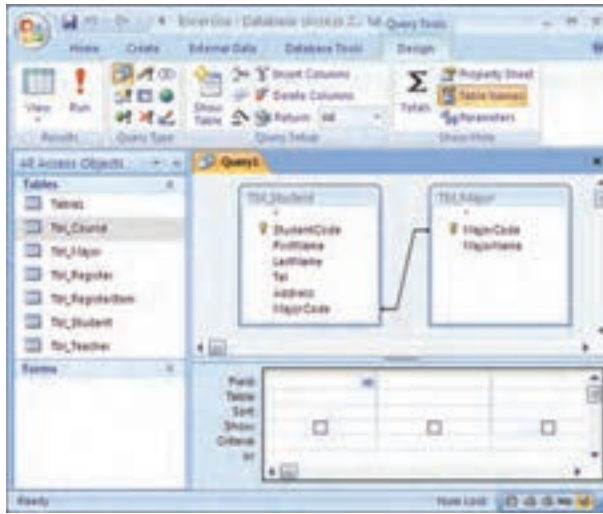


شکل ۵-۳- پنجره انتخاب جدول در طراحی پرس و جو

۵-۳- استفاده از جدول در پرس و جو

با انتخاب Query Design، کادر محاوره ای با عنوان Show Table نمایان می شود (شکل ۵-۳). در این کادر امکان انتخاب جدول، پرس و جوهای ایجاد شده و یا ترکیبی از آنها وجود دارد. در واقع می توانید از جدول های تعریف شده و یا سایر پرس و جوهای موجود برای ایجاد یک پرس و جوی جدید استفاده کنید.

با دو بار کلیک روی نام جدول یا انتخاب جدول و کلیک دکمه Add، آن را به پرس و جو اضافه می کنید. با استفاده از Shift می توانید جدول های مجاور و با استفاده از Ctrl می توانید جدول های غیر هم جوار را همزمان انتخاب نموده و با کلیک Add آنها را به پرس و جو اضافه کنید. پس از افزودن جدول ها و پرس و جوهای مورد نظر پنجره محاوره ای را با کلیک Close ببندید تا پنجره اصلی طراحی پرس و جو (Query Design) (شکل ۵-۴) ظاهر شود.

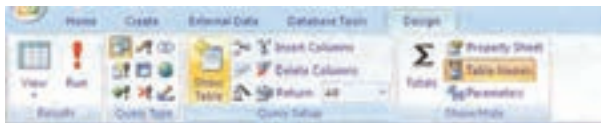


شکل ۴-۵- پنجره اصلی طراحی پرس و جو



آیا پس از بستن پنجره انتخاب جدول، راهی برای افزودن جدول جدید به پرس و جو وجود دارد؟

راهنمایی: پرس و جو را در حالت Design باز کرده و با دکمه  Show Table جدول‌های جدید مورد نیاز را اضافه کنید.



۴-۵- انتخاب فیلدهای جدول‌ها / پرس و جوهای اضافه شده مورد نیاز پرس و جوی جدید

پس از افزودن جدول یا جدول‌های مورد نظر و با پرس و جوهای قبلی، باید فیلدهای مورد نیاز را معین کنیم.

همان‌طور که در شکل ۴-۵ مشاهده می‌شود، دو قسمت در پنجره طراحی پرس و جو قابل توجه است:

- نیمه بالایی: در این قسمت جدول‌ها یا پرس و جوهای نمایش داده می‌شوند که در مرحله قبل اضافه کرده‌ایم و می‌خواهیم از آن‌ها در ایجاد پرس و جوی جدید استفاده کنیم.
- نیمه پایینی: فیلدهایی که می‌خواهیم در خروجی پرس و جو مشاهده شوند در این قسمت

معین می‌شود.

برای انتخاب فیلدها از یکی از راه‌های زیر استفاده می‌کنیم:


- ۱- دوبار کلیک روی نام فیلد در نیمه بالایی.
- ۲- درگ کردن فیلد از نیمه بالایی به نیمه پایینی.
- ۳- استفاده از Shift یا Ctrl.
- ۴- دوبار کلیک روی علامت * در لیست فیلدها برای انتخاب همه فیلدها.
- ۵- استفاده از لیست بازشوی Field.

کنجکاو

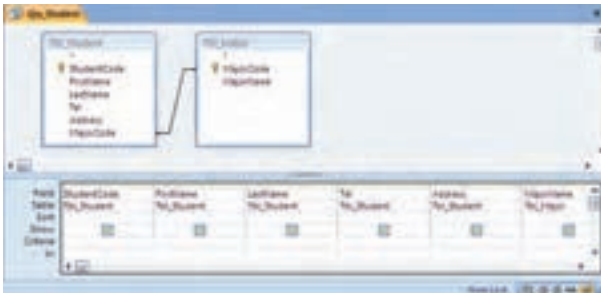
- ۱- دو فیلد نمونه از یک جدول را با دوبار کلیک به نیمه پایینی اضافه کنید. پس از آن علامت * را انتخاب کنید. به وسیله * تمام فیلدها در خروجی ظاهر می‌شوند. دو فیلد قبلی نیز فیلدهای مشابهی در خروجی دارند، این فیلدها چگونه و با چه نامی نمایش داده می‌شوند؟
- ۲- چگونه از Shift یا Ctrl برای انتخاب فیلدها استفاده می‌شود؟

پس از ایجاد پرس‌وجو، آن را با یک نام ذخیره کنید. یکی از ساده‌ترین راه‌های اجرای پرس‌وجو



استفاده از علامت  در نوار ابزار است (دکمه Run). نتیجه پرس‌وجو به صورت جدولی نمایش داده می‌شود.

دو جدول Tbl-Student و Tbl-Major را به پرس‌وجو اضافه کنید.



فیلدهای زیر را انتخاب نمایید و نام پرس‌وجو را Qry-Student بگذارید.

مثال ۱-۵

با اجرای آن، نتیجه را به صورت شکل زیر خواهید دید.

StudentCode	FirstName	LastName	Tel	Address	Signatures
00001	علی	محمد	021-4567890	تهران، خیابان ولیعصر، پلاک ۱۸	محمد علی محمدی
00002	سید	مجتهد	021-4567891	تهران، خیابان ولیعصر، پلاک ۱۸	سید مجتهد حسینی
00003	علیرضا	محمد	021-4567892	تهران، خیابان ولیعصر، پلاک ۱۸	علیرضا محمدی
00004	علیرضا	محمد	021-4567893	تهران، خیابان ولیعصر، پلاک ۱۸	علیرضا محمدی

۵-۵- حذف فیلد از پرسوجو

برای حذف فیلد از پرسوجو، ستون متناظر را از نیمه پایینی پنجره طراحی پرسوجو انتخاب و سپس دکمه Delete را می‌زنیم. به این ترتیب فیلد مذکور از نتیجه پرسوجو حذف می‌شود.

در پرسوجوی Qry-Student، فیلد شماره تلفن را حذف کنید.
 - مطابق شکل زیر ستون StudentTel را انتخاب کنید، سپس دکمه delete صفحه کلید را فشار دهید.



کنجکاوی

بدون حذف فیلد به روش بالا نیز می‌توان کاری کرد که فیلد در نتیجه خروجی نمایش داده نشود. به چه صورتی این کار انجام می‌شود؟
راهنمایی: علامت Checkbox ستون موردنظر در نیمه پایینی در سطر Show را حذف کنید یا ستون موردنظر را در پنجره نتیجه و خروجی پنهان کنید.



۱- هنگامی که در نمای Design هستیم، می‌توانیم با انتخاب یک ستون و درگ کردن آن محل قرار گرفتن فیلد در پنجره خروجی نتیجه را تغییر دهیم.

۲- برای ذخیره‌سازی پرس‌وجو از Save استفاده می‌کنیم. Access به صورت پیش‌فرض نام‌های Query۱، Query۲ و... را پیشنهاد می‌کند. بهتر است نام مناسبی برای پرس‌وجو انتخاب شود. مثلاً ابتدای نام پرس‌وجو را با Qry شروع کنید.

۶-۵- مرتب‌سازی نتیجه پرس‌وجو

پس از اجرای پرس‌وجو، رکوردهای نتیجه به ترتیب معینی ظاهر نمی‌شوند. اگر بخواهیم نتیجه بر اساس صعودی یا نزولی مقادیر یک فیلد ظاهر شود در ستون فیلد موردنظر در نیمه پایینی و در سطر Sort مقدار Ascending (صعودی) و یا Descending (نزولی) را انتخاب کنید.

در پرس‌وجوی Qry-Student، نتیجه پرس‌وجو را بر اساس صعودی مقادیر نام خانوادگی و در صورت تساوی نام خانوادگی بر اساس صعودی نام دانش‌آموز مرتب کنید.

- برای این کار می‌توانیم مطابق شکل زیر در ستون فیلد StudentLName در سطر Sort مقدار Ascending را برگزینیم و در ستون فیلد StudentFName در سطر



مثال ۳-۵

۱- Sort در بسته‌های نرم‌افزاری (۲) آشنا شده‌اید.

Sort مقدار Ascending را انتخاب می‌کنیم. در این صورت همان‌طور که در نتیجه شکل روبه‌رو می‌بینید، بین دو «یغمایی» رکوردی که فیلد نام آن کوچک‌تر باشد یعنی «علیرضا» بالاتر قرار می‌گیرد.

– نتیجه به صورت زیر خواهد بود :

StudentID	Firstname	Lastname	address	Stoghtname
00001	علیرضا	علیرضا	علیرضا علیرضا علیرضا، تهران ۱۱۱۱۱	علیرضا
00002	علیرضا	علیرضا	علیرضا علیرضا علیرضا، تهران ۱۱۱۱۱	علیرضا
00003	علیرضا	علیرضا	علیرضا علیرضا علیرضا، تهران ۱۱۱۱۱	علیرضا
00004	علیرضا	علیرضا	علیرضا علیرضا علیرضا، تهران ۱۱۱۱۱	علیرضا

– اگر ستون فیلد First Name را سمت چپ ستون فیلد Last Name قرار

دهیم، نتیجه به صورت زیر خواهد بود :

StudentID	Firstname	Lastname	address	Stoghtname
00001	علیرضا	علیرضا	علیرضا علیرضا علیرضا، تهران ۱۱۱۱۱	علیرضا
00002	علیرضا	علیرضا	علیرضا علیرضا علیرضا، تهران ۱۱۱۱۱	علیرضا
00003	علیرضا	علیرضا	علیرضا علیرضا علیرضا، تهران ۱۱۱۱۱	علیرضا
00004	علیرضا	علیرضا	علیرضا علیرضا علیرضا، تهران ۱۱۱۱۱	علیرضا



برای مرتب‌سازی با بیش از یک ستون باید ابتدا ستون‌های موردنظر را در نیمه پایینی پنجره طراحی از چپ به راست و براساس نظم موردنظر قرار دهیم. مثلاً در مثال بالا ستون فیلد First Name سمت چپ ستون فیلد Last Name قرار گرفته باشد. همان‌طور که قبلاً گفتیم، محل قرار گرفتن ستون فیلد با درگ کردن جابه‌جا می‌شود. همیشه روش مرتب‌سازی به ترتیب قرار گرفتن فیلدها، از سمت چپ به راست است.

۵-۷- معیارها' در پرس‌وجو

با استفاده از ردیف Criteria در نیمه پایینی، می‌توانیم شرط‌هایی را به پرس‌وجو اعمال کنیم. در شرط می‌توان از عملگرهای مختلفی استفاده کرد که در جدول ۵-۱ قابل مشاهده است.

جدول ۱-۵- عملگرهای شرطی

مثال	مفهوم	عملگر
= "یغمایی"	مساوی	=
<20	کوچک‌تر از	<
>20	بزرگ‌تر از	>
<= 20	کوچک‌تر یا مساوی	<=
>= 20	بزرگ‌تر یا مساوی	>=
<>20	نامساوی	<>
"هدایتی" and "یغمایی"	و (شرط‌ها باید درست باشند)	And
"هدایتی" or "یغمایی"	یا (حداقل یکی از شرط‌ها باید درست باشد)	Or
Like "محمد"	مانند	Like
Between 5 and 20	بین [دو مقدار]	Between
In ("هدایتی"، "یغمایی")	در (مانند Or)	In

۸-۵- استفاده از تاریخ در معیارها

همان‌طور که دیدیم می‌توانیم از عملگرها در پرس‌وجو استفاده کنیم. به غیر از عملگرها از توابع مثل تابع‌های مربوط به تاریخ نیز می‌توان استفاده کرد. در جدول ۲-۵ عبارت‌های مجاز تاریخ را می‌توانید مشاهده کنید.

جدول ۲-۵- تابع‌های مربوط به تاریخ

مثال	مفهوم	عبارت
<> Date()	تاریخ جاری سیستم به میلادی	Date ()
مقدار فیلد مثلاً ثبت نام برابر تاریخ امروز نباشد.		
Day[RegDate] >= 15	روز	Day (Date)
تاریخ ثبت نام از نیمه ماه گذشته باشد.		

\geq Month [Date] - 2 مقدار فیلد مثلاً ثبت نام از دو ماه قبل به این طرف باشد.	ماه	Month(Date)																						
Year[RegDate] = 1385 سال ثبت نام 1385 باشد.	سال	Year(Date)																						
WeekDay[RegDate] = 1 روز ثبت نام، شنبه باشد.	روز هفته	Weekday(Date)																						
Between#1/1/98#and #12/31/98# تاریخ ثبت نام بین این دو باشد.	محدوده ای از تاریخ	Between Date And Date																						
DatePart“m”[RegDate] = 5 ماه ثبت نام برابر ماه مه میلادی (حدود اردیبهشت ماه) باشد.	قسمت معینی از تاریخ مقادیر Interval عبارت است از:	DatePart(Interval,Date)																						
	<table border="1"> <thead> <tr> <th>مقدار</th> <th>توضیحات</th> </tr> </thead> <tbody> <tr> <td>yyyy</td> <td>سال</td> </tr> <tr> <td>q</td> <td>فصل</td> </tr> <tr> <td>m</td> <td>ماه</td> </tr> <tr> <td>y</td> <td>تعداد روزهای گذشته</td> </tr> <tr> <td>d</td> <td>روز</td> </tr> <tr> <td>w</td> <td>روز هفته</td> </tr> <tr> <td>ww</td> <td>هفته</td> </tr> <tr> <td>h</td> <td>ساعت</td> </tr> <tr> <td>n</td> <td>دقیقه</td> </tr> <tr> <td>s</td> <td>ثانیه</td> </tr> </tbody> </table>	مقدار	توضیحات	yyyy	سال	q	فصل	m	ماه	y	تعداد روزهای گذشته	d	روز	w	روز هفته	ww	هفته	h	ساعت	n	دقیقه	s	ثانیه	
مقدار	توضیحات																							
yyyy	سال																							
q	فصل																							
m	ماه																							
y	تعداد روزهای گذشته																							
d	روز																							
w	روز هفته																							
ww	هفته																							
h	ساعت																							
n	دقیقه																							
s	ثانیه																							

دانش‌آموزان ثبت‌نام شده در رشتهٔ ریاضی فیزیک را به دست آورید.
پرس‌وجوی Qry-Student را باز کنید. این پرس‌وجو دانش‌آموزان ثبت‌نام شده را برمی‌گرداند. اگر بخواهیم فقط دانش‌آموزان رشتهٔ «ریاضی فیزیک» را ببینیم، شرط MajorCode = 1 را به صورت شکل صفحه بعد در ستون MajorCode اعمال می‌کنیم.

با انتخاب گزینه... Save Object As از منوی اصلی زیر منوی Save As، آن را به نام Qry_StudentMath ذخیره کنید.

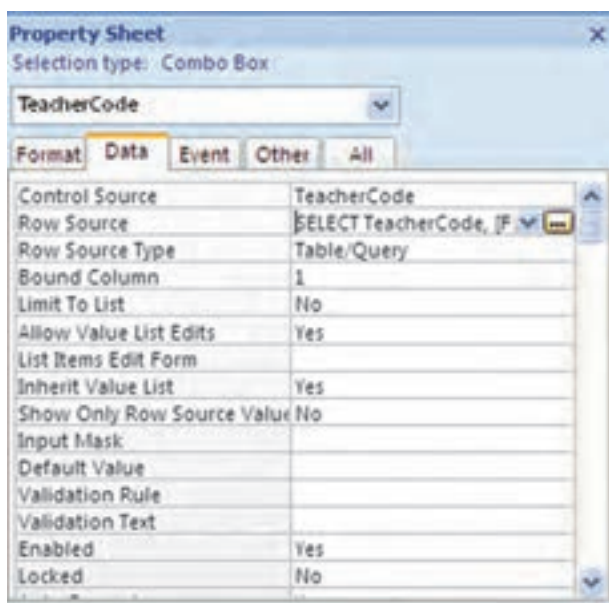


همان‌طور که در شکل بالا می‌بینید، مربع بالای شرط انتخاب نشده است، چرا؟ راهنمایی: پرس‌وجو را اجرا کنید، آیا در نتیجه پرس‌وجو، کد رشته مشاهده می‌شود؟

در فرم Frm_Course، Cobmo Box نام معلم را به گونه‌ای تغییر دهید که نام و نام خانوادگی به صورت یک متن نمایش یابد و به‌نظر نرسد دو فیلد مجزا برای نگهداری آن‌ها وجود دارد.

در پنجره مشخصه‌ها، در زبانه Data، گزینه Row Source به جای انتخاب

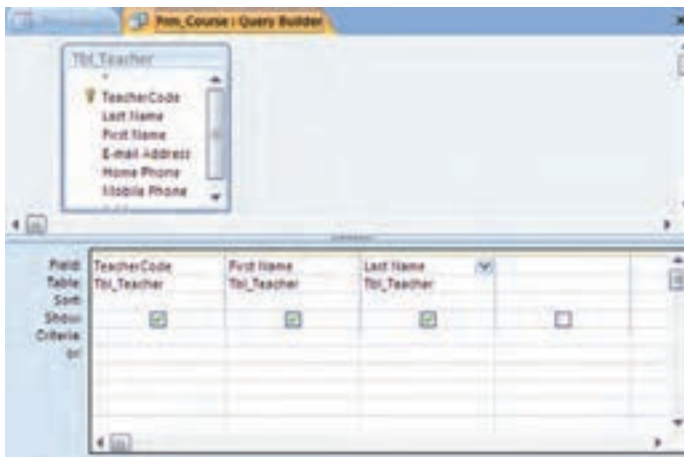
Tbl_Teacher، عبارت زیر را که زبان استاندارد ساخت یافته کار با پایگاه داده است تایپ نمایید. پایان این فصل در خصوص این زبان (SQL) بیشتر صحبت می‌کنیم.




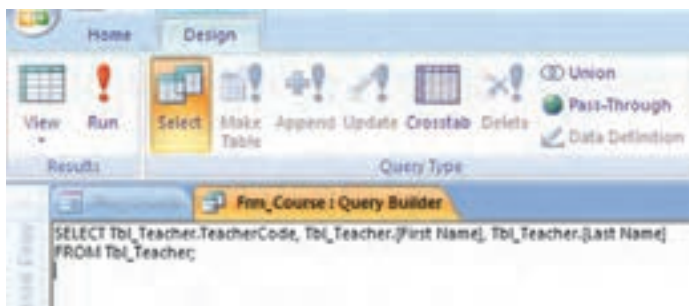
```
SELECT .... TeacherCode, [FirstName] +'...' + [LastName] ...
As ... TeacherName... FROM ... Tbl_Teacher;
```

دقت: هر یک از ... فاصله‌ها در عبارت فوق، حداقل یک کاراکتر Blank است. سایر فاصله‌ها برای خوانایی است و می‌تواند وجود نداشته باشد.

دقت: به جای تایپ عبارت SQL می‌توانید علامت سه‌نقطه کنار آن را کلیک کنید، محیطی مشابه محیط طراحی پرس‌وجو ظاهر می‌شود (شکل زیر) که می‌توانید جدول Tbl_Teacher را اضافه کنید و مانند محیط پرس‌وجو، فیلدهای موردنظر یعنی TeacherCode، FirstName و LastName را انتخاب کنید که در نیمه پایینی پنجره طراحی مشاهده شوند.



از دکمه View  زبانه Home نوار ابزار، می‌توان SQL View را انتخاب کرد و Select مربوطه را مشاهده کرد که به صورت زیر است:



حال آن را به صورت زیر تصحیح کنید :

```
SELECT Tbl_Teacher.TeacherCode ,Tbl_Teacher.FirstName +
FROM Tbl_Teacher;
```

دقت:

SELECT Tbl_Teacher.TeacherCode ,Tbl_Teacher.FirstName + “
+Tbl Teacher.LastName As TeacherName FROM Tbl_Teacher;
با بستن پنجره پرس وجو، SQL مربوط به آن در Row Source قرار می گیرد.
سپس با اجرای فرم، داده های زیر را به کمک آن ذخیره نمایید :

CourseCode	CourseName	CourseUnit	TeacherName	ProgramUnit	ProgramClass
400	مهندسی	2	مهندس محمد علی	10:30-12:00	C10
401	فرآیند	2	مهندس محمد علی	7:30-9:00	900
402	محاسبات عددی	3	مهندس امیر علی	13:30-15:00	C11
403	فرآیند	4	مهندس امیر علی	13:30-15:00	C12
404	آزمایش عددی	2	مهندس امیر علی	13:30-15:00	C13
405	روانشناسی	2	مهندس امیر علی	9:00-10:30	C14
406	روانشناسی	2	مهندس امیر علی	9:00-10:30	C15
407	مدیریت	2	مهندس امیر علی	7:30-9:00	400
408	آزمایش عددی	2	مهندس امیر علی	9:00-10:30	800
409	فرآیند و آزمایش	2	مهندس امیر علی	10:30-12:00	700
410	آزمایش	2	مهندس امیر علی	10:30-12:00	600



علت استفاده از عبارت پرس وجو به جای نام جدول در تمرین های فوق چیست؟

۵-۹ - فیلدهای محاسباتی

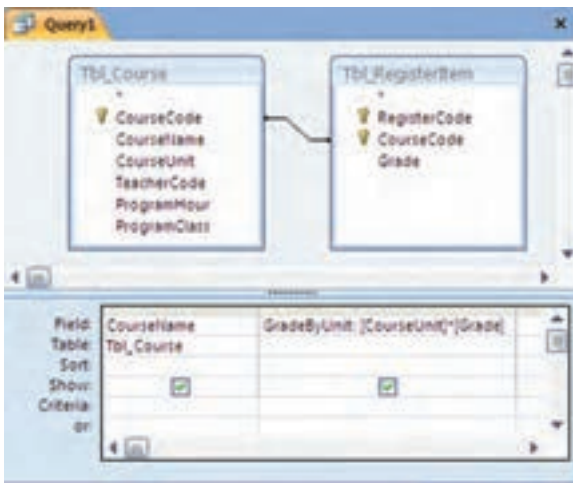
می توان در پرس وجوها، نتیجه عبارت های محاسباتی را در یک ستون مشاهده کرد. این ستون مشابه یک فیلد عمل می کند. این ویژگی یکی از مزایای پرس وجوهاست. به عنوان مثال فرض کنید، می خواهیم حاصل ضرب مقادیر دو فیلد Price و Quantity را مشاهده کنیم و در واقع یک فیلد محاسباتی ایجاد کنیم.

می‌خواهیم حاصل ضرب نمره درس و واحدهای درسی یک دانش‌آموز را به‌دست آوریم. Grade در جدول Tbl_RegisterItem و UnitCourse در جدول Tbl_Course و این مقدار در فیلد GradeByUnit قرار گیرد.

مراحل انجام کار

– در نمای طراحی پرس‌وجو، جدول‌های Tbl_RegisterItem و Tbl_Course را انتخاب می‌کنیم.

– از جدول Tbl_Course، فیلد CourseName را به پرس‌وجو اضافه می‌کنیم.



– در نیمه پایینی پنجره طراحی پرس‌وجو، ستون خالی بعدی را در نظر می‌گیریم و در سطر Field آن، عبارت: GradeByUnit: [CourseUnit]*[Grade] را تایپ می‌کنیم. (شکل روبه‌رو)



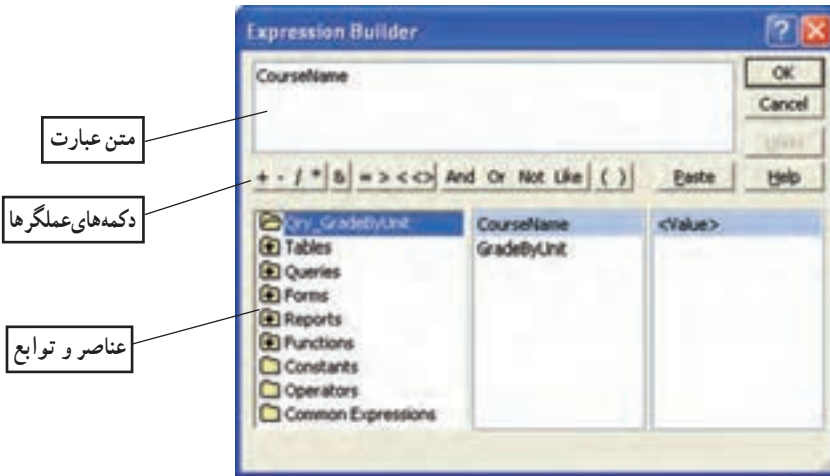
– پرس‌وجو را اجرا می‌کنیم. فیلد مورد نظر در انتهای جواب مشاهده می‌شود (شکل روبه‌رو).

عبارت GradeByUnit که قبل از [Grade]*[CourseUnit] قرار گرفته است،

چیست؟

راهنمایی: در عنوان فیلد در نتیجه خروجی دقت کنید.

استفاده از Expression Builder: ابزار مفیدی برای ایجاد «عبارت» در پرس و جو است. با کلیک روی سلول Field و انتخاب گزینه (Builder) از زبانه Design نوار ابزار (قسمت Query Setup)، پنجره آن ظاهر می‌شود (شکل ۵-۵).



شکل ۵-۵ پنجره ابزار Expression Builder

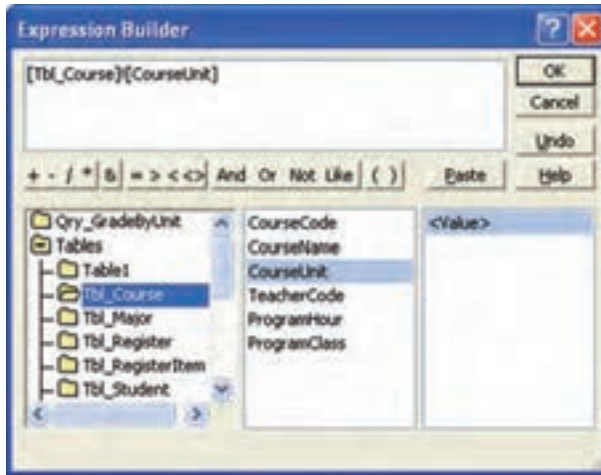
با کمک Expression Builder فیلدها و توابع مورد نظر را انتخاب می‌کنید و بدین ترتیب احتمال خطا و اشتباه تایی کاهش می‌یابد. با استفاده از قسمت Functions می‌توانیم توابع تعریف شده به وسیله کاربر (User Defined) و یا از پیش ساخته شده در اکسس را استفاده کنیم.


در مثال حاصل ضرب نمره درس در واحدهای درسی یک دانش‌آموز را با کمک

Expression Builder ایجاد کنید.


مراحل انجام کار

– از قسمت عناصر و توابع، گزینه Tables را انتخاب کرده، Tbl_Course را انتخاب می‌کنیم. از لیست وسط، فیلد CourseUnit را انتخاب و دوبار کلیک نمایید. این فیلد در قسمت متن عبارت مشاهده می‌شود (شکل زیر).



– سپس از دکمه‌های عملگرها،  را کلیک می‌کنیم تا علامت ضربدر به متن عبارت اضافه شود (شکل زیر).

[Tbl_Course]![CourseUnit]*

– سپس از گزینه Tables، جدول Tbl_RegisterItem را انتخاب می‌کنیم. فیلد Grade را انتخاب و دکمه  را از دکمه‌های عملگرها کلیک می‌کنیم، تا به متن عبارت اضافه شود.


[Tbl_Course]![CourseUnit]*[Tbl_RegisterItem]![Grade]

– در متن عبارت در ابتدای متن قرار گرفته و GradeByUnit: را تایپ می‌کنیم.

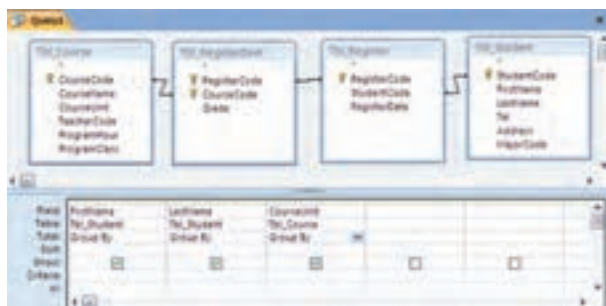
[GradeByUnit]: [Tbl_Course]![CourseUnit]*[Tbl_RegisterItem]![Grade]


– دکمه OK پنجره را کلیک می‌کنیم. با این کار به محیط طراحی پرس‌وجو برمی‌گردیم و عبارت ایجاد شده در قسمت Field ستون موردنظر اضافه می‌شود. مانند حالتی که عبارت را تایپ کرده‌ایم.

۱۰-۵- فیلدهای محاسباتی تجمعی (توابع Aggregate)

گاهی نیاز به محاسبه جمع کل، تعداد کل یک فیلد از یک جدول برحسب مقادیر فیلدی دیگر داریم. برای این محاسبات کاری فراتر از فیلد محاسباتی لازم است. برای این کار باید از توابع تجمعی (Aggregate) استفاده کنیم. در پنجره طراحی پرس و جو با انتخاب دکمه  (Totals) از نوار ابزار، ردیف Total در شبکه نیمه پایینی پنجره طراحی ظاهر می شود. این ردیف شامل مقادیر Max، Min، Avg، Count، Group by و ... است. در ستون فیلدی که می خواهیم مثلاً جمع کل آن حساب شود، در ردیف Sum، Total را انتخاب می نماییم. فیلدی که می خواهیم جمع کل مقدار برحسب آن لیست شود را به صورت Group by انتخاب می کنیم.

جمع واحدهای انتخاب شده هر دانش آموز را حساب کنید.
راهنمایی: شکل زیر در ایجاد پرس و جو ی اولیه کمک می کند.



فیلد **FirstName** و **LastName** و **CourseUnit** را انتخاب کنید. دکمه  (Totals) را کلیک کنید. همان طور که در شکل بالا می بینید در ردیف Total نیمه پایینی برای تمام ستون ها **Group By** نمایش داده می شود. در ستون **CourseUnit**، **Sum** را انتخاب کنید.
با اجرای پرس و جو، نتیجه را به صورت زیر مشاهده می کنید.

FirstName	LastName	SumOfCourseUnit
زهرا	مهرزایی	10
سحر	مهدایی	6
سید کاظم	مهداری	11
سپهرامرضا	یوسفی	2

معدل هر دانش‌آموز را حساب کنید.

راهنمایی: برای محاسبهٔ معدل، باید نمره در تعداد واحدهای درسی ضرب شده و بر مجموع تعداد واحدها تقسیم گردد.


مراحل انجام کار

– ابتدا پرس‌وجویی برای Tbl_Course ایجاد کنید. (شکل زیر به شما کمک می‌کند). آن را با نام Qry_Register ذخیره کنید.



– حال فیلد دیگری به آن اضافه کنید که حاصل ضرب $Grade * CourseUnit$ را برگرداند. نام فیلد جدید را GradeByUnit قرار دهید و ذخیره کنید.

– با کمک این پرس‌وجو، پرس‌وجویی می‌سازیم که از توابع Aggregate استفاده کرده و مجموع تعداد واحدها و مجموع ضرب حاصل شده را برگرداند، تا بتوانیم معدل را بدین طریق محاسبه کنیم.

– پرس‌وجوی جدیدی باز کنید و Qry_Register را به عنوان پرس‌وجوی موردنیاز اضافه کنید. فیلدهای GradeByUnit، StudentName و CourseUnit را انتخاب کنید. با زدن دکمه (totals)  نوار ابزار، ردیف Total در شبکهٔ نیمه پایینی ظاهر می‌شود.

– در ستون GradeByUnit و CourseUnit مقدار را از Group by به Sum تغییر دهید.

– از نوار ابزار، دکمهٔ View را انتخاب و گزینهٔ SQL View را برگزینید. SQL را از عبارت زیر به مقدار جدید اصلاح نمایید:

SELECT Qry_Register.StudentName, Sum(Qry_Register.Course

Unit) As SumOfCourseUnit, Sum(Qry_Register.GradeByUnit) As
SumOfGradeByUnit FROM Qry_Register
GROUP BY Qry_Register.StudentName;

مقدار جدید

SELECT Qry_Register.StudentName, Sum(Qry_Register.Grade-
ByUnit)/ Sum(Qry_Register.CourseUnit) As GradeAverage
FROM Qry_Register
GROUP BY Qry_Register.StudentName;

و آن را با نام Qry_Average ذخیره نمایید.
- با اجرای پرس وجو، نتیجه زیر حاصل می شود:

StudentName	GradeAverage
رضا میرزایی	17.5
سعید هدایتی	18.5
سید کاظم مهاجری	17.1818181818182
محمد رضا یغمایی	14

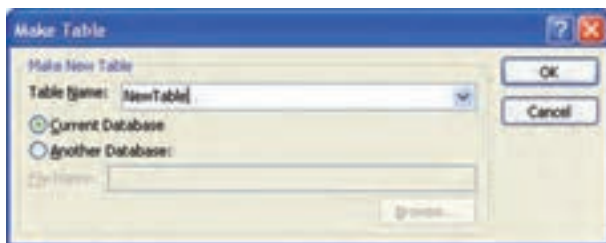
Record: 14 of 4 No Filter

۱-۵-۱۱ - پرس وجوهای عملیاتی Action Query


پرس وجوهای باز یافت رکوردها (Select) را دیدیم. یکی دیگر از انواع پرس وجوها، پرس وجوی عملیاتی است. از این پرس وجو بیشتر برای تغییر داده‌ها (اضافه، ویرایش، حذف) استفاده می‌شود. پرس وجوهای عملیاتی عبارتند از: پرس وجوی جدول ساز، پرس وجوی ویرایش (به هنگام سازی) داده‌ها و پرس وجوی حذف رکورد.

۱-۱۱-۵- ایجاد پرس وجوهای جدول ساز: دیدیم که چگونه از پرس وجوها برای انتخاب مجموعه‌ای از رکوردها و فیلدهای خاص استفاده می‌شود. رکوردهای نتیجه پرس وجو را می‌توانیم در یک جدول جدید ذخیره نماییم. در این صورت فیلدها با همان نوع و مشخصه جدول اصلی ایجاد

می‌شوند و رکوردها نیز به عنوان داده‌های جدول جدید در آن ذخیره می‌شوند. مشابه روش ایجاد پرس‌وجوی بازیافت داده (Select)، پرس‌وجوی موردنظر را ایجاد می‌کنیم. پس از ایجاد پرس‌وجوی جدید (انتخاب جدول/ جدول‌های مورد نیاز و فیلدهای موردنظر)، از قسمت QueryType زبانه Design گزینه Make Table Query را انتخاب می‌کنیم. شکل ۵-۶ ظاهر می‌شود.



شکل ۵-۶- پنجره نام جدول جدید

نام جدول جدید را در قسمت Table Name پنجره تایپ کنید و OK نمایید. پرس‌وجو را با گزینه Save، ذخیره کنید. علامت  در کنار نام پرس‌وجو نشان می‌دهد که این پرس‌وجو از نوع عملیاتی (Action) است. با اجرای این پرس‌وجو یک جدول جدید با نام NewTable در لیست جدول‌های بانک اضافه می‌شود.

نکته



زمانی که نمی‌توانیم با استفاده از پرس‌وجو، اطلاعات موردنیاز را به دست آوریم، می‌توانیم از نتیجه پرس‌وجو، جدول بسازیم و از آن جدول در پرس‌وجوی بعدی استفاده کنیم تا به جواب مطلوب برسیم.

کنجکاوی

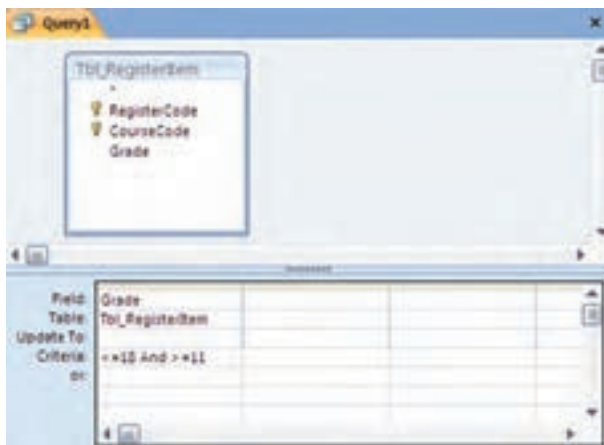
سایر پرس‌وجوهای عملیاتی که می‌توان در Access ایجاد کرد را نام برده، توضیح دهید هر یک برای انجام چه کاری استفاده می‌شوند؟

✓ تمرین ۱-۵:

پرسوجویی ایجاد کنید که معدل دانش‌آموزان را در جدولی با نام Tbl_Average ذخیره کند.

راهنمایی: از پرسوجوی Select تمرین معدل هر دانش‌آموز (Qry_Average) استفاده کنید.

۲-۱۱-۵- ایجاد پرسوجوی به هنگام‌سازی داده‌ها: یکی از انواع پرسوجوهای عملیاتی است که می‌توانیم فیلد خاصی را در رکوردهای خاص به مقدار جدید یا مقدار محاسبه شده جدیدی تغییر دهیم. پس از ایجاد پرسوجوی Select که رکوردهای مربوطه را برگرداند، از قسمت Query Type زبانه Design، Update Query را انتخاب می‌کنیم. ردیفی با عنوان Update to در شبکه پنجره پایینی طراحی پرسوجو (شکل ۷-۵) ظاهر می‌شود. مقدار جدید را در ستون فیلد موردنظر که می‌خواهیم تغییر کند، تایپ می‌کنیم و پرسوجو را ذخیره می‌نماییم. یک پرسوجوی جدید با علامت  که نشان‌دهنده عملیاتی بودن آن است خواهیم داشت. با اجرای آن، فیلد موردنظر در جدول اصلی به مقدار جدید به هنگام می‌شود. برای ایجاد مقدار محاسباتی، می‌توانیم در ردیف Update to، عبارت محاسباتی را تایپ نماییم. حتی می‌توانیم جدول دیگری را افزوده، ارتباط بین آن دو جدول را ایجاد کنیم و فیلد جدول جدید را به عنوان مقدار جدید برای جدول اصلی در قسمت Update to انتخاب کنیم و یا در عبارت محاسباتی، از فیلد جدول جدید استفاده کنیم. استفاده از جدول دیگر در این نوع پرسوجو مشروط به آن است که با جدول اصلی رابطه داشته باشد، در غیر این صورت به هنگام‌سازی معنی‌دار نخواهد بود.




شکل ۷-۵- پنجره به هنگام‌سازی

تمرین ۲-۵:

پرس وجویی ایجاد کنید که به نمره درسی دانش آموزانی که نمره آن‌ها از ۱۸ کمتر و از ۱۱ بیشتر است، یک نمره اضافه کند.


راهنمایی: پرس وجویی ایجاد کنید که از جدول Tbl_Course آن‌هایی که نمره‌شان از ۱۸ کمتر و از ۱۱ بیشتر باشد را برگرداند، سپس با انتخاب گزینه Update Query از منوی Query، ستون Grade را انتخاب و مقدار ردیف Update to آن را +۱ Grade را قرار دهید. با اجرای آن، یک نمره به رکوردهای موردنظر اضافه می‌شود.

۳-۱۱-۵- ایجاد پرس وجوی حذف رکوردها: به کمک این نوع پرس وجو می‌توانیم داده‌هایی (رکوردهایی) با شرایط خاص را حذف کنیم. ابتدا پرس وجویی می‌سازیم که رکوردهای موردنظر برای حذف را برگرداند. در قسمت شرط، معیار موردنظر را اعمال می‌کنیم، در صورت هیچ‌گونه اعمال شرط، تمام رکوردها حذف خواهند شد.

از قسمت Query Type زبانه Design، Delete Query را برمی‌گزینیم. پرس وجو را ذخیره می‌کنیم علامت  در کنار آن ظاهر می‌شود و با اجرای آن رکوردهای موردنظر از جدول حذف می‌شوند و اثری از آن‌ها در جدول اصلی نمی‌بینید.

دقت: رکوردهای تغییر یافته به وسیله این پرس وجوها را نمی‌توان بازگردانید. مثلاً با اجرای حذف، آن رکوردها برای همیشه حذف شده‌اند. پس در استفاده از پرس وجوی عملیاتی خصوصاً حذف باید بسیار دقت کنید. می‌توانید با کمک پرس وجوی جدول‌ساز، ابتدا از داده‌های جدول یک جدول جدید حاوی نسخه پشتیبانی از داده‌ها تهیه کنید. تا در صورت هرگونه اشتباه در پرس وجو، پس از اجرای پرس وجوی حذف، داده‌ها را از دست نداده باشید.

نکته

تمام پرس وجوهای عملیاتی دارای علامت  در کنار نام خود هستند. در هنگام اجرا، Access با پیغامی از شما می‌خواهد که اعمال و اجرای عملیات را تأیید نمایید. البته ظاهر شدن یا نشدن این اخطار و پیغام را می‌توان با تغییر مشخصات Options در محیط Access تغییر داد.

چگونه با اجرای پرس وجوی عملیاتی، پیغام تأیید نمایش داده نمی شود؟

تمرین ۳-۵ :



پرس وجویی ایجاد کنید که در جدول حاوی معدل، دانش آموزانی که معدل آن‌ها زیر ۱۰ است را حذف کند.

راهنمایی: پرس وجویی ایجاد کنید که از جدول Tbl-Average دانش آموزانی که معدل زیر ۱۰ دارند را برگرداند، سپس از قسمت QueryType زیانه Design، گزینه Delete Query را انتخاب کرده و پرس وجو را ذخیره و اجرا نمایید.

۱۲-۵- ایجاد پرس وجوی CrossTab

با این نوع پرس وجو می توانیم اطلاعات یک یا چند جدول را در قالب ردیف و ستون خلاصه نماییم. مثلاً محاسبه sum، average، count، یا دیگر انواع مجموع داده که با دو نوع اطلاع (سطری و ستونی) گروه بندی می شود.

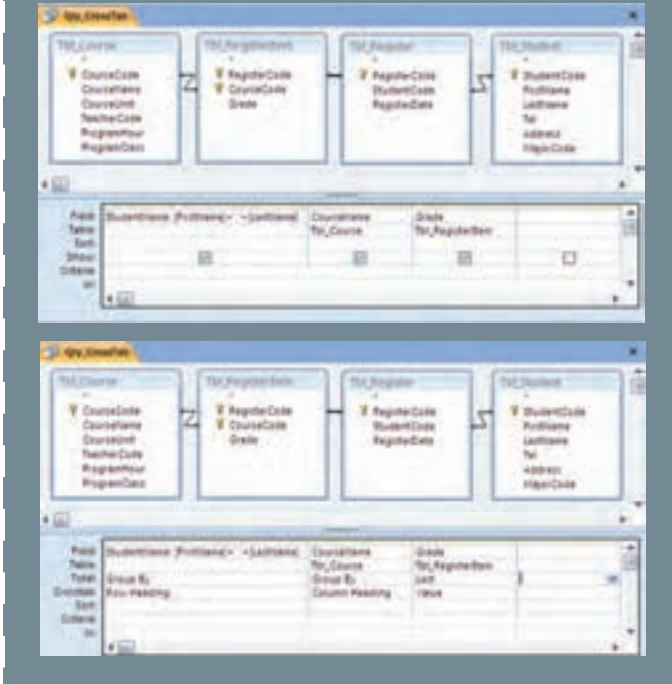
مثلاً یک پرس وجوی ساده، فقط معدل درسی شاگردان کلاس را برمی گرداند. در صورتی که اطلاعات زیاد باشد، مقایسه سخت می شود.

یک پرس وجوی CrossTab همان اطلاع را نشان می دهد. با این تفاوت که اطلاعات سطری با شاگردان و ستونی با درس گروه بندی می شود و این کار تحلیل را آسان تر می کند. بنابراین یک نوع پرس وجوی انتخاب رکوردها (Select) است، با این تفاوت که اطلاعات را خلاصه تر ارائه می کند. در این پرس وجو، داده های یک فیلد خاص را به عنوان ستون های جدول خروجی و فیلد دیگری را به عنوان ردیف های جدول خروجی تعیین می کنیم و مثلاً جمع کل مقادیر سطر بر حسب ستون را به عنوان مقدار محل برخورد سطر و ستون (سلول) تعیین می کنیم.

برای ایجاد این نوع پرس وجو، از Query Wizard، گزینه CrossTab Query wizard را انتخاب کنید. سپس جدول را انتخاب نموده و یک فیلد را به عنوان «عنوان سطر» Row heading و فیلد دیگری را به عنوان «عنوان ستون» Column heading برمی گزینیم. یکی از توابع را به عنوان داده های محل تقاطع سطرها و ستونها انتخاب می کنیم.

تمرین ۴-۵: ✓

معدل هریک از درس‌های هر دانش‌آموز را حساب کنید.
راهنمایی: از شکل‌های زیر کمک بگیرید.



۵-۱۳- زبان پرس‌وجوی ساخت‌یافته SQL

همان‌طور که قبلاً دیدیم در قسمت طراحی پرس‌وجو، در نوار ابزار گزینهٔ SQL View وجود دارد. با انتخاب این گزینه، عبارت SQL مربوط به پرس‌وجو دیده می‌شود. می‌توانید پرس‌وجوی Qry_Student را انتخاب کرده، به Design بروید و نمای SQL view را انتخاب کنید. شکل ۵-۸ را خواهید دید.

```
Qry_Student
SELECT Tbl_Student.StudentCode, Tbl_Student.LastName, Tbl_Student.FirstName,
Tbl_Student.Tel, Tbl_Student.Address, Tbl_Major.MajorName
FROM Tbl_Major INNER JOIN Tbl_Student ON Tbl_Major.MajorCode=Tbl_Student.MajorCode
ORDER BY Tbl_Student.LastName, Tbl_Student.FirstName;
```

شکل ۵-۸- نمای SQL در پنجرهٔ طراحی پرس‌وجو

در این پنجره، عباراتی نوشته شده است که یک دستور Select است که با قواعد زبان استاندارد ساخت یافته بانک SQL نوشته شده است. اگر این عبارت را با عبارت سایر پرس و جوهای موجود مقایسه کنید، می توانید شباهتی بین آنها دریابید یعنی قالب دستورات زبان ثابت است. با رعایت اصول و قواعد زبانی، می توان آن را به کار گرفت.

SQL، زبانی است که تمام دستورات پرس و جو مثل Delete، Update، Select و ... به وسیله آن تعریف می شوند. ممکن است کلماتی چون Order by، Where و ... نیز داشته باشد. عموماً این زبان برای ایجاد پرس و جوهای بانک استفاده می شود ولی می توان از آن برای ایجاد یا تغییر ساختار پایگاه داده نیز استفاده کرد که به این نوع SQL زبان تعریف داده (DDL)^۱ گویند که در این کتاب به آنها نمی پردازیم. در ادامه روش ایجاد پرس و جوهای معمول را با قواعد این زبان بررسی می کنیم که به آنها زبان دستکاری داده (DML)^۲ گفته می شود.

با کمک محیط گرافیکی پرس و جو به سادگی برخی پرس و جوها را ایجاد کردیم. ولی پرس و جوهای پیچیده تر فقط با نوشتن عبارات زبان ساخت یافته و با قالب مجاز ایجاد می شوند و نمی توان آن ها را به طریق گرافیکی که در موارد بالا دیدیم، ایجاد کنیم. مثلاً:

Sub – query, Union, Data–Definition

قالب زبان عبارات پرس و جو : ابتدا علایم استفاده شده در قواعد را شرح می دهیم :

۱- {} باید یکی از موارد داخل آن استفاده شود. موارد انتخابی با "[]" و یا [] از هم جدا شده اند.

۲- [] اختیاری بودن.

۳- پررنگ بودن قلم نشانه کلمه کلیدی بودن آن واژه است. مثل SELECT (بزرگ یا کوچک بودن حروف مهم نیست ولی ما برای خوانایی از حروف بزرگ استفاده کرده ایم).

۱-۳-۵- دستور SELECT : برای انتخاب رکوردها از این دستور و با قالب زیر استفاده می شود. این دستور می تواند بسیار ابتدایی و ساده باشد و یا بسیار پیچیده گردد. این دستور زیربنایی ترین دستورات است و سایر دستورات را می توان با تغییر این دستور ایجاد کرد. قالب قواعد زبانی آن عبارت است از :

SELECT [ALL |DISTINCT | DISTINCTROW [[TOP n[PERCENT]]]

{*|Table.*| [table.] field1[AS alias1][, [table.] field2 [AS alias2][, ...]]

FROM table [...][IN externaldatabase]

[WHERE...]

[GROUP BY...]

[HAVING...]

[ORDER BY...]

پرس وجوهای زیر را ایجاد کنید.

۱- مثال Select، می‌خواهیم تمام رکوردهای معلمان مدرسه را مشاهده کنیم:

```
SELECT ALL * FROM Tbl_Teacher  
یا  
SELECT * FROM Tbl_Teacher
```

۲- مثال Distinct، می‌خواهیم درس‌های ثبت‌نام‌شده ترم را به دست آوریم:

```
SELECT DISTINCT CourseCode, CourseName  
FROM Tbl_Course, Tbl_RegisterItem  
WHERE Tbl_Course. Code = Tbl_RegisterItem. CourseCode
```

* اگر از DISTINCT استفاده نکنیم، چون هر درس به وسیله چندین دانش‌آموز ثبت‌نام شده است، رکوردهای مشابه برمی‌گرداند.

۳- مثال As alias، می‌خواهیم نام و نام خانوادگی معلمان را به صورت یک فیلد اطلاعاتی با عنوان TeacherName داشته باشیم:

```
SELECT TeacherCode, LastName + ' ' + FirstName FROM Tbl_Teacher
```

این پرس وجو را اجرا کنید. نتیجه به صورت زیر خواهد بود:



TeacherCode	Expr1001
1045	سعدی نژاد حامد رضا
1046	مهدی فر محمد
1047	میرزا پور علیرضا
1048	هدایتی فر کامبیز
1049	حسینی امیر علی
*	0

حال نام مستعاری به فیلد منتسب می‌کنیم، عبارت AS TeacherName :
 SELECT TeacherCode, LastName + ' ' + FirstName AS TeacherName
 FROM Tb1_Teacher

نتیجه جدید به صورت زیر خواهد بود :

TeacherCode	TeacherName
1045	سعیدی نژاد حامد رضا
1046	مهدی فر محمد
1047	میرزاپور علیرضا
1048	هدایتی فر کامبیز
1049	حسینی امیر علی
*	0



۱- Top n [percent] چگونه عمل می‌کند؟ کجا می‌توان از این قابلیت استفاده کرد؟

راهنمایی: وقتی داده‌های جدول زیاد باشد. حجم رکوردهای نتیجه بالا است و سرعت بازبازی کند باشد و مشاهده نتیجه پرس و جو طول بکشد.

۲- در قسمت FROM پرس و جو می‌توان از گزینه IN استفاده کرد. این گزینه چه عملکردی دارد؟

راهنمایی: جدول مورد نظر در بانک داده دیگری قرار داشته باشد.

۲-۱۳-۵ دستور SELECT ... INTO : ایجاد جدول جدید شامل رکوردهای نتیجه

پرس و جو (پرس و جوی جدول ساز).

قسمت SELECT آن مشابه قبل است. تنها در خط اول پس از انتخاب فیلدهای مورد نظر،

کلید واژه INTO را به همراه نام جدول جدید قرار می‌دهیم. در صورتی که بخواهیم جدول جدید در پایگاه داده دیگری قرار گیرد، آدرس آن را در IN قید می‌کنیم.

```
SELECT field1 [,field2[,...]] INTO newtable [IN externaldatabase] FROM source-  
TableName
```

مثال ۵-۱

نسخه پشتیبانی از جدول معلمان ایجاد کنید :
پرس‌وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT * INTO Tb1-TeacherBackUp  
FROM Tb1-Teacher
```

۳-۱۳-۵- دستور DELETE : حذف رکوردها.

```
DELETE [table.*]  
FROM table  
WHERE criteria
```

رکوردهای جدول معلمان را حذف کنید.
پرس‌وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
DELETE  
FROM Tb1-Teacher
```

مثال ۵-۲

۴-۱۳-۵- دستور INSERT : اضافه کردن رکوردها. دو نوع دستور اضافه کردن رکورد وجود دارد. اضافه کردن به وسیله مقادیر ثابت داده (Single append) و اضافه کردن رکوردهای از پرس‌وجوی نوع Select (Multiple append).

• Multiple-record append query:

```
INSERT INTO target-TableName [( field1[, field 2 [,...]])] [IN externaldatabase]  
SELECT [source-TableName.] field1[, field 2 [,...]]  
FROM Source-TableName
```

اطلاعات جدول معلمان را از نسخه پشتیبان برگردانید.
پرسوجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
INSERT INTO Tb1-Teacher (TeacherCode, HomePhone Address)
SELECT TeacherCode, FirstName, LastName, HomePhone, Address
FROM Tb1-TeacherBackUp
```

• Single-record append query:

```
INSERT INTO target-TableName [(field1[, field2[,...]])]
VALUES (value 1[,value 2[,...])]
```

اطلاعات معلم جدید را اضافه کنید.

پرسوجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
INSERT INTO Tb1-Teacher (TeacherCode, FirstName, LastName,
HomePhone, Address)
VALUES ('۸۸۹۷۱۰۳۴' و 'یعقوبی' و 'صدف' و '۰۷۰')
```

۵-۱۳-۵ - دستور UPDATE : ویرایش رکوردها

```
UPDATE table
SET newvalue
WHERE criteria;
```

نام معلم جدید به اشتباه «صدف» ثبت شده است. آن را ویرایش نمایید: پرسوجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
Update Tb1-Teacher SET FirstName='صدف'
WHERE TcaherCode=1070
```

۵-۱۳-۶ - عملگر UNION : برای ایجاد پرسوجوی ترکیب رکوردهاست.

```
[TABLE] query1 UNION [ALL] [TABLE] query2
[UNION[ALL] [TABLE] queryn[...]]
```

در این دستور باید تعداد فیلدهای دو Select که با هم Union می‌شوند، یکسان باشد. نوع، ترتیب و نام آن‌ها نیز یکی باشد، در صورت عدم تساوی نام، باید از Alias (افزودن عنوان جدید با AS) استفاده کرد.

لیست اسامی معلمان و دانش‌آموزان مدرسه را به‌دست آورید.
پرس‌وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT StudentCode AS PersonCode, FirstName AS
PersonFName, LastName AS PersonLName
FROM Tb1-Student
UNION
SELECT TeacherCode AS PersonCode, FirstName AS
PersonFName, LastName AS PersonLName
FROM Tb1-Teacher
```

۷-۱۳-۵- دستور JOIN: دیدیم یکی از روش‌های انتخاب اطلاعات از دو یا چند جدول، آن است که در FROM نام جدول‌ها را با علامت "،" از هم جدا کرده و در WHERE ارتباط آن‌ها را به‌صورت شرایط دوه‌دو بیان کنیم. یکی دیگر از راه‌ها که زمان ایجاد پرس‌وجو با کمک گرافیک نیز دیده‌ایم، اتصال جدول‌ها با دستور JOIN است. شامل دو روش INNER JOIN و OUTER JOIN است.

INNER JOIN: اتصال دو جدول، زمانی که اطلاعات متناظر در هر دو جدول وجود داشته باشند. اطلاعاتی که در یکی باشد و در دیگری وجود نداشته باشد، در نتیجه پرس‌وجو نمایش داده نخواهند شد.

```
FROM table INNER JOIN table2 ON table1. field1 compopr table2. field2
```

اطلاعات معلمان هر درس را به‌دست آورید.
پرس‌وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT CourseName, FirstName, LastName
FROM Tb1-Teacher INNER JOIN Tb1-Course
ON Tb1-Teacher. TeacherCode = Tb1-Course. TeacherCode
```

— Outer Join (Right/Left): اطلاعات متناظر دو جدول در نتیجه پرس‌وجو مشاهده می‌شود، حتی اگر در دیگری وجود نداشته باشد. LEFT JOIN تمام اطلاعات را از جدول اول نشان می‌دهد حتی اگر در جدول دوم رکوردی متناظر آن وجود نداشته باشد. در این صورت مقدار فیلدهای انتخاب‌شده از جدول دوم را NULL می‌آورد. به همین ترتیب، RIGHT JOIN تمام اطلاعات جدول

دوم را می‌آورد. عبارت تساوی بعد از ON نیز، ارتباط دو جدول را نشان می‌دهد.
 FROM table1 {LEFT\RIGHT} JOIN table2 ON table1. field1 compopr table2. field2

مثال ۱۸-۵

اطلاعات معلمان هر درس را به‌دست آورید. در ضمن معلمانی که در این ترم درسی
 ارایه نداده‌اند نیز آورده شود.
 پرس‌وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT FirstName, LastName, CourseName
FROM Tb1-Teacher LEFT JOIN Tb1-Course
ON Tb1-Teacher. TeacherCode = Tb1-Course. TeacherCode
```

دروس ثبت‌نام‌شده دانش‌آموزان را به‌دست آورید. در ضمن دانش‌آموزانی که در
 این ترم هنوز درسی ثبت‌نام نکرده‌اند نیز آورده شود.

```
SELECT FirstName, LastName, CourseName
FROM Tb1-Student LEFT JOIN (Tb1-Register LEFT JOIN
(Tb1-Course RIGHT JOIN Tb1-RegisterItem ON
Tb1-Course. CourseCode = Tb1-RegisterItem. CourseCode)
ON Tb1-Register. RegisterCode=
Tb1-RegisterItem. RegisterCode) ON
Tb1-Student. StudentCode = Tb1-Register. StudentCode
```

مثال ۱۹-۵



NULL چه مقداری است. اعمال شرط و یا محاسبات روی آن چگونه است؟

۸-۱۳-۵- اعمال سایر شرایط به پرس‌وجوها

● عبارت WHERE شرط و معیار محدود ساختن رکوردهاست.

یک عبارت Where، می‌تواند شامل ۴۰ عبارت باشد که با عملگرهای منطقی And یا Or به هم

وصل شده‌اند. عملگرهای ریاضی و منطقی عبارت شرط را می‌سازند. معیارهای عبارت Where نیز مشابه فیلتر که در قسمت ۷-۴ توضیح دادیم است و از همان قواعد ذکر شده پیروی می‌کند.

معلمی با نام 'مهدی‌فر' را حذف کنید.
پرس‌وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
DELETE  
FROM Tb1-Teacher  
WHERE TeacherCode= 1046
```

می‌توانستیم از عبارت زیر استفاده کنیم ولی دقت نمایید که در صورتی که چند رکورد با نام خانوادگی مذکور وجود داشته باشد، با اجرای این پرس‌وجو تمامی آن‌ها حذف می‌شوند.

```
DELETE  
FROM Tb1-Teacher  
WHERE LastName = 'مهدی‌فر'
```

● عبارت **ORDER BY**: مرتب‌سازی رکوردها.

```
SELECT fieldlist  
FROM table  
WHERE selectcriteria  
[ORDER BY field1 [ASC|DESC][,field2[ASC|DESC]][,...]]
```

با این عبارت ترتیب نمایش رکوردها را تغییر می‌دهیم. بدون ذکر Order by رکوردها بدون هیچ ترتیب خاصی و براساس همان ترتیب ورود اطلاعات نمایش داده می‌شوند. مرتب‌سازی ابتدا براساس فیلد اول و در صورت تساوی مقادیر این فیلد، روی فیلد دوم ذکر شده در عبارت خواهد بود. دو واژه ASC و DESC ترتیب صعودی و نزولی را مشخص می‌کنند:

– ASC: مرتب‌سازی به صورت صعودی است. از مقدار کم‌تر به بیش‌تر.

– DESC: مرتب‌سازی به صورت نزولی است. از مقدار بیش‌تر به کم‌تر.

لیست اسامی معلمان براساس نام خانوادگی و در صورت تساوی نام خانوادگی براساس نام مرتب‌شده باشد به دست آورید.

پرس وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT Tb1-Teacher. FirstName,
Tb1-Teacher. LastName
FROM Tb1-Teacher
ORDER BY Tb1-Teacher. LastName ASC,
Tb1-Teacher. FirstName ASC
```

● عبارت **GROUP BY**: پرس وجوی تجمعی (Aggregate) شامل: جمع کل، تعداد کل

و ...

```
SELECT fieldlist
FROM table
WHERE criteria
[GROUP BY groupfieldlist]
```

SQL پرس وجوی محاسبه معدل Qry-Average را مشاهده کنید.

تمرین ۵ - ۵:



```
SELECT Qry-Register. StudentName,
Sum (Qry-Register. GradeByUnit)/
Sum (Qry-Register. CourseUnit) AS GradeAverage
FROM Qry-Register
GROUP BY Qry-Register. StudentCode
```

این پرس وجو یعنی نام دانش آموز را در نظر می گیرد و مجموع حاصل ضرب نمره در واحد درسی را به دست آورده، برای محاسبه معدل تقسیم انجام می دهد.



نکته

تمام فیلدهای عبارت Select غیر از آنهایی که در توابع تجمعی Count، Min، Sum استفاده شده اند، باید در لیست قسمت Group By آورده شده باشند. گروه بندی باید روی فیلد پرس وجو باشد.

● عبارت **HAVING**: شرط و معیار در پرس‌وجوی تجمعی است.

زمانی که از توابع تجمعی Aggregate و Group By استفاده شود، برای اعمال شرط پس از اجرای پرس‌وجو، باید از معیار در HAVING استفاده کنیم. اگر پرس‌وجو دارای عبارت شرط Where باشد، رکوردها ابتدا به‌وسیله آن شرط محدود می‌شوند، سپس نتیجه تابع تجمعی و فیلدها محاسبه می‌شوند و پس از آن شرط HAVING روی نتیجه اعمال می‌گردد. در این صورت می‌توان از فیلدهای Group By و فیلدهای توابع تجمعی نیز در شرط استفاده کنیم. در صورتی که در عبارت Where فقط می‌توانستیم فیلدهای Group By را استفاده و بر آن اساس پرس‌وجو را محدود کنیم.

```
SELECT fieldlist  
FROM table  
WHERE selectcriteria  
GROUP BY groupfieldlist  
[HAVING groupcriteria]
```

می‌خواهیم معدل دانش‌آموزانی را به‌دست آوریم که بیش از ۱۰ واحد درسی انتخاب کرده‌اند. پرس‌وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT Qry-Register. StudentName,  
SUM (Qry-Register.GradeByUnit)/SUM (Qry-Register.CourseUnit)  
AS GradeAverage  
FROM Qry-Register  
GROUP BY Qry-Register. StudentName  
HAVING SUM (Qry-Register.CourseUnit)>10
```

مثال ۲۲-۵



دقت کنید که نمی‌توانیم با کمک عبارت شرط Where این پرس‌وجو را ایجاد کنیم.

دیدیم که با محیط گرافیکی ایجادشده به‌وسیله Access به‌آسانی و بدون نوشتن دستورات زبانی SQL، می‌توانیم پرس‌وجوی مورد نظرمان را ایجاد کنیم. اما همان‌طور که گفتیم برخی پرس‌وجوها از قبیل: Data-Definition، Union، Sub-query را باید با نوشتن دستورات زبان در قسمت SQL View پرس‌وجو ایجاد کنیم.

● **UNION**: برای ایجاد UNION ابتدا دو پرس‌وجوی مورد نیاز را ایجاد می‌کنیم که تعداد، نوع، ترتیب و نام مشابهی از فیلدها را برگردانند. سپس در یک پرس‌وجوی جدید SQL اولی را کپی می‌کنیم، کلیدواژه UNION را تایپ می‌کنیم و پس از آن SQL دومی را کپی می‌کنیم. می‌توانیم تعداد Select‌های بیشتری را نیز اضافه کنیم، یعنی تایپ مجدد UNION در ادامه و کپی SQL دیگر و به همین ترتیب.

● **SUBQUERY**: این نوع پرس‌وجو، Select‌ای است که داخل پرس‌وجوی دیگر استفاده شود. سه روش استفاده از آن در پرس‌وجو در زیر آورده شده است:

1. comparison [ANY|ALL|SOME] (sqlstatement)
2. expression [NOT] IN (sqlstatement)
3. [NOT] EXISTS (sqlstatement)

می‌توانیم از SubQuery به جای استفاده از عبارت محاسباتی در فیلد Select یا شرط Where و شرط Having استفاده کنیم. به‌طور مثال:

1.

```
SELECT* FROM Tb1-Student
WHERE StudentCode = ANY
(SELECT Student Code
From Tb1_Register INNER JOIN Tb1-RegisterItem ON
Tb1-Register. RegisterCode = Tb1-Register Item. RegisterCode
WHERE Tb1-RegisterItem. Grade>=10);
```
2.

```
SELECT* FROM Tb1-Student
WHERE StudentCode IN
(SELECT StudentCode
FROM Tb1-Register INNER JOIN Tb1_RegisterItem ON
Tb1-Register. Registercode= Tb1_Register Item. Registercode
WHERE Tb1-RegisterItem. Grade>=10);
```
3.

```
SELECT * FROM Tb1- Student
WHERE EXISTS
(SELECT StudentCode
FROM Tb1-Register INNER JOIN Tb1-RegisterItem ON
Tb1-Register. RegisterCode = Tb1-RegisterItem. RegisterCode
WHERE Tb1-RegisterItem. Grade > =10);
```

● **Data Definition**: همان طور که گفتیم دستورات SQL ای وجود دارند که برای ایجاد یا تغییر ساختار پایگاه داده استفاده می‌شوند. در این جا یک مثال از ایجاد جدول با دستور SQL را می‌بینیم. تمام تغییراتی که در طراحی بانک با کمک محیط گرافیکی Access انجام می‌دادیم، به وسیله دستورات زبان نیز قابل انجام است. مثل تغییر نام فیلد، تغییر نوع داده یک فیلد، افزودن فیلد جدید، افزودن کلید اصلی به جدول و

به طور مثال با اجرای دستور زیر که در SQL View یک پرس و جوی جدید تاپ می‌نماید، جدولی با نام Tb1-Friends و با شش فیلد و کلید اصلی روی فیلد FriendID ایجاد می‌شود.

```
CREATE TABLE Tb1-Friends
([FriendID] integer,
[LastName] text (20),
[FirstName] text (30),
[Bitrhdate] date,
[Phone] text (25),
[Notes] memo,
CONSTRAINT [Index1] PRIMARY KEY ([FriendID]))
```

خلاصه فصل

برای تصمیم‌گیری نیاز به بازیافت اطلاعات ثبت شده در جدول‌های بانک داریم و باید این اطلاعات را تجزیه و تحلیل کنیم. از پرس و جوها برای دریافت اطلاعات از بانک استفاده می‌شود. پرس و جوها انواع مختلفی دارد که معمول‌ترین آن پرس و جوی Select است. به هر پرس و جو می‌توان محدودیت و شرط اضافه کرد. می‌توان فیلدهای غیرجدولی از نوع محاسباتی به پرس و جو اضافه کرد. با اجرای پرس و جو نتیجه آن به صورت جدولی نشان داده می‌شود که شبیه نمای Datasheet جدول است و همان عملیات را می‌توان روی آن اعمال کرد مثل: مرتب‌سازی، فیلتر کردن و ... البته نتیجه

پرس وجو در فرم یا گزارش‌ها نیز استفاده می‌شود. نوع دیگر پرس وجو، پرس وجوهای عملیاتی هستند که از آن‌ها برای تغییر روی داده‌ها استفاده می‌شود و به آن زبان دستکاری داده DML می‌گوییم. البته پرس وجوی عملیاتی برای تغییر ساختار بانک نیز وجود دارد که به آن زبان تعریف داده DDL می‌گوییم. از انواع DMLها، می‌توانیم به پرس وجوی جدول‌ساز، به هنگام‌سازی، حذف اشاره کنیم که به وسیله محیط گرافیکی Access نیز می‌توانند ایجاد شوند. در پشت ظاهر گرافیکی هریک از این پرس وجوها، زبان ساخت‌یافته SQL قرار دارد. یعنی معادل هریک از این امکانات یک دستور زبانی قرار دارد که با آن‌ها آشنا می‌شویم. پرس وجوهای پیچیده‌تر و پیشرفته‌تر به‌سختی به‌وسیله محیط گرافیکی قابل ایجاد هستند و باید به‌وسیله برنامه‌نویس و با قواعد زبان SQL نوشته شوند.

خودآزمایی

- ۱- پرس وجوها چه کاری انجام می‌دهند و چه زمانی قابل استفاده هستند؟
- ۲- پنجره طراحی پرس وجو را به اختصار توضیح دهید.
- ۳- معیارهای پرس وجو به چه منظور استفاده می‌شوند؟
- ۴- فیلدهای محاسباتی را توضیح دهید. با چه ابزار گرافیکی قابل ایجاد هستند؟
- ۵- پرس وجوهای عملیاتی را شرح دهید.
- ۶- زبان پرس وجوی ساخت‌یافته SQL چیست؟
- ۷- عملکرد HAVING را با ذکر یک مثال توضیح دهید.

ایجاد و کار با گزارش‌ها

در این فصل ایجاد گزارش و عملیات روی آن را بررسی می‌کنیم. گزارش‌ها یکی از راه‌های مفید نمایش و ارائه داده‌ها در یک قالب چاپی است. چون اندازه و ظاهر هر شیء روی گزارش قابل کنترل است، بنابراین می‌توانید اطلاعات را همان‌گونه که می‌خواهید عرضه کنید.

- هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:
 - تفاوت گزارش با سایر اشیاء را تشخیص دهد.
 - نماهای مشاهده گزارش Preview، ایجاد گزارش و تغییر طراحی را بشناسد.
 - گزارش‌ها را به وسیله Report Wizard یا report Wizard ایجاد کند.
 - می‌توانید گزارش‌هایی ایجاد کنید که:
 - داده‌ها را در گروه‌هایی دسته‌بندی کرده و نمایش دهد.
 - جمع کل، جمع برحسب گروه‌ها، درصد و غیره را محاسبه کرده، نمایش دهد.
 - شامل زیر گزارش، زیر فرم، گراف و نمودار باشد.
 - برچسب‌های پستی ایجاد کند.
 - داده‌های جدول‌ها و پرس‌وجوهای مختلف را ترکیب کرده و نمایش دهد.
 - داده‌ها را با کمک نوع قلم، خطوط و تصاویر در قالبی جذاب ارائه دهد.

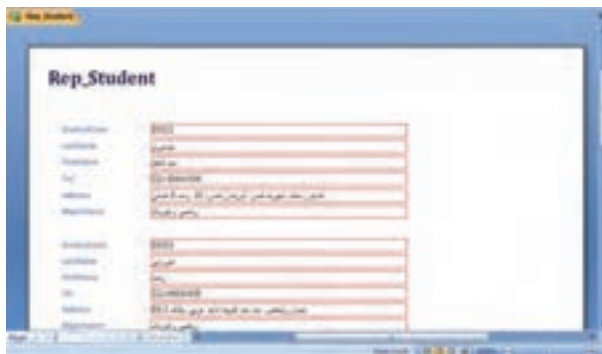
یک گزارش، رکوردها را شبیه نمای Datasheet فرم‌ها و جدول‌ها نشان می‌دهد. تفاوت نمای View در گزارش با Datasheet آن است که در گزارش خلاصه آماری، عناوین، زیرنویس، شماره صفحه، قالب نمایش داده زیباتری فراهم می‌شود. همچنین می‌توان داده‌های قابل نمایش را انتخاب کرد، داده‌ها را گروه‌بندی کرد و جمع کل روی آن گروه و جمع نهایی را نمایش داد. گزارش‌ها در سه نما قابل نمایش است.

- نمای Design، می‌توانید ظاهر گزارش را ایجاد و یا تغییر دهید.
- نمای PrintPreview، با مرور داده‌های واقعی در هر صفحه گزارش می‌توانید گزارش را تأیید کنید.
- نمای Layout Preview، با داده‌های نمونه فقط قالب نمایش گزارش را سریع مشاهده کنید.



گزارش فقط لیستی از مقادیر داده نیست. هر زمان که نیاز دارید داده‌هایی را با قالب خاص چاپ کنید باید گزارش ایجاد نمایید. گزارش می‌تواند لیست اطلاعات، برچسب چاپی، صورت‌حساب و ... باشد.

قبل از چاپ گزارش به وسیله چاپگر، ابتدا آن را در نمای Preview مشاهده کنید. این حالت نمای گزارش را روی صفحه نشان می‌دهد و مشابه نمونه چاپی است. در این حالت متوجه تغییرات مورد نیاز خواهید شد و در زمان و کاغذ صرفه‌جویی می‌شود. با انتخاب دکمه Preview در پنجره ابزار بانک، گزارش موجود را در صفحه پنجره مشاهده کنید (شکل ۱-۶).



شکل ۱-۶ - پنجره نمای Preview یک گزارش نمونه

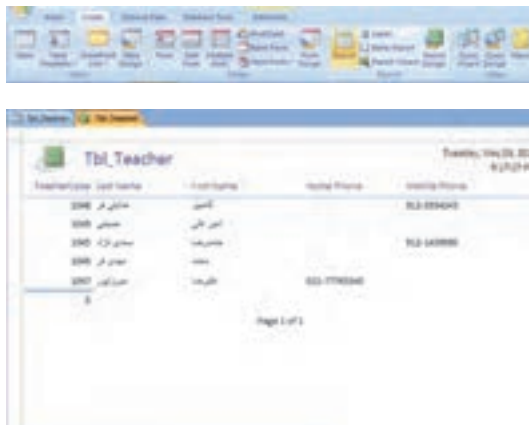
معمولاً نباید تمام داده‌ها در یک گزارش نمایش داده شوند، مگر گزارش برای نمایش جزئیات باشد مثل: لیست موجودی انبار، لیست مشتریان و غیره. به‌طورکل بخشی از اطلاعات و مقادیر جدول‌ها که مفید و برای تصمیم‌گیری مورد نیاز است نمایش داده می‌شوند.

بهتر است گزارش‌ها از داده‌های پرس‌وجوها ایجاد شوند، هرچند می‌توانیم جدول‌ها را در گزارش اضافه کرده و مانند یک پرس‌وجو عمل کنیم ولی بهتر است برای نمایش اطلاعات چند جدول و حتی یک جدول، ابتدا پرس‌وجویی که داده‌های مورد نیاز را برگرداند ساخته شود و از آن پرس‌وجو در ایجاد گزارش استفاده گردد.

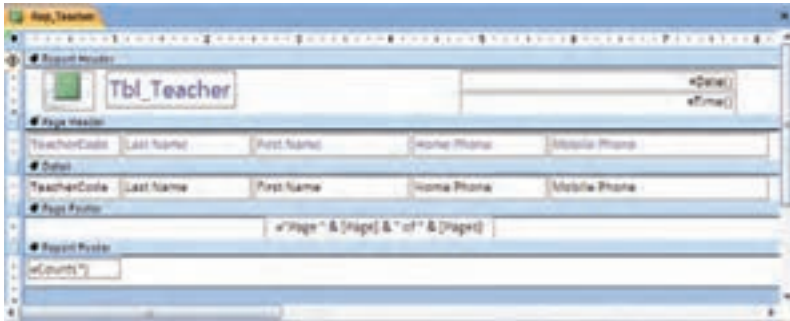
۱-۶- ایجاد گزارش ساده به‌وسیلهٔ AutoReports

با این امکان در بانک اطلاعاتی Access، می‌توان در نمای Datasheet، گزارش‌ها را به‌سرعت ایجاد کرد. این نوع گزارش فقط نام فیلدها و مقادیر متناظر را لیست می‌کند (شکل ۲-۶). تفاوت این لیست با چاپ نمای Datasheet آن است که داده‌ها در این نما قالب‌بندی می‌شود و به‌صورت خوانا شکل می‌گیرد بدون آن که اطلاعات زیادی را در یک صفحه فشرده سازد.

می‌خواهیم گزارش سریعی از معلمان مدرسه ایجاد کنیم:
 جدول یا پرس‌وجوی مورد نظر را انتخاب نمایید، مثلاً جدول Teacher - Tb1.
 آن را در نمای Datasheet باز کنید.
 در صورت نیاز برای محدود کردن رکوردها، Filter اعمال نمایید.
 مطابق شکل زیر روی گزینهٔ Report کلیک کنید.



- بانک گزارشی ساخته، آن را در نمای Preview نشان می‌دهد.
- اگر نمایش گزارش مطابق آنچه می‌خواهید باشد، آن را با دکمه Print چاپ کنید.
- دکمه Design را کلیک کنید تا طرح گزارش را ببینید (شکل زیر). به‌خاطر داشته باشید که شاید درک این نما برای بار اول کمی مشکل به‌نظر رسد.



- نگران ارقام خاص روی صفحه نباشید. AutoReport (Report) طراحی گزارش را ایجاد کرده است و نیازی به استفاده از ابزارهای نمای Design و ساخت گزارش نداریم.
- نمای Design را ببینید. در صورتی که می‌خواهید آن را ذخیره نمایید، نامی به آن اختصاص دهید. زمانی آن را ذخیره کنید که مجدداً به آن نیاز پیدا خواهید کرد.



نکته

برای ایجاد گزارش‌های استاندارد و مناسب‌تر و همچنین تا حدودی ساده، از گزینه Wizard استفاده می‌شود.

نکته: Report: ابزارهای طراحی گزارش را برای ایجاد آن به کار می‌برد. زمانی که یاد گرفتید چگونه از این ابزارها استفاده کنید و گزارش را اصلاح کنید و تغییر دهید، می‌توانید از این امکان برای ایجاد گزارش اولیه استفاده کنید و سپس گزارش را مطابق نظر خود تغییر دهید.

AutoForm چه کاری انجام می‌دهد و کجا در دسترس است؟

۶-۲ - ایجاد گزارش به وسیله Report Wizard

از زبانه Create نوار ابزار گزینه Report Wizard را انتخاب نمایید . .

Access، ابزارهای متنوعی برای ایجاد گزارش دارد. این روش‌ها به اجمال عبارت است از :
Design View : گزارش خالی را در نمای Design باز می‌کند. در این نما می‌توانید عناوین، پانوشته‌ها، جزییات و خلاصه داده‌ها را اضافه کنید. زمانی که بخواهید گزارش را از ابتدا ایجاد کنید از این گزینه استفاده نمایید.

Report Wizard : با دنبال کردن قدم‌های فرایند تولید گزارش، یعنی انتخاب منابع داده (جدول‌ها/ پرس‌وجوها)، فیلدهای مورد نظر در گزارش و غیره گزارش تولید می‌شود. این روش ایجاد گزارش معمول‌ترین است.

Columnar Layout : گزارشی را ایجاد می‌کند که تمام فیلدهای جدول یا پرس‌وجوی انتخاب شده را دربر دارد. شبیه AutoReport است. با این تفاوت که از فضای صفحه استفاده بهتری دارد.

Tabular Layout : گزارشی تولید می‌کند که رکوردهای جدول/پرس‌وجو را در یک ردیف نشان می‌دهد و اندازه قلم را به گونه‌ای اختیار می‌کند که صفحه گزارش تنظیم شود. این نوع گزارش بهتر از نمای Datasheet ساده به نظر می‌آید و اغلب مفیدتر از Report‌های قبلی است (شکل ۶-۲).



شکل ۶-۲ - پنجره نمای Preview گزارش ایجادشده به وسیله Wizard

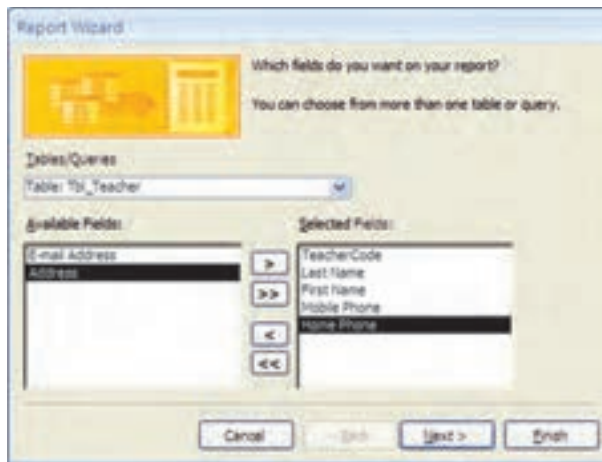
Access معمولاً نمی‌تواند عناوین فیلدها را در بالای گزارش به صورت صحیح قرار دهد. برای تنظیم دقیق همانطور که در ادامه خواهیم دید باید در نمای Design آن را اصلاح نمایید.

با کمک ویزارد گزارشی از لیست معلم‌های مدرسه و از نوع Tabular ایجاد کنید.

– پرس‌وجویی ایجاد کنید که تمام رکوردهای جدول معلمان را برگرداند و آن را Qry_Teachers نام‌گذاری کنید.

– در قسمت گزارش‌های بانک، دکمه New و سپس Report Wizard را انتخاب کنید و OK نمایید.

– در پنجره باز شده، در قسمت Tables/Queries جدول Qry_Teachers را انتخاب کنید.



– تمام فیلدها را با دکمه >> به لیست انتخاب شده اضافه کنید.
– قسمت Selected Field، فیلدهای انتخاب شده برای نمایش در گزارش را نشان می‌دهد. فیلدها به وسیله دکمه‌های > یا >> به لیست اضافه می‌شوند. همچنین می‌توان روی فیلد در لیست Available Field دو بار کلیک کرد.

نکته: در قسمت Tables/Queries می‌توان جدول‌های دیگر را نیز انتخاب کرده، فیلدهای مورد نظر را به فیلدهای انتخاب شده خود اضافه کرد. اگرچه گفتیم بهتر

است ابتدا پرس وجویی که داده‌های مورد نیاز را برمی‌گرداند، انتخاب کنید و سپس از این پرس و جو در گزارش استفاده نمایید.

– بعد از انتخاب فیلدهای مورد نیاز، دکمه Next را کلیک کنید.

– سایر مراحل را بدون تغییر تأیید کنید و در انتها Finish را انتخاب کنید.

– گزارش را در حالت Design یا Preview مشاهده کنید(شکل زیر).

RegisterCode	RegisterDate	StudentCode	StudentName
1388	1388	1388	1388
1389	1389	1389	1389
1390	1390	1390	1390
1391	1391	1391	1391
1392	1392	1392	1392



در ویزارد طراحی گزارش، توضیح دهید Sort, Grouping و Layout/Style چه عملی انجام می‌دهند؟

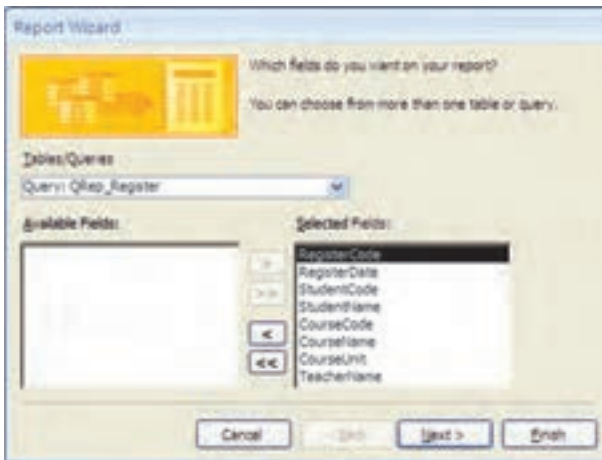
گزارش ثبت نام دانش آموزان در ترم را همان گونه که در طراحی در فصل اول مشخص شده است، ایجاد کنید.

ابتدا پرس وجویی به نام QRep-Register ایجاد می‌کنیم که اطلاعات مورد نیاز را در بر داشته باشد. SQL مورد نظر در زیر آمده است:

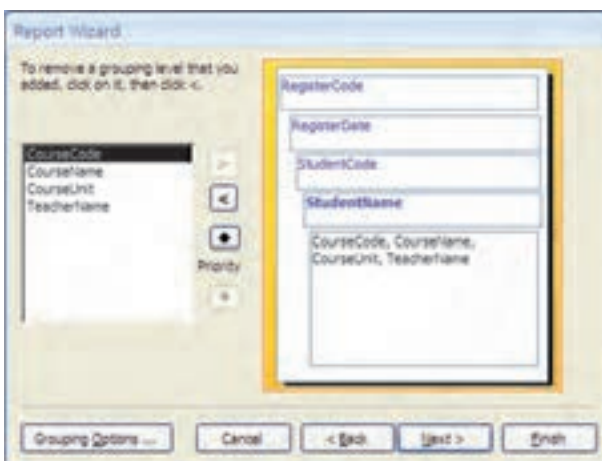
```
SELECT Tb1-Register.RegisterCode,Tb1-Register.RegisterDate,
Tb1-Register.StudentCode,[FirstName]+'+[LastName] AS
StudentName,
Tb1-RegisterItem.CourseCode,Tb1-Course.CourseName,
Tb1-Course.CourseUnit,
Tb1-Teacher.FirstName+' '+Tb1-Teacher.LastName AS
```

TeacherName
 FROM Tb1_Teacher INNER JOIN (Tb1_Course INNER JOIN
 (Tb1_Student INNER JOIN
 (Tb1_Register INNER JOIN Tb1_Register Item ON Tb1_Register.
 RegisterCode=
 Tb1_Register Item. RegisterCode)ON Tb1_Student. StudentCode=
 Tb1_Register. StudentCode) ON Tb1_Course. CourseCode =
 Tb1_Register Item. CourseCode)
 ON Tb1_Teacher. TeacherCode = Tb1_Course. TeacherCode

– ویزارد گزارش Report Wizard را اجرا می‌کنیم و پرس‌وجوی
 QRep_Register را انتخاب می‌کنیم.



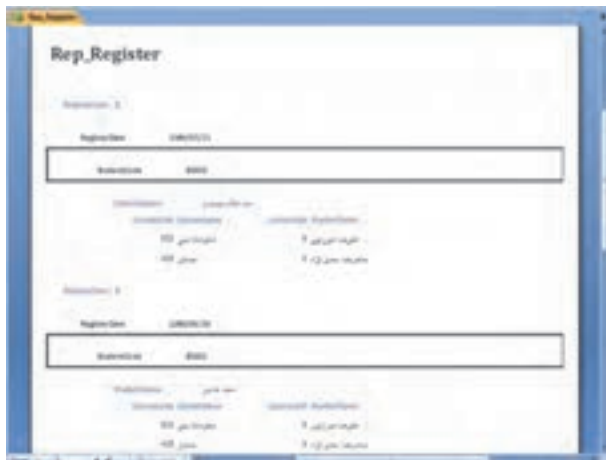
– تمام فیلدها را
 به وسیله دکمه >> به لیست
 اضافه می‌کنیم و دکمه
 Next را انتخاب می‌کنیم
 (شکل روبه‌رو).



– از پنجره باز شده
 ۴ مورد اول را انتخاب
 و دکمه Next را کلیک
 کنید.
 – در مرحله بعدی
 گزینه Outline را انتخاب
 کنید.

– در پنجره Style، پیش فرض Office را انتخاب کرده، دکمه Next را انتخاب کنید.


– نام گزارش را Rep_Register قرار دهید و دکمه Finish را انتخاب کنید.




– گزارش از لحاظ شکل ظاهر مشابه گزارش طراحی شده نیست، بنابراین گزارش را ذخیره می‌کنیم تا در نمای Design آن را تغییر دهیم.
– نمای Design این گزارش به صورت زیر است:



– در این نما خطوط اضافه و موارد غیر ضروری مثل تاریخ جاری سیستم را حذف می‌کنیم. (انتخاب مورد و زدن دکمه delete صفحه کلید)
– متن لاتین را به فارسی تغییر می‌دهیم. با انتخاب جعبه متن مورد نظر و کلیک

مجدد و تایپ فارسی متن مورد نظر را تغییر می‌دهیم. همچنین می‌توانیم از طریق پنجره مشخصه‌ها و از طریق دکمه  روی نوار ابزار، این کار را انجام دهیم. با استفاده از زبانه Format، گزینه نمایش یافته را تغییر دهیم.



- جعبه متن‌ها، مشابه روش‌هایی که در تغییر طراحی فرم دیدید، جابجا می‌شوند. یادآوری: مثلاً درگ کردن از طریق مربع بزرگ سمت راست جعبه متن که با انتخاب آن ظاهر می‌گردد.
- همچنین پس از انتخاب جعبه متن مورد نظر، تنظیم چیدمان متن درون آن (Align)، با کمک دکمه‌های  در نوار ابزار و براساس مورد، وسط‌چین، چپ‌چین یا راست‌چین قرار می‌گیرد. همچنین می‌توانیم از گزینه Text Align در زبانه Format پنجره مشخصه‌ها نیز این کار را انجام دهیم.
- برای افزودن امکانات گرافیکی، از ابزارهای موجود در زبانه Design استفاده می‌کنیم.



- گزارش خود را به کمک ابزارهای موجود، به صورت روبه‌رو تغییر می‌دهیم.

– سپس گزارش را ذخیره و در نمای Preview مشاهده می‌کنیم (شکل زیر).

– گزارش‌های لیست حضور و غیاب نیز به صورت مشابه ایجاد می‌شود.

گزارش کارنامه دانش‌آموزان را ایجاد می‌کنیم.

– این گزارش مشابه گزارش‌های قبلی ایجاد می‌شود، فقط کمی پیچیدگی در محاسبه معدل وجود دارد که به آن می‌پردازیم.

ابتدا پرس و جوی مورد نظر را ایجاد می‌کنیم. به نام QRep_Certificate و با

SQL زیر:

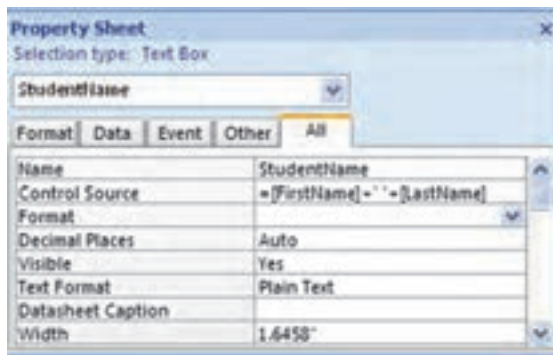
```
SELECT Tb1_Student.StudentCode, FirstName, LastName,
RegisterDate,
MajorName, Tb1_RegisterItem.CourseCode, CourseName, Course
Unit, Grade
FROM Tb1_Course INNER JOIN (Tb1_Major INNER JOIN ((Tb1_
Student INNER JOIN
Tb1_Register ON Tb1_Student.StudentCode= Tb1_Register.
StudentCode) INNER JOIN
Tb1_RegisterItem ON Tb1_Register.RegisterCode = Tb1_Register
Item.RegisterCode) ON
Tb1_Major.MajorCode = Tb1_Student.MajorCode) ON Tb1_
```

Course. CourseCode =

Tbl_Register Item. CourseCode

– با کمک ویزارد و این پرس و جو گزارش اولیه‌ای مشابه مثال قبلی و به نام Rep_Certificate ایجاد می‌کنیم.

– دقت نمایید که ما در پرس و جو نام و نام خانوادگی را ترکیب نکرده‌ایم، بنابراین باید در گزارش این کار صورت گیرد. برای این منظور فیلد FirstName را حذف کرده و نام فیلد LastName را به StudentName تغییر می‌دهیم سپس در پنجره مشخصه‌ها، زبانه Data، گزینه Control Source به صورت زیر تنظیم می‌کنیم:



– با کلیک راست روی فضای خالی صفحه گزارش، گزینه Sorting and Grouping را انتخاب می‌کنیم.

– در این پنجره گزینه Group Footer را مقدار Yes انتخاب می‌کنیم، تا پانویست گروه‌بندی نیز نمایش داده شود. جعبه متن‌های مربوط به جمع واحدها و محاسبه معدل باید در این قسمت قرار گیرد.

– از جعبه ابزار، دکمه Text box را انتخاب می‌کنیم و با درگ کردن، آن را در قسمت پانویشت گروه قرار می‌دهیم. مقدار Visible آن را در پنجره مشخصه‌ها No انتخاب می‌کنیم و مقدار Control Source آن را $\text{SUM}([\text{CourseUnit}]*[\text{Grade}])$ قرار می‌دهیم. این فیلد، حاصل جمع حاصل ضرب نمره در واحد درسی را براساس گروه‌بندی محاسبه می‌کند چرا که در پانویشت گروه قرار داده شده است. نام آن را در زبانه All یا Other، گزینه Name مقدار SumGradeByUnit قرار می‌دهیم.

– Text box دیگری را در قسمت پانویشت گروه، زیر ستون واحدها قرار می‌دهیم و مقدار Control Source آن را $\text{SUM}([\text{CourseUnit}])$ قرار می‌دهیم. این فیلد جمع کل واحدهای ثبت‌نام‌شده دانش‌آموز را برمی‌گرداند، زیرا در پانویشت گروه قرار گرفته است و جمع را براساس گروه محاسبه می‌کند. نام آن را به‌وسیله گزینه Name در پنجره مشخصه‌ها، مقدار SumUnit قرار می‌دهیم.

– Text box دیگری را در قسمت پانویشت گروه قرار می‌دهیم و مقدار Control Source آن را $[\text{SumGradeByUnit}]/[\text{SumUnit}]$ قرار می‌دهیم و در پنجره مشخصه‌ها گزینه Format آن را مقدار $\#\#\#$ قرار می‌دهیم. همچنین گزینه Border Style را مقدار Solid قرار می‌دهیم.

– طراحی گزارش را در شکل زیر مشاهده می‌کنید.



– نمای Preview گزارش به صورت زیر است.

نام و نام خانوادگی	تاریخ تولد	شماره کارت ملی	نمره
علیرضا محمدی	1382/05/10	0000000000	82
علیرضا محمدی	1382/05/10	0000000000	82

مثال ۴-۶



محیط طراحی گزارش‌ها مشابه محیط طراحی فرم است و ابزارهای محیط، منوها، کنترل‌ها، قابلیت‌ها و چگونگی استفاده از آن‌ها مشابه است. بنابراین از ذکر مجدد آن در این قسمت خودداری می‌کنیم.

خلاصه فصل

برای چاپ اطلاعات با قالبی زیبا و مناسب از گزارش استفاده می‌کنیم. گزارش نمای قابل چاپ را روی صفحه نمایش ارائه می‌دهد. می‌توانیم اطلاعات را در گزارش‌ها گروه‌بندی کنیم و گزارش‌های با جزئیات و یا اطلاعات خلاصه ایجاد کنیم. محیط ایجاد گزارش در Access، شبیه برنامه‌های گزارش ساز در محیط ویندوز است. مثلاً همان‌طور که در محیط Word، اطلاعات چاپی را در نمای Preview مشاهده می‌کردید و یا بخشی از آن را چاپ می‌کردید، در محیط Access نیز همان امکانات برای گزارش ایجاد شده وجود دارد. اشیای بانک مشابه کنترل‌های فرم هستند و همان عملیات مشخصات ظاهری کنترل‌ها، جابه‌جایی آن‌ها و ... روی آن قابل انجام است.

خودآزمایی

- ۱- گزارش چیست؟
- ۲- چرا از گزارش استفاده می‌شود؟
- ۳- روش‌های ایجاد گزارش را توضیح دهید.
- ۴- به نظر شما کدامیک از روش‌های ساخت گزارش مناسب‌تر است و چرا؟
- ۵- کنترل‌های گزارش را بررسی کنید، آیا اختلافی بین آن‌ها و کنترل‌های فرم وجود دارد؟
- ۶- یک گزارش نموداری چگونه ایجاد می‌شود؟

فصل ۷

نرمال سازی

پس از شناسایی موجودیت‌ها و رسم ER، صفات مورد نیاز و ارتباط بین موجودیت‌ها، می‌باید جدول‌های (روابط) مورد نیاز طراحی شوند. یک تصور نادرست این است که هرچه صفات جدول بیشتر باشد، از اکمال و جامعیت بالاتری برخوردار است، حال آن که گزینش درست صفات، از بروز برخی مشکلات در بانک‌های اطلاعاتی اجتناب می‌کند. نارسایی‌ها در طراحی و یک طراحی نامناسب، باعث ایجاد پدیده «آنومالی» می‌گردد.

آنومالی، در سه محور قابل طرح و بررسی است:

الف - «انجام‌ناپذیری» یکی از عملیات در بانک.

ب - «بروز تبعات نامطلوب» در پی انجام یک عملیات مبنایی.

ج - «فزون‌کاری» برای انجام یک عملیات مبنایی.

قبل از شرح موارد آنومالی، ویژگی‌های رابطه و چند مفهوم مقدماتی دیگر ذکر می‌شود، سپس در قالب یک مثال آنومالی و راه‌های رفع آن گفته خواهد شد.

اهداف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

- ویژگی‌های رابطه را تعریف کند.
- وابستگی تابعی و وابستگی تابعی کامل را شرح دهد.
- آنومالی را تعریف کند.
- ویژگی‌های تجزیه مناسب را شرح دهد.
- مفهوم سطوح نرمال 1NF، 2NF و 3NF را توضیح دهد.

۷-۱- ویژگی‌های رابطه

رابطه که در بانک‌های اطلاعاتی رابطه‌ای^۱، به صورت جدول پیاده‌سازی می‌شود دارای ویژگی‌های زیر است:

الف - منظم نیست، به این معنی که ترتیب صفات، در هنگام تعریف بی‌اهمیت است و سطرهای جدول^۲ نظم ندارد.

ب - دارای صفات تک‌مقداری است یعنی به اجزای معنی‌دار تقسیم نمی‌شود. به عنوان مثال اگر مقدار شماره دانش‌آموزی برابر ۸۵۱۷۴۳ باشد و ۸۵ نشان‌دهنده سال تحصیلی باشد و در پرس‌وجوها بخواهیم دانش‌آموزانی که در سال ۱۳۸۵ ثبت‌نام کرده‌اند را مشخص کنیم، فیلد «شماره دانش‌آموزی» یک فیلد تک‌مقداری نیست.

ج - جدول، سطرهای تکراری ندارد.

۷-۲- وابستگی تابعی

شبهه تعریف تابع در ریاضیات، در رابطه R ، صفت y با صفت x وابستگی تابعی دارد اگر به ازای هر مقدار x ، تنها یک مقدار y وجود داشته باشد.

وابستگی تابعی y با x را به صورت $x \rightarrow y$ نشان می‌دهیم^۵.

در رابطه R_1 ، وابستگی تابعی $A \rightarrow B$ و $B \rightarrow C$ را بررسی کنید:

R_1

A	B	C
a_1	b_1	c_1
a_2	b_2	c_3
a_1	b_1	c_2
a_3	b_4	c_2
a_5	b_1	c_1

۱- Relational Data Base

۲- به سطرهای جدول در این مدل داده‌ای اصطلاحاً تاپل گفته می‌شود.

۳- Functional Dependency

۴- x و y می‌توانند صفات خاصه ساده یا مرکب باشند.

۵- A را اصطلاحاً دترمینان گویند.

پاسخ: وابستگی $A \rightarrow B$ برقرار است. زیرا به ازای هیچ مقدار مساوی از A، دو مقدار متفاوت از B وجود ندارد. اما وابستگی $B \rightarrow C$ برقرار نیست. زیرا که به ازای b_1 ، دو مقدار c_1 و c_2 در R وجود دارد. ($B \rightarrow C$)

وابستگی تابعی بیان کننده قواعد محیط عملیاتی است. این قواعد را می توان در نمودار «وابستگی تابعی» یا نمودار FD نمایش داد.

فرض کنید قواعد زیر در یک محیط عملیاتی برقرار است. نمودار FD متناظر را رسم کنید. یک جدول با مقادیر دلخواه متناظر و با نام FIRST ایجاد کنید.

قاعده ۱: هر ناشر، تعدادی کتاب منتشر می کند.

قاعده ۲: هر ناشر از یک کتاب، شمارگان مشخصی منتشر می کند.

قاعده ۳: هر ناشر در یک شهر دفتر دارد.

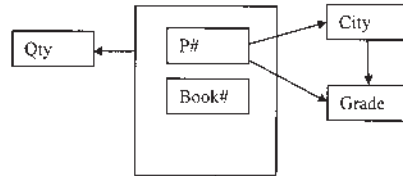
قاعده ۴: هر ناشر دارای یک رتبه صنفی است.

قاعده ۵: ناشران یک شهر دارای یک رتبه صنفی هستند.

پاسخ: با توجه به قواعد بالا داریم:

$P\# =$ شماره ناشر، $Book\# =$ شماره کتاب، $Qty =$ شمارگان، $City =$ شهر و $Grade =$ رتبه صنفی ناشر)

- 1) $P\# \rightarrow Book\#$
- 2) $(P\#, Book\#) \rightarrow Qty$
- 3) $P\# \rightarrow City$
- 4) $P\# \rightarrow Grade$
- 5) $City \rightarrow Grade$



P#	Book#	Qty	City	Grade
P_1	b_1	1000	c_1	40
P_2	b_4	3000	c_2	60
P_3	b_3	2000	c_1	40
P_1	b_2	3000	c_1	40
P_4	b_3	4000	c_3	50
P_4	b_5	3500	c_3	50

در یک محیط عملیاتی چهار قاعده بیان کنید و نمودار FD متناظر را رسم کنید.

تمرین ۱-۷:



تمام صفات یک موجودیت با صفت کلید آن، وابستگی تابعی دارند.

اگر $A \rightarrow B$ و $B \rightarrow C$ برقرار باشد، آیا $A \rightarrow C$ برقرار است؟
پاسخ: فرض می‌کنیم $A \not\rightarrow B$ در این صورت داریم:

$a_1 \dots c_1$

$a_1 \dots c_2$

$a_1 \ b_1 \ c_1$

$a_1 \ b_1 \ c_1$

دو حالت روبه‌رو قابل تصور است:

$a_1 \ b_1 \ c_2$

$a_1 \ b_2 \ c_2$

در هر دو حالت با تناقض در فرض مواجه می‌شویم. در قسمت (الف) $B \not\rightarrow C$

و در قسمت (ب) $A \not\rightarrow B$ در نتیجه طبق برهان خلف داریم: $A \rightarrow C$.

مثال ۳-۷

۷-۳-۲- وابستگی تابعی کامل

صفت y ، با صفت x وابستگی تابعی کامل دارد ($x \Rightarrow y$) اگر:

۱- y با x وابستگی تابعی داشته باشد.

۲- y با هیچ زیرمجموعه x وابستگی تابعی نداشته باشد.

با توجه به R2 تحقیق کنید، آیا وابستگی تابعی کامل بین C و (A و B) برقرار است؟

R^۲

A	B	C
a ₁	b ₁	c ₁
a ₂	b ₁	c ₃
a ₁	b ₂	c ₂

پاسخ: شرط اول برقرار است. یعنی: $(A, B) \rightarrow C$

شرط دوم نیز برقرار است. زیرا: $A \nrightarrow C ((a_1, c_1), (a_1, c_2))$

$B \nrightarrow C ((b_1, c_1), (b_1, c_2))$

در نتیجه: $(A, B) \Rightarrow C$



اگر x صفت ساده باشد و وابستگی تابعی بین x و y برقرار باشد $(x \rightarrow y)$ ، وابستگی تابعی کامل $(x \Rightarrow y)$ همواره برقرار است.

۴-۷- شرح آنومالی‌ها

همان‌طور که گفتیم آنومالی سه وجه دارد. اینک با در نظر گرفتن رابطه مثال ۲-۷ آنومالی را توضیح می‌دهیم.

۱- درج: در آنومالی ناشی از درج، تمام یا بخشی از کلید اصلی تعریف نشده و نامعین است. طبق یکی از قواعد عام بانک اطلاعاتی، مقدار کلید اصلی باید کاملاً مشخص باشد. فرض کنید می‌خواهیم رکورد $\langle P_{10}, 1000, c_4, 10 \rangle$ را درج کنیم. این درج امکان‌پذیر نیست. زیرا مشخص نیست ناشر چه کتابی را منتشر کرده است.

۲- حذف: در آنومالی نوع دوم پس از انجام یک عمل، عوارض نامطلوب داریم. فرض کنید می‌خواهیم رکورد $\langle P_2, b_4, 3000 \rangle$ را حذف کنیم. این حذف اگرچه شدنی است اما رتبه صنفی ناشران

شهر c_2 (مقدار 6°) ناخواسته حذف می‌شود (دقت کنید فعلاً تنها یک ناشر ساکن شهر c_2 است).
۳- به‌هنگام‌سازی: این نوع آنومالی، موجب فزون‌کاری می‌شود. در فزون‌کاری به‌ازای یک عمل مبنایی چندین عملیات صورت می‌گیرد. فرض کنید قرار است رتبهٔ صنفی ناشران ساکن شهر c_3 از 5° به 7° تغییر کند. این عمل موجب به‌هنگام‌سازی منتشرشونده^۱ (فزون‌کاری) در سیستم می‌شود، یعنی در بیش از یک سطر جدول باید تغییر صورت گیرد.

همان‌طور که مشاهده می‌شود، رابطهٔ FIRST (مثال ۲-۷) دارای آنومالی است. در یک بیان غیردقیق، علت آنومالی‌ها، «اختلاط اطلاعاتی» است. به این معنی که اطلاعات نشر و اطلاعات پایه‌ای ناشر در یک رابطه (جدول) با هم جمع شده است. برای ایجاد یک طراحی مناسب از رابطه‌ها باید آن‌ها را از لحاظ سطوح نرمال بررسی کنیم و در صورت نرمال نبودن در آن سطح، آن رابطه را با «تجزیه» اصلاح نماییم. در این صورت از بروز آنومالی جلوگیری می‌شود. اینک مناسب است سطوح نرمال رابطه را با بیان دقیق‌تر مورد بررسی قرار دهیم.

۵-۷- سطوح نرمال

روابط از سطح غیرنرمال تا سطح نرمال ۵ قابل تبیین و بررسی است. سطوح نرمال عبارتند از:

۱- سطح نرمال اول 1NF

۲- سطح نرمال دوم 2NF

۳- سطح نرمال سوم 3NF

۴- سطح نرمال 2 BCNF

۵- سطح نرمال چهارم 4NF

۶- سطح نرمال پنجم 5NF

در این کتاب تا سطح نرمال 3NF را بررسی می‌کنیم. چرا که رابطه در این سطح، از بسیاری آنومالی‌ها جلوگیری می‌کند.

۱-۵-۷- سطح نرمال اول: رابطهٔ FIRST در مثال ۲-۷ را در نظر بگیرید. تمام فیلدها به نحوی هستند که مقادیر آن‌ها به‌صورت منطقی قابل تقسیم نیست. به هر یک از این فیلدها، یک تک‌مقداری گوئیم. اگر تمام فیلدها تک‌مقداری باشد، رابطه در سطح نرمال اول قرار دارد.

۱- Propagated Update

۲- به احترام نام Boyce و Codd نامیده شده است. Boyce-Codd Normal Form (BCNF)

رابطه R در سطح نرمال اول (INF) است. اگر تمام فیلدهای آن در هر سطر جدول، تکمقداری باشد.

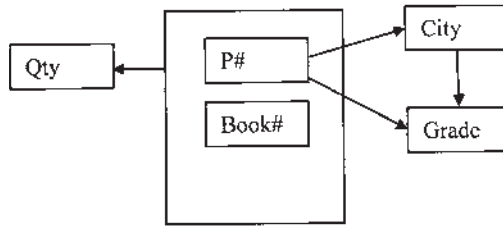
رابطه کارمند را در نظر بگیرید، آیا این رابطه در سطح INF است؟

نام	شماره شناسنامه	میزان تحصیلات	کد پرسنلی
علی علوی	۲۴۳	دیپلم	۱۱۰
ستایش یمقانی	۲۷۱۹	کارشناسی	۱۴۰
رضا قاسمیان	۵۹۳	کاردانی	۱۳۰
اسماعیل میرزایی	۹۰۹	کارشناسی	۱۷۰
علیرضا مسچی	۷۱۴	کارشناسی	۱۹۰

پاسخ: فیلد نام، قابل تجزیه به دو فیلد نام و نام خانوادگی است. اگر بخواهیم نام خانوادگی کارمندان را از رابطه به دست آوریم، فیلد نام را تجزیه کرده ایم؛ بنابراین رابطه غیرنرمال است و در سطح INF قرار ندارد. شکل نرمال INF آن به صورت زیر است:

نام	نام خانوادگی	شماره شناسنامه	میزان تحصیلات	کد پرسنلی
علی	علوی	۲۴۳	دیپلم	۱۱۰
ستایش	یمقانی	۲۷۱۹	کارشناسی	۱۴۰
رضا	قاسمیان	۵۹۳	کاردانی	۱۳۰
اسماعیل	میرزایی	۹۰۹	کارشناسی	۱۷۰
علیرضا	مسچی	۷۱۴	کارشناسی	۱۹۰

۲-۵-۷- سطح نرمال دوم: در قسمت‌های قبل با تعریف وابستگی تابعی کامل آشنا شدیم. رابطه وقتی در سطح INF باقی می‌ماند که وابستگی تابعی کامل بین حداقل یک فیلد با کلید اصلی نقض شود. به عنوان مثال نمودار FD برای رابطه FIRST را در نظر بگیرید:



کلید اصلی در این رابطه، صفت مرکب (P#, Book#) است. از آنجایی که تمام فیلدها، با کلید FD دارند، داریم: $(P\#, Book\#) \rightarrow City$
 برای برقراری وابستگی تابعی کامل باید داشته باشیم:

$P\# \twoheadrightarrow City$

$Book\# \twoheadrightarrow City$

$P\# \rightarrow City$ اما همان‌طور که در نمودار FD قابل مشاهده است، داریم:

در نتیجه FD کامل بین City و $(\#Book, \#P)$ نقض شده است.

اینک می‌توانیم دلیل آنومالی‌های رابطه FIRST را بهتر دریابیم. نقض FD کامل.

رابطه R در سطح نرمال دوم (2NF) است، اگر:
اولاً: 1NF باشد.
ثانیاً: صفت غیرکلید، ویژگی وابستگی تابعی کامل را با کلید نقض نکند.

برای رفع آنومالی و افزایش سطح نرمال رابطه، می‌باید رابطه FIRST را تجزیه نماییم. برای این کار

FIRST را به صورت زیر تجزیه می‌کنیم: $SECOND(P\#, City, Grade)$

$PB(P\#, Book\#, Qty)$

SECOND			PB		
P#	City	Grade	P#	Book#	Qty
P ₁	c ₁	40	P ₁	b ₁	1000
P ₂	c ₂	60	P ₂	b ₄	3000
P ₃	c ₁	40	P ₃	b ₃	2000
P ₄	c ₃	50	P ₁	b ₂	3000
			P ₄	b ₃	4000
			P ₄	b ₅	3500

ملاک تجزیه رابطه: تجزیه رابطه R به رابطه‌های R_1 و R_2 می‌باید به نحوی باشد که پیوند دو رابطه R_1 و R_2 را ایجاد کند و تاپلی (رکورد یا سطری) کم و زیاد نشود. از طرف دیگر تجزیه R می‌باید، وابستگی‌های تابعی را حفظ کند.

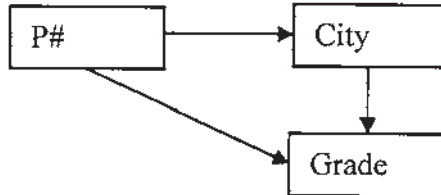
تمرین ۲-۷: آیا معیارهای تجزیه مناسب در تجزیه FIRST لحاظ شده است؟

رابطه SECOND، در سطح دوم نرمال است. زیرا اولاً: INF است و ثانیاً: وابستگی تابعی کامل نقض نشده است. اما این رابطه نیز دارای آنومالی می‌باشد.

تمرین ۳-۷: آنومالی‌های SECOND را ذکر کنید.

۳-۵-۷ سطح نرمال سوم: رابطه SECOND را در نظر می‌گیریم. نمودار FD آن به صورت زیر است:

1. $P\# \rightarrow City$
2. $City \rightarrow Grade$
3. $P\# \rightarrow Grade$



همان‌طور که در مثال ۳-۷ دیدیم، از دو وابستگی تابعی اول و دوم، می‌توان وابستگی تابعی سوم را نتیجه گرفت، یعنی Grade وابستگی تابعی با واسطه با $P\#$ از طریق City دارد. علت بروز آنومالی‌های SECOND نیز همین عامل یعنی وابستگی تابعی با واسطه است.

رابطه R در سطح نرمال سوم (3NF) است. اگر:
اولاً: 2NF باشد.

ثانیاً: هر صفت غیر کلید با کلید اصلی، وابستگی تابعی با واسطه نداشته باشد.

برای رفع آنومالی‌های آن، می‌باید این رابطه تجزیه شود. بنابراین SECOND را به رابطه‌های CG و PC تجزیه می‌کنیم:

PC(P#,City)

CG (City, Grade)

PC		CG	
P#	City	City	Grade
P ₁	c ₁	c ₁	40
P ₂	c ₂	c ₂	60
P ₃	c ₁	c ₃	50
P ₄	c ₃		

با حذف آنومالی‌های SECOND، روابط جدید نه تنها در سطح 3NF قرار می‌گیرند بلکه در سطح بالاتر یعنی BCNF نیز قرار می‌گیرند که ما به بررسی آن نمی‌پردازیم.

۱- آیا تجزیه انجام شده، ضوابط یک تجزیه مناسب را دارا می‌باشد؟
 ۲- حالت‌های دیگر تجزیه SECOND را در نظر بگیرید. آیا این تجزیه‌ها مناسب هستند؟

تمرین ۴-۷:

اگرچه نرمال‌سازی، آنومالی‌ها را از بین می‌برد اما در پرس‌وجوها، به علت نیاز به پیوند رابطه‌ها (جدول‌ها) سرعت انجام پرس‌وجو کاهش می‌یابد.



پس از اعمال سطوح نرمال و تجزیه جدول‌ها، سرعت بازیابی اطلاعات مورد نیاز کاربران بررسی می‌شود. در صورت عدم کارایی پرس‌وجوهای مورد نظر، و بنابر ملاحظات ممکن است جدول‌ها مجدداً ترکیب گردند. چه ملاحظاتی در این مورد مطرح است؟

ادگار فرانک تد کاد (۲۳ آگوست ۱۹۲۳-۱۸ آوریل ۲۰۰۳)



کاد یک دانشمند علوم کامپیوتر بود. وی نقش به‌سزایی در بنیان‌گذاری تئوری بانک رابطه‌ای داشت و می‌دانیم که این مدل رایج‌ترین مدل داده‌ای بانک اطلاعاتی است. زمانی که برای IBM فعالیت می‌کرد، مدل رابطه‌ای را برای مدیریت بانک اطلاعاتی بنا کرد. اگر چه نقش ارزشمند بسیاری در سایر پیشرفت‌های علوم کامپیوتر داشت ولی مدل رابطه‌ای وی از تئوری‌های مهم و به‌یادماندنی او است.

کاد در پورتلند انگلستان متولد شد. او ریاضی و شیمی را در دانشکدهٔ اکستر و آکسفورد گذراند. در سال ۱۹۴۸، به نیویورک رفت و به عنوان برنامه‌نویس با شرکت IBM کار کرد. کاد دکترای علوم کامپیوتر خود را از دانشگاه میشیگان دریافت کرد و دو سال بعد مجدداً در دفتر پژوهشی IBM فعالیت خود را آغاز کرد.

دو دهه (۱۹۶۰ و ۱۹۷۰) را بر روی تئوری‌های مدیریت داده کار کرد و در سال ۱۹۷۰ مقالهٔ خود را در مورد مدل رابطه‌ای برای بانک‌های دادهٔ مشترک و بزرگ^۱ منتشر کرد.

کاد بر توسعه و بسط مدل رابطه‌ای خود ادامه داد و با کریس دیت^۲ همکاری خود را شروع کرد. بر این اساس پس از کاد، یکی از مدل‌های نرمال را به نام Boyce-Codd نامیدند که به BCNF معروف است. پس از دهه ۱۹۸۰ که مدل رابطه‌ای به اوج شهرت رسید، بحث‌های کاد با تولیدکنندگان بانک اطلاعاتی منجر به پیشنهاد اصول ۱۲ گانهٔ وی شد که یک پایگاه دادهٔ رابطه‌ای باید از آن‌ها پیروی کند. با ایجاد زبان SQL، به نظر وی تئوری رابطه‌ای به نادرستی پیاده‌سازی شد و مخالفت‌هایش، پست و مقام شغلی وی را در IBM با مشکل مواجه کرد که شرکت را ترک گفت. پس از آن وی با کمک کریس دیت و دیگر همراهنانش، شرکتی مشاوره‌ای راه‌اندازی کرد.

کاد جایزهٔ تورینگ را در سال ۱۹۸۱ دریافت کرد و در سال ۱۹۹۴ وی به‌عنوان عضو افتخاری انجمن ماشین‌های محاسب منصوب شد.

۱- "A Relational Model of Data for Large Shared Data Banks"

۲- Chris Date

کریستوفر جی دیت ۱۹۴۱



دیت یکی از نویسندگان، محققان، مشاوران و متخصصان مستقل در زمینه فن آوری بانک داده رابطه‌ای است. زمانی که برای شرکت IBM کار می‌کرد، در زمینه برنامه‌ریزی و طراحی فنی محصولات شرکت چون SQL/DS و DB2 همکاری داشت. او زمانی با کاد در مدیریت پایگاه داده‌ها و مدل رابطه‌ای همکاری می‌کرد. در سال ۱۹۸۳ شرکت IBM را ترک کرد و یکی از توسعه‌دهندگان و پایه‌گذاران مدل رابطه‌ای شناخته می‌شود.

کتاب وی به عنوان مقدمه‌ای بر سیستم‌های بانک داده، تاکنون به هشتمین ویرایش خود رسیده است و کتابی مرجع در زمینه بانک‌های داده‌ای است و در صدها دانشکده و دانشگاه در سراسر دنیا تدریس می‌شود. وی همچنین نویسنده کتاب‌های بسیاری در زمینه مدیریت داده است.

خلاصه فصل

پس از طراحی مفهومی بانک اطلاعاتی و رسم نمودار ER، جداول، فیلدها و کلید اصلی آن‌ها تعیین می‌شوند. در برخی موارد جدول‌های ایجادشده دارای آنومالی هستند. یعنی برخی عملیات حذف، اضافه، ویرایش روی آن‌ها غیرممکن می‌شود، یا در صورت انجام اطلاعات مهم از دست می‌رود و یا با دشواری همراه است. یکی از دلایل ایجاد آنومالی اختلاط اطلاعاتی است. شش سطح نرمال داریم که سه سطح آن را بررسی کرده‌ایم. بیشتر جدول‌های نرمال در سطح سوم، در سطح بالاتری نیز قرار می‌گیرند. پس از تعیین جدول‌ها، هر جدول از نظر سطح نرمال بررسی شده و در صورت لزوم تجزیه می‌شود. هر سطح نرمال دارای شرایطی است. هر جدول غیرنرمال برای نرمال شدن باید تجزیه شود. عمل تجزیه ملاک‌هایی دارد که به برخی از آن‌ها در فصل اشاره کرده‌ایم. تجزیه‌شدن جدول به جدول‌های جدید که حافظ اطلاعات جدول قبلی باشد، آنومالی‌ها را از بین می‌برد ولی در مواردی سرعت بازیافت اطلاعات در پرس‌وجوها را کاهش می‌دهد.

- ۱- آنومالی به چه معنی است؟
- ۲- وابستگی تابعی و وابستگی تابعی کامل را با ذکر مثال توضیح دهید.
- ۳- سطح نرمال اول، دوم و سوم چه ویژگی‌هایی دارند؟
- ۴- محاسن و معایب نرمال‌سازی چیست؟
- ۵- دلیل دیگری ذکر کنید که نشان دهد وابستگی تابعی کامل بین صفت خاصه غیر کلید و کلید اصلی در FIRST، نقض شده است.
- ۶- آیا می‌توان از وابستگی تابعی $A \rightarrow D$ عبارت $(A, C) \rightarrow D$ را نتیجه گرفت؟
- ۷- رابطه Z را در نظر بگیرید و آن را به دو رابطه $X(A, B)$ و $Y(A, C)$ تجزیه کنید. آیا پیوند (JOIN) آن‌ها رابطه Z را ایجاد می‌کند؟

Z	A	B	C
	a_1	b_1	c_1
	a_1	b_2	c_2
	a_2	b_1	c_3
	a_3	b_3	c_4

۸- فرض کنید در رابطه (a_1, a_2, \dots, a_6) ، وابستگی‌هایی تابعی زیر وجود دارد. به نظر شما

کلید اصلی چیست؟ چرا؟

$$a_1 \rightarrow a_2$$

$$a_1 \rightarrow a_3$$

$$a_1 \rightarrow a_4$$

$$a_1 \rightarrow a_5$$

$$a_1 \rightarrow a_6$$

۹- در رابطه $R(A, B, C, D)$ وابستگی‌های تابعی زیر وجود دارد و (A, B) کلید است سطح

نرمال رابطه را مشخص کنید.

$$C \rightarrow D$$

$$A \rightarrow C$$

فصل ۸

مطالعه موردی

در فصل‌های قبل با امکانات محیط بانک اطلاعاتی آشنا شدیم. حال می‌خواهیم با یک مثال از این امکانات برای ایجاد یک پایگاه داده برای مدیریت اطلاعات کودکان تحت حمایت خود است. والدین، نگهداری روزانه کودک خود را به این مؤسسه محول کرده‌اند. روزانه غذاهای متنوعی توزیع می‌گردد، با این حال باید بدانیم آیا کودکی وجود دارد که نتواند از این تنوع غذایی استفاده کند و در این صورت، باید تدارک این گونه استثنائات نیز دیده شود. بنابراین اطلاعات نیاز روزانه غذایی و رژیمی کودکان باید ذخیره گردد، مشخصات فردی کودکان و والدین آن‌ها باید نگهداری شود و همچنین برای بهبود فعالیت مهدکودک، می‌توانیم زمان واکسیناسیون، پایش رشد (کنترل قد و وزن) را نیز نگهداری و برآورد کنیم و به والدین اطلاع دهیم و یادآوری کنیم.

هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

- یک بانک اطلاعاتی مطابق تجزیه و تحلیل انجام شده را ایجاد کند.
- در ایجاد بانک سطح نرمال سازی 3FN را رعایت کند.
- بانک اطلاعاتی تحلیل شده را توجیه کند.

۱-۸ - اطلاعات کودکان Children

زمانی که یک کودک برای اولین بار به مهد کودک آورده می‌شود، یک مشخصه منحصر به فرد می‌گیرد. هر کودک در یک تخت مخصوص به خود و در اتاقی خاص نگهداری می‌شود. کودکان نوپا می‌توانند از انواع غذاها استفاده کنند ولی نوزادان به برنامه غذایی خاص نیاز دارند. همچنین برخی کودکان ممکن است به غذاهای خاصی حساسیت داشته باشند و نتوانند برخی انواع مواد غذایی مثل توت‌فرنگی، تخم‌مرغ، گوجه‌فرنگی و ... را مصرف کنند. همچنین اطلاعات واکسیناسیون و آخرین تاریخ پایش رشد کودکان نیز باید نگهداری شود.

۲-۸ - اطلاعات والدین Parents

هر یک از والدین می‌تواند مراقبت یک یا چند کودک خود را به مهد کودک بسپارد. برخی والدین علاقه‌مند هستند کودک خود را در اتاق غیر خصوصی و مشترک نگهداری کنند تا با کودکان دیگر مرادده داشته باشند و همچنین هزینه آن ارزان‌تر باشد.

۳-۸ - تحلیل

مهارت و دانشی که در زمان تحلیل مورد نیاز است دانش کامپیوتری فرد طراح نیست، بلکه دانش درک مسأله است. توجه ما در این کتاب، تحلیل و شناخت سیستم نیست، بنابراین فرض می‌کنیم تحلیل قبلاً انجام شده است تا بتوانیم کاربرد Access را با هم بیاموزیم.

۱-۳-۸ - موجودیت‌ها Entity: این اصطلاح برای هر چیزی به کار می‌رود که باید اطلاعاتی در مورد آن ذخیره کنیم. موجودیت‌های دنیای واقع به جدول‌های بانک تبدیل می‌شوند.

در مثال، دو جدول Parents و Children قابل شناسایی است.

۲-۳-۸ - مشخصه‌ها Attribute: جزئیات بیشتری که موجودیت را در بر دارد.

- مشخصه Parents عبارت است از: اطلاعات تماس والدین، شناسه منحصر به فرد که به هر یک از اولیاء تخصیص می‌دهیم، نام و نام خانوادگی وی، همچنین اطلاعات آدرس، شهر، کشور، کدپستی و شماره تلفن وی که برای ما اهمیت دارد. به خاطر بسپارید که مشخصات کودک جزو مشخصه والدین نیست و به صورت مستقل به عنوان یک موجودیت مطرح است.

- مشخصه Children عبارت است از: شناسه کودک، نام، سن، جنسیت، تاریخ واکسیناسیون و نیازهای خاص رژیمی و ملاحظات تغذیه‌ای وی.

۳-۳-۸ - ارتباط بین موجودیت Relationship: برای تعیین ارتباط Relation بین

این دو موجودیت، شناسه جدول Parents را در جدول Children اضافه می‌کنیم.



- ۱- در زمان طراحی بانک، بهتر است شناسه‌های منحصر به فرد را تعیین و یا ایجاد کنیم و از شناسه‌های ترکیبی استفاده نکنیم. مثلاً می‌توانیم از نام و آدرس به صورت مشترک برای کلید استفاده کنیم ولی این کار ایجاد ارتباط بین دو جدول را مشکل می‌سازد.
- ۲- شاید بتوان از شماره کارت ملی و یا شماره حساب سبیا که منحصر به فرد است استفاده کرد.

مشخصه‌ها به فیلدهای جدول تبدیل می‌شوند. فیلدهای دو جدول مزبور در زیر نمایش داده شده‌اند. ارتباط بین آن‌ها نیز به وسیله ParentsId تعریف شده است. (Id نمایانگر عبارت Identifier به معنی شناسه است.)

یادآوری: Table، روش ساخت یافته نمایش داده‌ها است. فیلدها در Access به صورت عنوان ستون‌ها نمایش داده می‌شوند. جدول، مقادیر داده‌ها را برای نام، نام خانوادگی و ... نگهداری می‌کند. این مقادیر به صورت ردیف نمایش می‌یابند که در اصطلاح بانک داده، رکورد Record نامیده می‌شوند.

جدول ۱-۸ - فیلدهای جدول Parents

ParentsId	FirstName	LastName	Address	City	Country	PostCode	PhoneNumber
شناسه	نام	نام خانوادگی	آدرس	نام شهر	نام کشور	کد پستی	شماره تلفن تماس

شرح فیلد :

جدول ۲-۸ - فیلدهای جدول Children

ChildId	ParentsId	FName	Age	Sex	VacDate	RunNo	DietNotes
شناسه	شناسه والدین	نام	سن	جنسیت	تاریخ واکسیناسیون	وضعیت حضور	ملاحظات تغذیه‌ای

شرح فیلد :

۴-۸ - استاندارد نام‌گذاری اشیاء بانک

اعمال یک استاندارد برای نام‌گذاری مناسب فایل‌ها و اشیاء بانک، کار را قابل اعتماد می‌سازد، خصوصاً زمانی که فایل به اشتراک گذاشته می‌شود. مثلاً تمام اشیاء، توصیف‌کننده‌ای داشته باشند که در شروع نام قرار داده شود. استاندارد Access را می‌توانید در آدرس زیر و راهنمای Leszynski/Reddick پیدا کنید. مثلاً این که از کاراکتر فاصله در اسامی استفاده نکنید، ارتباط بین جدول‌ها با استفاده از اسامی مشابه فیلدهای متناظر ایجاد گردد و ...

http://www.microsoft-accesssolutions.co.uk/naming_conv.htm

۵-۸ - ایجاد پایگاه داده

فایل پایگاه داده‌ای که در این قسمت ایجاد می‌کنید، در واقع ظرف نگهدارنده (Container) کلیه اشیای بانک خواهد بود مثل: جدول‌ها، فرم‌ها و ...

- برنامه Microsoft Access را اجرا کنید.
- گزینه New را از منوی اصلی برنامه انتخاب کنید.
- در پانل سمت راست نام پایگاه داده را به CaseExample.accdb تغییر دهید، محل ذخیره‌سازی آن را نیز انتخاب نموده، دکمه Create را کلیک کنید.
- حال که فایل نگهدارنده ایجاد شد، می‌توان اشیاء (جدول‌ها، فرم‌ها، پرس‌وجوها و گزارش‌ها) را به آن اضافه کرد.

۶-۸ - ایجاد جدول‌ها

۱-۶-۸ - ایجاد جدول به وسیله Table Templates: چون اشیاء فرم‌ها، گزارش‌ها و پرس‌وجوها بر مبنای جدول‌ها ایجاد می‌شوند. ابتدا باید دو جدول را ایجاد کنیم.

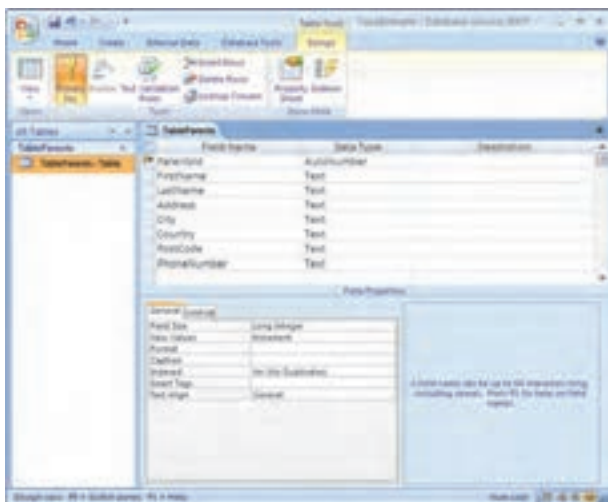
- از زبانه Create نوار ابزار، روی Table Templates کلیک کرده و Contacts را انتخاب نمایید.
- جدول ایجاد شده جدید را به نام Table Parents ذخیره نمایید.



شکل ۸-۱

۸-۷- اصلاح طراحی جدول ها

- جدول Table Parents را در نمای Design باز کنید.
- تغییرات زیر را روی ساختار فیلدهای جدول اعمال کنید.
- فیلد ID را به Parents Id تغییر نام دهید.
- جای فیلدهای First Name و Last Name را با هم عوض کنید.
- فیلدهای Home Phone، Business Phone، Job Title، Emaid Address، Company و Mobile Phone را حذف نمایید.
- فیلد Fax Number را به Phone Number تغییر نام دهید.
- نوع داده‌ای فیلد Address را از Memo به Text تغییر دهید.
- فیلد Zip/PostalCode را به PostCode تغییر نام دهید.



شکل ۸-۲


- فیلد State/Pravince را
- به Country تغییر نام دهید.
- چهار فیلد آخر را از جدول حذف نمایید.
- فیلد Phone Number را به انتهای لیست منتقل نمایید.

در نمای Design، می‌توانید نام، نوع فیلد، توضیحات و دیگر خصوصیات فیلد را تغییر دهید.

توجه شود که ParentsId از نوع داده AutoNumber است، یعنی هر زمان داده‌ای (رکورد هر یک از والدین جدید) اضافه شود، Access به‌طور خودکار، عددی منحصر به فرد به عنوان ParentsId ایجاد می‌کند و بدین ترتیب هر رکورد، کلید منحصر به فردی خواهد داشت.

همچنین در Access می‌توان مشخصه Input Mask فیلد را تعریف کرد. این مشخصه کمک می‌کند کاربر داده غلط وارد نکند. مثلاً جا افتادن رقمی از شماره تلفن و ...
– کلیک روی PostCode و انتخاب آن.

– کلیک در بخش Field Properties و انتخاب ردیف Input Mask. (باید زبانه general انتخاب شده باشد)

– روی دکمه  سمت راست آن کلیک کنید. پیغام ذخیرهٔ جدول را تأیید کنید.

– زمانی که Mask Wizard شروع شد، PostCode را انتخاب نمایید.

– در قسمت Try It، عبارت LLaoLL را تایپ کنید تا عبارت باید شامل دو حرف سپس دورقم/

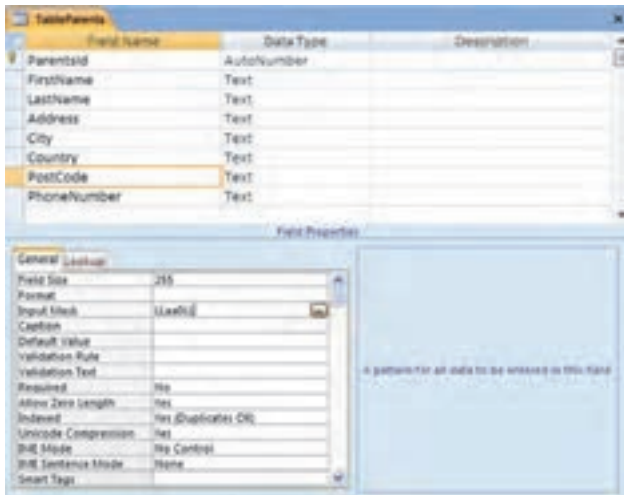
حرف و ارقام صفر تا ۹ و سپس دو حرف دیگر باشد (شکل ۳-۸).

– روی Finish کلیک کنید.

– از منوی اصلی Save را انتخاب کنید.

– در صورتی که کار با

طراحی جدول تمام شده باشد، آن را Close کنید. ولی ما به حالت Datasheet می‌رویم تا داده‌ها و رکوردهایی را در جدول اضافه کنیم.



شکل ۳-۸

عبارت‌های مجاز در Input Mask

کاراکتر	توضیحات
0	رقم (صفر تا ۹ و علامت [+] یا [-] مجاز نیست).
9	رقم یا خالی (ورود اطلاع ضروری نیست و علامت مثبت یا منفی مجاز نیست).
#	رقم یا فضای خالی (ورود اطلاع ضروری نیست. در زمان ویرایش علامت Space به صورت فضای خالی نمایش داده می‌شود ولی در زمان ذخیره‌سازی حذف می‌گردد. علامت مثبت و منفی مجاز است).
L	حروف (فارسی از الف تا ی و انگلیسی از A تا Z. ورود اطلاع ضروری است).
?	حروف (فارسی از الف تا ی و انگلیسی از A تا Z. ورود اطلاع ضروری نیست).
A	حروف یا رقم (ورود اطلاع ضروری است).
a	حروف یا رقم (ورود اطلاع ضروری نیست).
&	هر کاراکتری و یا فضای خالی (ورود اطلاع ضروری است).
C	هر کاراکتری و یا فضای خالی (ورود اطلاع ضروری نیست).
. , : ; - /	جداساز (برای محل رقم هزارگان، تاریخ، زمان و استفاده از کاراکترهای جداساز استاندارد در محیط ویندوز).
<	باعث می‌شود تمام کاراکترها به حروف کوچک تبدیل شوند.
>	باعث می‌شود تمام کاراکترها به حروف بزرگ تبدیل شوند.
!	باعث می‌شود، عبارت از راست به چپ نمایش یابد. کاراکترهایی که در این فیلد تایپ می‌شوند، فیلد را از چپ به راست پر می‌کنند. این علامت در هر جایی از عبارت می‌تواند قرار گیرد.
\	باعث می‌شود کاراکترها به صورت کاراکتر ثابت نمایش داده شوند (مثلاً \A به صورت A نمایش داده می‌شود).

۸-۸ - نمای Datasheet جدول

پس از تغییرات طراحی جدول، نمای Design را ترک کرده، به نمای Datasheet می‌رویم تا بتوانیم در این نما، داده‌هایی را به جدول اضافه کنیم (شکل ۸-۴).



شکل ۸-۴

ورود داده به جدول: مطابق شکل ۸-۵، داده‌ها را به جدول اضافه کنید. با کمک کلیدهای TAB روی فیلدها و از رکوردی به دیگری جابه‌جا شوید و یا بالعکس، از Shift+TAB استفاده کنید. پس از اتمام ورود داده‌ها، پنجره مربوطه را ببندید.



شکل ۸-۵

۹-۸ - استفاده از فرم‌ها

۹-۸-۱ - ایجاد فرم جدید ورود اطلاعات: با کمک فرم می‌توان اطلاعات وارد شده را کنترل کرد و از ثبت اطلاعات غلط جلوگیری کرد. مثلاً زمان ورود اطلاعات کودکان مهدکودک، می‌توانیم از جدول والدین استفاده و لیستی از آن‌ها ارایه کنیم. بدین ترتیب شناسه والدین را به‌طور صحیح به‌دست آورده و در جدول Children به همراه سایر اطلاعات کودک ذخیره نماییم. ابتدا فرمی بدون هیچ‌گونه کنترلی ایجاد می‌نماییم.

۹-۸-۲ - ایجاد فرم:

برای ایجاد فرم ابتدا جدول TableParents را انتخاب کنید سپس از زبانه Create نوار ابزار گزینه Form را انتخاب نمایید.

فرم جدید ایجاد شده را به نام FormParents ذخیره نمایید.

۳-۹-۸- ورود اطلاعات به وسیله فرم: FormParents را در نمای FormView باز کنید، برای درج اطلاعات جدید می توانید از نوار موجود در قسمت پایین فرم استفاده نمایید و یا با فشردن کلیدهای ترکیبی (+) Ctrl+Shift در وضعیت ثبت اطلاعات جدید قرار بگیرید.
 - اطلاعات نام و آدرس بیشتری را مطابق شکل ۶-۸ وارد نمایید.

43008765	ایران	تهران	تهران	مطمربها 7
(0261)-76564211	ایران	کرج	کرج	آبت 8
	ایران	تهران	تهران	مهدا 9
				(New)

شکل ۶-۸

برای این کار می توانید از فرم شکل ۷-۸ نیز استفاده کنید.

The screenshot shows a window titled 'FormParents' with a 'TableParents' header. The form contains the following fields and values:

- ParentId: 7
- FirstName: مطمربها
- LastName: بختی
- Address: جاده ساربان - تهران رود
- City: تهران
- Country: ایران
- PostCode: (empty)
- PhoneNumber: 43008765

شکل ۷-۸

- با دکمه Close فرم را ببندید.
 - جدول TableParents را باز کنید و بررسی کنید آیا اطلاعات اضافه شده است (شکل ۸-۸).

The screenshot shows a table with the following columns: ParentId, FirstName, LastName, Address, City, Country, PostCode, and PhoneNumber. The data rows are:

ParentId	FirstName	LastName	Address	City	Country	PostCode	PhoneNumber
1	علی	محمدی	تهران	تهران	ایران	449964	(+98) 21-229476
2	علی	محمدی	تهران	تهران	ایران	449964	(+98) 21-229476
3	علی	محمدی	تهران	تهران	ایران	449964	(+98) 21-229476
4	علی	محمدی	تهران	تهران	ایران	449964	(+98) 21-229476
5	علی	محمدی	تهران	تهران	ایران	449964	(+98) 21-229476
6	علی	محمدی	تهران	تهران	ایران	449964	(+98) 21-229476
7	مطمربها	بختی	جاده ساربان - تهران رود	تهران	ایران	430087	43008765
8	آبت	کاشانی	کرج	کرج	ایران	(0261)-76564211	
9	مهدا	برائینی	تهران	تهران	ایران		

شکل ۸-۸

یادآوری: به خاطر داشته باشید که فرم، یکی از راه‌های ورود داده به جدول است و تمام اطلاعات در جدول ذخیره می‌شود. زمانی که داده‌ای در فرم تایپ می‌شود، اطلاعات آن در جدول ذخیره می‌شود و نه در فرم مربوطه.

۱-۸ - روابط Relationship

انواع رابطه‌ها: سه نوع اصلی ارتباط وجود دارد:

● یک - به - یک (One-to-One): یک رکورد در یک جدول با فقط یک رکورد از جدول دیگر مرتبط است.

● یک - به - چند (One-to-Many): یک رکورد در یک جدول می‌تواند به رکوردهای زیادی از جدول دیگر مرتبط باشد.

● چند - به - چند (Many-to-Many): یک رکورد در یک جدول می‌تواند به یک یا چند رکورد جدول دیگر مرتبط باشد و یک رکورد آن جدول می‌تواند به یک یا چند رکورد جدول اولی مرتبط گردد.

ارتباط یک - به - یک با استفاده از یک جدول به جای دو جدول حل می‌شود. ارتباط چند - به - چند معمولاً به جدول سومی نیاز دارد که به عنوان رابط بین دو موجودیت به کار رود. برای توضیحات بیشتر در این خصوص به کتاب‌های پایگاه داده و بخش مفاهیم اولیه مراجعه کنید.

۱-۸ - افزودن جدول دیگر

اکنون به جدول دیگری نیاز داریم که موجودیت Children را ایجاد کند. چون جدول‌های نمونه Table Templates مناسب به نظر نمی‌رسند، باید خودمان جدول را ایجاد و طراحی کنیم. باید نام فیلد، نوع فیلد و سایر خصوصیات را تعریف کنیم.

نام فیلدهای جدول عبارت‌اند از:

ChildId, ParentsId, Name, Age, Sex, VacDate, RunNo و DietNotes

از توصیف فارسی نام فیلدها به‌عنوان (Caption) فیلد استفاده می‌کنیم. در این‌جا انواع داده و خصوصیات فیلدها را بررسی می‌کنیم:

- ChildId از نوع AutoNumber است. چرا که Access خود به‌طور اتوماتیک به هر رکورد

جدید مقدار منحصر به فردی را تخصیص می‌دهد. این فیلد به عنوان کلید اصلی (Primary key) شناخته می‌شود.

— ParentsId از نوع داده Number است و اندازه فیلد (Field size) آن Long Integer است. این موضوع بسیار مهم است چرا که باید بتوانیم Children و Parents را به هم مرتبط سازیم. ParentsId در جدول Parents، والدین را مشخص می‌کند و از نوع AutoNumber است. اگر بخواهیم این دو جدول را به هم مرتبط کنیم باید فیلد متناظر در جدول دیگر از نوع Long Integer باشد. در این حالت ParentsId در جدول Children، به عنوان کلید خارجی (Foreign key) شناخته می‌شود و کلید اصلی جدول Parents است.

— Age از نوع Number است. می‌توانیم اندازه فضایی که برای این فیلد ذخیره می‌شود را کاهش دهیم، بدین صورت که اندازه فیلد را Byte قرار دهیم. این نوع، امکان ذخیره تا مقدار ۲۵۵ را فراهم می‌کند و بیشتر کودکان بیش از ۱۰۰ سال زندگی نمی‌کنند و حد مقدار حتی بیشتر از کفایت است.

— Sex از نوع Text است. در قواعد کنترلی (validation rule) مقدار ("M"; "F") In قرار می‌دهیم. این قاعده به معنی آن است که مقدار وارد شده در این فیلد باید M یا F باشد. چون این مقدار یک حرف است، باید اندازه فیلد ۱ قرار داده شود. (مقدار پیش‌فرض اندازه فیلد text، ۵۰ بوده است).

— VacDate از نوع Date/Time در نظر می‌گیریم. (به خاطر داشته باشید که این نوع فقط مقادیر تاریخ میلادی را ذخیره می‌کند و می‌توان محاسبات ریاضی روی آن اعمال کرد. در صورتی که بخواهید تاریخ شمسی را ذخیره نمایید باید از نوع text و اندازه ۱۰ استفاده کنید ولی دیگر قادر به استفاده از توابع تاریخ و عملگرهای ریاضی روی آن نخواهید بود).

— RunNo عددی مثبت با اندازه فیلد Byte است. Access به‌طور اتوماتیک مقدار پیش‌فرض (Default Value) صفر را برای آن لحاظ می‌کند. ما از این خصوصیت در زمان ایجاد Query استفاده خواهیم کرد.

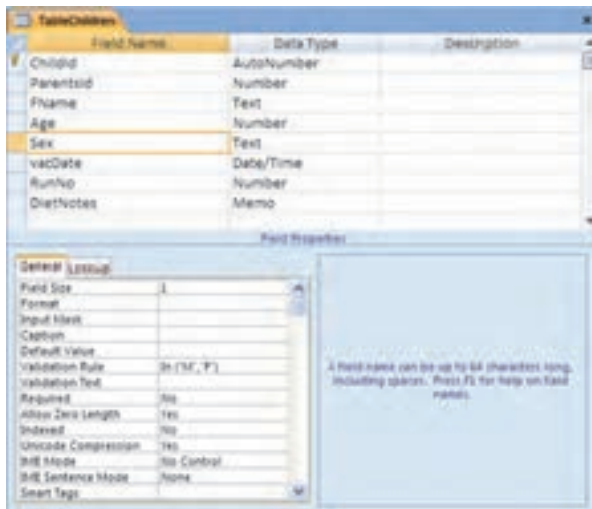
— DietNotes از نوع فیلد Memo است. چون نوع text می‌تواند حداکثر تا ۲۵۵ کاراکتر را دربرگیرد. این مقدار برای توضیحات ما کافی نیست بنابراین از نوع Memo استفاده می‌کنیم و کاربرد آن را نشان دهیم.

کلیدهای میانبر

- کلید TAB، جابه‌جایی بین ستون Field Name و Field Type.
- کلید F6، برای جابه‌جایی بین بخش Field Type/Field Name و بخش Field Properties.
- ALT+ down arrow باز کردن لیست. برای جابه‌جایی در لیست از کلیدهای Arrow استفاده کرده و با Enter مورد را انتخاب نمایید.

۸-۱۱-۸ - ایجاد جدول جدید با استفاده از نمای Design

با انتخاب گزینه TableDesign از زبانه Create نوار ابزار، فیلدهای جدول را مطابق شکل ۸-۹ تعریف نمایید.



شکل ۸-۹

- یادآوری می‌شود که در قسمت Field Properties، خصوصیات را به صورت زیر تغییر دهید:
 - برای فیلد Age، اندازه فیلد را Byte قرار دهید.
 - برای فیلد Sex، اندازه فیلد را ۱ و Validation Rule را به In ("M"; "F") قرار دهید.
 - برای فیلد RunNo، اندازه فیلد را Byte قرار دهید.

با ذخیره جدول به نام Access TableChildren، پیغامی را نمایش خواهد داد با انتخاب Yes، فیلد ChildId کلید اصلی می‌شود.



۱- دقت کنید که در این حالت فیلد ChildId چون از نوع AutoNumber است، کلید اصلی می‌گردد. در صورتی که فیلدی از نوع AutoNumber در جدول موجود نباشد، یک فیلد جدید با نام ID و از نوع AutoNumber ایجاد می‌گردد و کلید اصلی می‌شود.

۲- تعریف کلید اصلی به صورت زیر است: «یک یا چند فیلد که مقدار یا مقادیر آنها هر رکورد را به صورت منحصر به فرد تعریف می‌کند. در یک ارتباط کلید اصلی به رکوردهای مشخصی در جدول دیگر اشاره می‌کند. کلید اصلی که در جدول دیگر اشاره می‌شود، کلید خارجی نامیده می‌شود.»

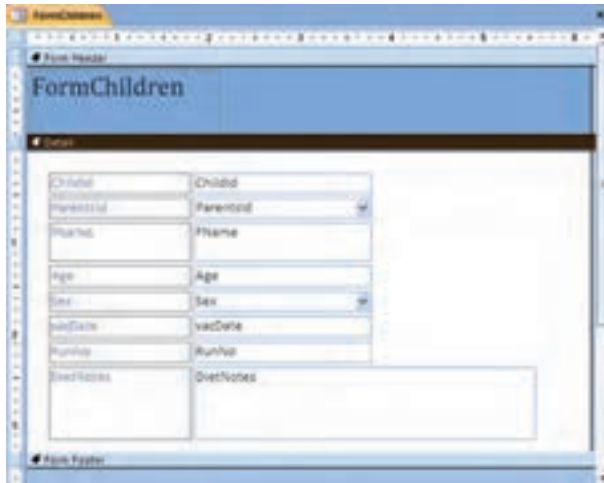
۲-۱۱-۸- ورود اطلاعات در جدول: براساس جدول TableChildren و با استفاده از Form Wizard یک فرم Columnar ایجاد نمایید. نوع ParentsId و Sex را به Combo Box تبدیل کنید (شکل ۱۱-۸). مشخصات تغییر Combo Boxها آورده شده است. سپس از فرم‌های ساخته شده برای ورود اطلاعات مطابق شکل ۱۰-۸ به جدول استفاده کنید.

ChildId	ParentName	Phone	Age	valDate	Burhu	DistNotes
1	محمد خورشیدی	021	2	2007-09-18	1	حساسیت به گرید لرننگی
3	علی‌اکبر خورشیدی	021	4	2003-03-21	3	مشکل در تالی زودرس
4	حمیده خورشیدی	021	5	2003-11-08	1	
6	حمیده خورشیدی	021	2	2006-09-13	1	
2	رضا خورشیدی	021	1	2009-03-10	1	درای آنتی بویوتام هر 6 ساعت
5	علی‌اکبر خورشیدی	021	3	2006-11-13	1	

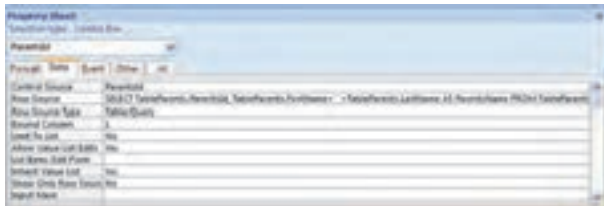
شکل ۱۰-۸



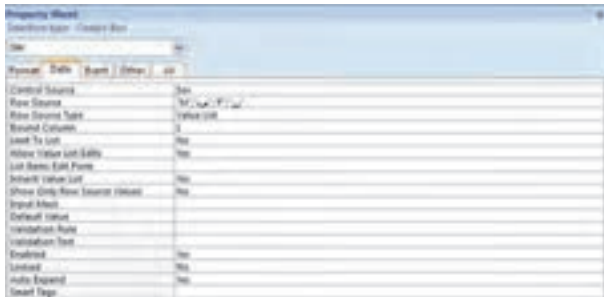
به خاطر داشته باشید که ممکن است AutoNumberهای تولید شده در جدول بانک شما متفاوت باشد. اگر اشتباه کرده باشید و رکوردی را حذف کنید، مقدار Autonumber آن رکورد نیز از دست می‌رود و عدد قبلی هرگز برای رکوردهای جدید ایجاد نمی‌گردد.



الف



ب



ج

شکل ۱۱-۸

۸-۱۲ - پرس و جوها Queries

یک امکان دیگر از اشیاء محیط Access، پرس و جو است. از پرس و جو برای مشاهده، تغییر و تحلیل داده‌ها به طرق مختلف استفاده می‌شود. همچنین از آن به عنوان یک منبع رکورد برای فرم‌ها و گزارش‌ها استفاده می‌شود.

۸-۱۲-۱ - قدم‌های ایجاد پرس و جو عبارت است از :

- مشخص کنید چه نوع پرس و جویی می‌خواهید ایجاد کنید.

– جدول‌هایی که داده‌های موردنظر را دربر دارند اضافه کنید.

– فیلدهایی که موردنظر است انتخاب کنید.

– معیاری که بر آن اساس Access مشخص می‌کند چه رکوردهایی و با چه محدودیتی نمایش داده شوند را مشخص کنید.

۲-۱۲-۸ – ساخت پرس‌وجو برای استفاده در گزارش: گزارشی نیاز داریم که نیازهای تغذیه‌ای کودکانی که در مهدکودک نگهداری می‌شوند را نشان دهد. می‌توانیم به سادگی از روی جدول TableChildren گزارش را بسازیم. در این حالت مریبان مهدکودک باید کودکانی که $RunNo > 0$ است یعنی امسال در مهدکودک ثبت نام شده‌اند را از لیست گزارش جدا کنند. معمولاً تعداد کل کودکانی که به مهد آمده‌اند به مراتب بیشتر از کودکان فعلی است و کار کنترل چشمی مشکل‌تر می‌شود. همچنین چنین گزارشی اطلاعات زاید زیادی دربر خواهد داشت که نه تنها گمراه‌کننده خواهد بود، بلکه ارتکاب اشتباه را نیز بیشتر می‌سازد.

در این حالت شرط مقدار $RunNo=0$ نشان می‌دهد آن کودک در مهدکودک نیست. فیلد RunNo زمانی که کودک در مهدکودک ثبت‌نام می‌شود به هنگام می‌شود و مقدار یک می‌گیرد و در پایان سال یا زمانی که کودک، مهد را ترک کند به هنگام شده و صفر می‌شود.

۳-۱۲-۸ – ایجاد یک پرس‌وجو با معیار ثابت:

– در زبانه Create نوار ابزار، گزینه Query Wizard را انتخاب نمایید.

– با انتخاب گزینه Simple Query Wizard پنجره باز شده را OK کنید.

– در قسمت Tables/Queries جدول TableChildren را انتخاب نمایید.

– از لیست Name, Available Fields و کلیک > را کلیک کنید تا این فیلد به قسمت

لیست Selected Fields منتقل شود. در مورد Sex, Age و DietNotes نیز همین گونه عمل کنید. سپس روی Next کلیک کنید.

– اطمینان یابید که Detail انتخاب شده باشد و کلیک Next را بزنید.

– عنوان Query Children in Runs را برای پرس‌وجو برگزینید. اطمینان یابید که

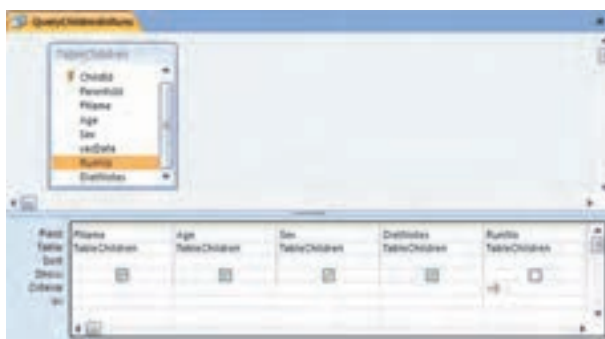
Modify Query Design انتخاب شده باشد و سپس روی Finish کلیک نمایید.

– پرس‌وجو باید در نمای Design مشاهده شود. در غیر این صورت با کمک دکمه View


روی نوار ابزار، نما را به Design تغییر دهید.

– در پنجره طراحی پرس‌وجو (شکل ۱۲-۸)، روی ردیف Criteria در ستون RunNo کلیک

کنید و عبارت > 0 را تایپ نمایید. توجه کنید که نباید این عبارت در [] قرار گیرد چرا که در این صورت، اکسس در مورد مقدار آن سؤال می‌کند.



شکل ۸-۱۲

– در نمای Design، دکمه میانبر  روی نوار ابزار را برای اجرای پرس‌وجو کلیک نمایید. نتیجه پرس‌وجو شبیه شکل ۸-۱۳ خواهد بود.
– پرس‌وجو را Close نمایید.

PName	Age	Sex	DietNotes
علی	2	M	حسیت به گرده فردگی
پریا	1	M	داروی آنتی بیوتیک هر 6 ساعت
مناشی	3	F	
آریسا	5	F	
فاطمه	2	F	

شکل ۸-۱۳

۸-۱۳ - گزارش‌ها Reports

کلاس بعدی اشیاء که ایجاد می‌کنیم، گزارش است. گزارش روشی برای نمایش محتوای داده‌های بانک است که مشابه صفحه‌های قابل چاپ باشد. به کمک آن مشخص می‌کنید کدام فیلدها چاپ شوند و چه داده‌هایی نمایش داده شوند. همچنین می‌توان خلاصه داده‌ها را نیز نمایش داد. Access امکان طراحی گزارش‌ها با الگوی خاص را فراهم می‌آورد مثلاً: محل سرتیتر، اسامی فیلدها، خود داده‌ها به همراه گرافیک و تصویر خاص Logo.

گزارش داده‌ها را از جدول‌ها، پرس‌وجوهای ایجاد شده و یا پرس‌وجوهایی که منحصرأ برای

گزارش و در زمان طراحی گزارش ایجاد می‌شوند، دریافت می‌کند.

بهرتر است ابتدا پرس‌وجوی موردنظر برای ایجاد گزارش ایجاد شود که تمام فیلدهای موردنیاز را در برداشته باشد، سپس از آن برای ایجاد گزارش استفاده شود.

۱-۱۳-۸- ایجاد یک گزارش و اصلاح آن : می‌توان از پرس‌وجوی QueryChildrenInRuns به عنوان مبنای ایجاد گزارش استفاده کرد. اگر چه گزارشی که به وسیله Report Wizard ایجاد می‌شود دقیقاً آن چیزی نیست که مورد انتظار ماست، ولی ما از آن برای ایجاد گزارش استفاده می‌کنیم و سپس طراحی آن را تغییر داده و اصلاح می‌کنیم.

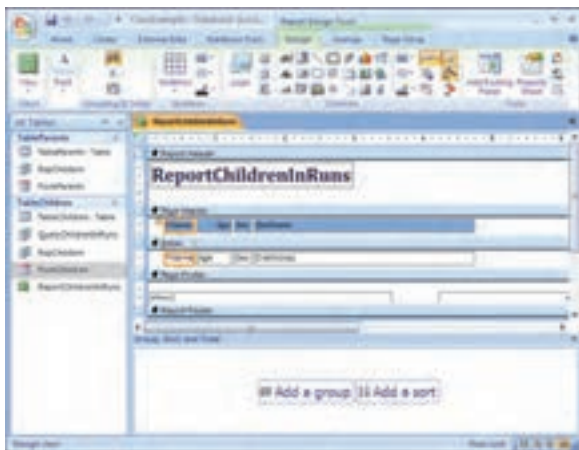
– در زبانه Create نوار ابزار، گزینه Reports Wizard را انتخاب می‌کنیم.

– در پنجره Report Wizard از لیست Tables/Queries، پرس‌وجوی Query Children In Runs را انتخاب کرده و تمام فیلدهای آن را به لیست سمت راست اضافه می‌کنیم.

– در قسمت تنظیمات Layout، گزینه Tabular را انتخاب می‌کنیم.

– گزارش را به نام ReportChildren InRuns ذخیره کرده و دکمه Finish را می‌زنیم.

– گزارش در نمای Print Preview باز می‌شود. با بستن نمای Preview، حالت Design View را برای گزارش انتخاب می‌کنیم.



شکل ۸-۱۴

۲-۱۳-۸- انتخاب اشیاء در گزارش : قبل از آن که کاری روی یک شیء انجام شود، باید آن شیء انتخاب گردد. زمانی که در حالت نمای Design قرار دارید با کلیک روی یک شیء می‌توانید آن را انتخاب کنید. در فرم‌ها و گزارش‌ها می‌توان همزمان چندین شیء را انتخاب کرد.



برای انتخاب اشیاء نزدیک به هم، کلید Shift را نگه داشته و به ترتیب روی اشیاء کلیک کنید.

برای انتخاب اشیایی که نزدیک هم نیستند، کلید Ctrl را نگه داشته و به نوبت روی اشیاء کلیک کنید.

یک شیء خصوصیات متنوعی دارد مثل Events، Data و

خصوصیات اشیاء مختلف، متفاوت هستند و تمام اشیاء، تمام خصوصیات را ندارند.

۳-۱۳-۸- مرتب‌سازی داده‌ها در گزارش: پس از آن که اشیاء را انتخاب کردید، می‌توانید عملیات موردنظر را روی آن‌ها اعمال کنید، مثلاً مرتب‌سازی داده‌ها یا تغییر خصوصیات مثل نوع قلم (Fonts)، رنگ‌ها (Colours)، چیدمان (Alignment).

داده‌های جدول TableChildren که پرس‌وجوی QueryChildrenInRuns ما مبتنی بر آن است، به ترتیب داده‌های وارد شده مرتب شده است. ولی شاید اطلاعات مرتب شده براساس سن کودک مناسب‌تر باشد.

– اطمینان داشته باشید که ReportChildrenInRuns در نمای Design باز باشد.

– روی دکمه Group & Sort در نوار ابزار کلیک نمایید. پنجره‌ای ظاهر می‌شود (شکل ۱۵-۸).

– در این پنجره، از لیست مقابل Sortby فیلد Age را انتخاب کنید. (در صورت انتخاب چند

فیلد برای مرتب‌سازی می‌توانید با استفاده از ▼ در لیست حرکت کنید.)



شکل ۱۵-۸

– پنجره را با دکمه (X) ببندید. (دقت کنید که فقط این پنجره را ببندید، نه کل گزارش را)
 – روی دکمه Print Preview در نوار ابزار کلیک کنید تا لیست ملاحظات غذایی کودکان فعلی
 مهدکودک را مشاهده کنید (شکل ۱۶-۸). این لیست براساس Age مرتب شده است.

Name	Age	Sex	BirthDate
علیا	2	M	۰۱/۰۱/۱۳۸۵
فاطمه	2	F	
علی	2	M	۰۱/۰۱/۱۳۸۵
سناپل	3	F	
فریده	5	F	

شکل ۱۶-۸

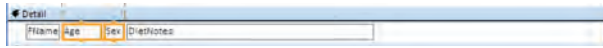
۴-۱۳-۸ – تغییر خصوصیات شیء: یکی از تغییرات مفید و تأثیرگذار در ظاهر گزارش، خصوصیت Alignment فیلدهای Age، Sex و name است. این خصوصیت قالب نمایش اطلاعات در این فیلدها را تغییر می‌دهد و از نوع Format است.



اگر پنجره خصوصیات مشاهده نمی‌شود. Properties را از کلیک راست ماوس انتخاب نمایید.

برای تغییر یک خصوصیت، روی فیلد کلیک کنید. در پنجره Properties، برای برخی خصوصیات علامت ► نمایش داده می‌شود. بدین معنی که می‌توانید از یک منو، مقدار خصوصیت را انتخاب کنید.

– اطمینان یابید که گزارش در نمای Design باز شده باشد. در قسمت detail گزارش، فیلدهای Sex، Age را انتخاب نمایید (شکل ۸-۱۷).



شکل ۸-۱۷

– در پنجره Property sheet، خصوصیت Text align (در زبانه Format) را انتخاب و مقدار آن را Left قرار دهید. یا از دکمه‌های روی نوار ابزار، چیدمان نمایش متن را تغییر دهید و فونت مناسب فارسی را انتخاب کنید.

– چیدمان قسمت عنوان را وسط‌چین Center قرار دهید و عنوان آن را به «گزارش ملاحظات غذایی کودکان» تغییر دهید. طول اندازه آن را بزرگتر کنید و فونت آن را مناسب فونت فارسی تنظیم کنید.



برای نمایش بهتر متن‌های فارسی باید فونت فارسی مناسب نصب شود. پس از نصب می‌توانید از هر کدام که در نمای Preview گزارش بهتر باشد و همچنین در چاپ نیز کیفیت مناسبی داشته باشد، استفاده کنید. ما از Nazanin استفاده کرده‌ایم.

– گزارش را ذخیره نمایید. سپس با دکمه Preview آن را مشاهده کنید (شکل ۸-۱۸).



شکل ۸-۱۸

۱۴-۸- اضافه شدن نیازهای جدید

تحلیل سیستم در دو بخش طراحی منطقی (logical) و طراحی فیزیکی (physical) صورت می‌گیرد. همان‌طور که در بیشتر مواقع نیز رخ می‌دهد، مشتری نیازهای خود را در ابتدا و شروع پروژه به طور کامل بیان نمی‌کند و حتی شاید در ابتدا آن را نداند.

کارفرمای مهدکودک ما پس از مشاهده پایگاه داده ایجاد شده خوشحال شده است و نیازهای جدیدی را مطرح می‌سازد. مثلاً می‌خواهد اطلاعات رزرو ثبت‌نام را داشته باشد. برای این منظور ابتدا ساختار داده‌ای (جدول‌ها) تحلیل می‌شود تا کسری اطلاعات اضافه شود و سپس مراحل بعدی طراحی و از آن جمله شناسایی سایر اشیاء پایگاه داده صورت می‌گیرد.

۱-۱۴-۸- نیازهای جدید مطرح شده: والدین در تابستان و از قبل، ثبت نام کودک خود را به صورت رزرو انجام می‌دهند. مهدکودک مورد نظر دوست دارد بداند:

- چه تعدادی درخواست رزرو داشته‌اند؟
- چه زمانی کودک به مهدکودک آورده می‌شود؟
- برای سال جدید چه تعداد تخت و اتاق نگهداری لازم است؟
- هزینه نگهداری مثل استفاده از تخت و غذا چقدر است؟
- هزینه ایجاد گرما چقدر است؟ (در زمستان برای همه و در سایر فصل‌ها برای برخی خردسالان مورد نیاز است.)

- هزینه بیمه چقدر می‌شود؟
- نرخ‌های روزانه به صورت زیر است: (نرخ‌ها در زمان‌های مختلف تغییر می‌کنند)
- هزینه نگهداری: به ازای هر اتاق ۳۰۰۰+ به ازای هر کودک ۲۰۰۰
- هزینه تولید حرارت: به ازای هر اتاق ۵۰۰۰۰
- هزینه بیمه: به ازای هر کودک ۱۰۰۰۰

۲-۱۴-۸- تحلیل نیازهای جدید: قبل از هر چیز و قبل از ایجاد اشیاء جدید، باید به سؤالات زیر پاسخ دهیم:

- آیا می‌توانیم نیازهای جدید را با کمک اطلاعات موجود در پایگاه داده ارایه کنیم؟
- پاسخ: خیر، اطلاعات مورد نظر رزرو در هیچ جدولی (نه Parents و نه Children) وجود

ندارد.

- اگر نه، چه اطلاعات بیشتری لازم است تا در پایگاه داده ذخیره شود؟

پاسخ: باید جزئیات رزرو را ذخیره کرد مثل: تاریخ ثبت نام، زمان ورود کودک، تعداد کودکان، تعداد اتاق‌ها و نرخ‌های روزانه.

— آیا می‌توان با افزودن مشخصه‌ها (فیلدها)ی جدید به موجودیت‌های موجود (جدول‌ها) آن‌ها را ذخیره کرد؟

پاسخ: چون والدین رزرو را انجام می‌دهند، ممکن است وسوسه شوید و بخواهید فیلدهای جدید را به جدول والدین اضافه کنید. با توجه به مفاهیم اولیه و بحث نرمال‌سازی می‌دانید که این کار موجب آنومالی‌هایی می‌شود.

— اگر نیاز به افزودن جدول جدید است، چه موجودیت‌هایی باید در نظر گرفته شود؟

پاسخ: موجودیت رزرو (Booking) مطرح می‌شود.

— اگر تصمیم به افزودن جدول‌های جدید گرفته‌ایم، ارتباطات آن‌ها با جدول‌های دیگر چیست؟

پاسخ: ارتباط، بین والدین و رزرو وجود دارد ولی ارتباطی بین کودک و رزرو وجود ندارد. هریک از والدین ممکن است یک یا بیشتر رزرو داشته باشد. هر رزرو فقط به وسیله یک فرد صورت می‌گیرد. در اصطلاح پایگاه داده این نوع ارتباط از نوع یک — به — چند (One-to-Many) است.

— کلید اصلی هر یک از جدول‌های جدید چیست؟

پاسخ: شماره رزرو را به عنوان کلید اصلی ایجاد می‌کنیم و ثبت آن را در جدول به Access محول می‌کنیم. یعنی شماره رزرو از نوع AutoNumber باشد.

— چه کلیدهای خارجی باید به خصوصیات جدول جدید یا جدول‌های قبلی اضافه شود؟

پاسخ: ارتباط جدول جدید با والدین از نوع یک — به — چند است بنابراین کلید خارجی ParentsId را به جدول جدیدمان اضافه می‌کنیم.

تحلیل بالا نشان می‌دهد تا این‌جا، دو شیء جدید مورد نیاز است. جدول TableBookings که داده‌های رزرو را نگه می‌دارد. همچنین شیء فرم FormBookings که برای ورود اطلاعات رزرو مورد نیاز است.



همیشه برای تعیین نوع داده فیلدهای جدول دقت کنید، چرا که فضای حافظه مورد نیاز را تعیین می‌کند.

۱۵-۸- طراحی منطقی جدول رزرو Booking

اکنون که نیاز به جدول جدید را بررسی کردیم، باید فیلدهای آن را تعیین کنیم.

● این جدول نیز مانند سایر جدول‌ها، نیاز به شناسه منحصر به فردی دارد که آن را BookingId می‌نامیم و نوع آن را AutoNumber قرار می‌دهیم.

● همچنین باید مشخص کنیم چه کسی رزرو را درخواست کرده است. می‌دانیم ParentsId ابزار ایجاد این ارتباط است. همان‌طور که از آن به عنوان کلید خارجی در جدول کودکان استفاده کردیم، آن را به جدول رزرو نیز اضافه می‌کنیم و همان‌طور که قبلاً گفتیم نوع آن باید Long Integer باشد.

● همچنین باید بدانیم، کودک چه زمانی به مهدکودک آورده می‌شود. این فیلد را ChildInDate می‌نامیم و نوع آن را Date فرض می‌کنیم.

● برای ذخیره تعداد کودکان در درخواست، فیلد NoOfChildren را از نوع Byte در نظر می‌گیریم.

● برای ذخیره تعداد اتاق‌های مورد درخواست، فیلد NoOfRooms را از نوع Byte در نظر می‌گیریم.

● برای دانستن هزینه نگهداری به اطلاعات تعداد کودکان، تعداد اتاق‌های مورد نیاز، تعداد روزها و نرخ‌های روزانه نیاز داریم تا هزینه نگهداری محاسبه شود. بنابراین این فیلد محاسباتی است و نیازی به ذخیره آن در یک فیلد نداریم. با این وجود می‌دانیم که نرخ‌ها متغیر هستند بنابراین به جای ذخیره فیلد هزینه، دو فیلد دیگر را اضافه می‌کنیم: RoomRate و ChildRate به ترتیب نرخ هزینه اتاق و نرخ کودک. هر دو آن‌ها از نوع Currency باشد (چون نشان‌دهنده قیمت هستند). با عبارت زیر می‌توان هزینه نگهداری را حساب کرد.

$$([\text{RoomRate}] * [\text{NoOfRooms}] + [\text{ChildRate}] * [\text{NoOfChildren}])$$

● همچنین باید هزینه تولید حرارت را در نظر بگیریم. می‌توانیم HeatRate را از نوع داده Currency ذخیره کنیم و از عملگر IIF() برای محاسبه هزینه تولید حرارت در غیر زمستان که فقط برای خردسالان لازم است استفاده کنیم. این عملگر پیش‌ساخته، بسته به درستی یا نادرستی عبارت قرار گرفته در قسمت اول، قسمت دوم یا قسمت سوم را اجرا می‌کند.

Syntax Iif (expr ; truepart ; falsepart)

آرگومان‌های تابع

Expr: عبارتی است که باید محاسبه شود.

Truepart : مقدار یا عبارتی است که در صورت درست بودن عبارت `expr` برمی گردد.
Falsepart : مقدار یا عبارتی است که در صورت نادرست بودن عبارت `expr` برمی گردد.
 عبارت محاسبه هزینه تولید حرارت عبارت است از :

$\text{Iif}([\text{Heating}], [\text{HeatRate}] * [\text{NoOfRooms}], 0)$

● هزینه بیمه، اگر نرخ آن را بدانیم به آسانی قابل محاسبه است. با داشتن فیلد `InsRate` (نرخ بیمه) از نوع `Currency` می توان این هزینه را با فرمول زیر محاسبه کرد :

$[\text{InsRate}] * [\text{NoOfChildren}]$

قبل از ایجاد جدول، به طراحی منطقی `FormBookings` نگاهی می اندازیم تا ببینیم چه اشیایی در فرم موردنیاز است. با طراحی منطقی این فرم (که ورودی و خروجی جدول `TableBookings` را فراهم می آورد) می توانیم طراحی جدول را کنترل کنیم و از هرگونه خطای منطقی در ساختار جدول جلوگیری شود.

۱۶-۸- طراحی منطقی فرم رزرو Booking

شناسه رزرو :	نام والدین :
تعداد کودکان :	زمان ورود :
اتاق های مورد نیاز :	
نرخ شارژ روزانه اتاق :	نرخ شارژ روزانه کودک :
نرخ شارژ روزانه تولید حرارت :	نرخ بیمه :
هزینه نگهداری :	
هزینه تولید حرارت :	
هزینه بیمه :	

شکل ۱۹-۸- فرم تعریف رزرو

اشیاء زیر در فرم موردنیاز است :

BookingNo : شناسه منحصر به فرد جدول را نشان می دهد. تا در صورت نیاز به اطلاع دهی به وسیله این شماره به والدین اطلاع داده شود. از شیء `Text Box` برای آن استفاده می شود. منبع داده ای آن فیلد `BookingNo` از جدول `Bookings` است.

ChildInDate : زمان ورود کودک به مهدکودک را نشان می دهد. شیء `Text Box` برای آن مناسب است و با فیلد `ChildInDate` جدول `Bookings` در ارتباط است و بنابراین خصوصیت

Control Source آن، ChildInDate است.

NoOfRooms: تعداد اتاق‌های مورد نیاز برای نگهداری کودک است. شیء Text Box که خصوصیت فیلد متناظر آن NoOfRooms است، برای آن مناسب است.

Heating: برای ذخیره آن که حرارت مورد نیاز است یا خیر استفاده می‌شود. از Check Box برای آن استفاده می‌کنیم که خصوصیت فیلد متناظر آن Heating است.

RoomRate: نرخ شارژ روزانه اتاق است. Text Box ای قرار می‌دهیم که به RoomRate متصل می‌شود.

ChildRate: نرخ شارژ روزانه کودکان است. Text Box ای است که به ChildRate متصل است.

HeatRate: نرخ شارژ روزانه تولید حرارت است. Text Box ای قرار می‌دهیم که به HeatRate متصل است.

InsRate: نرخ شارژ روزانه بیمه است. Text Box ای است که به InsRate متصل است.

Text Box ای برای نمایش KennelCost که هزینه نگهداری را محاسبه و نشان دهد.

Text Box ای برای نمایش HeatingCost که هزینه تولید حرارت را محاسبه و نشان دهد.

Text Box ای برای نمایش InsuranceCost که هزینه بیمه را محاسبه و نشان دهد.

حال باید طراحی پیشنهاد شده را با نیازهای جدید تطابق دهیم. تا مشخص شود تمام اطلاعات مورد نیاز برای پاسخ‌گویی به نیازهای جدید فراهم آمده است.

۸-۱۷ - ایجاد جدول رزرو Booking

جدول ۸-۳

نام فیلد	نوع داده‌ای	اندازه فیلد
BookingNo	AutoNumber	
ParentsId	Number	Long Integer
ChildInDate	Date/Time	
NoOfChildren	Number	Byte
NoOfRooms	Number	Byte
Heating	Yes/No	
RoomRate	Currency	
ChildRate	Currency	
HeatRate	Currency	
InsRate	Currency	

روش ایجاد جدول را به طور کامل آموخته‌اید. پس جدول TableBooking را با توجه به اطلاعات بالا ایجاد کنید.

در نمای Datasheet داده‌های زیر را در جدول مطابق شکل ۲۰-۸ ذخیره کنید.

BookingNo	ParentId	ChildInDate	NoOfChildren	NoOfRooms	Heating	RoomRate	ChildRate	HeatRate	InsRate
2007-01-01	1	2007-01-01	1	1	Yes	100000	50000	100000	100000
2007-01-01	1	2007-01-01	1	1	No	100000	50000	100000	100000
2007-01-01	1	2007-01-01	1	1	No	100000	50000	100000	100000
2007-01-01	1	2007-01-01	1	1	No	100000	50000	100000	100000
2007-01-01	1	2007-01-01	1	1	No	100000	50000	100000	100000

شکل ۲۰-۸

فیلدهای از نوع Yes/No به صورت Check box دیده می‌شوند. اگر انتخاب شده باشد، Yes و در غیر این صورت No است. با کلید Space صفحه کلید می‌توانید با هر بار فشردن کلید، بین انتخاب شدن و یا عدم انتخاب جابه‌جا شوید. این حالت را به صورت زیر می‌توانید تغییر دهید. در Field Properties فیلد موردنظر را کلیک کنید. در زبانه Lookup خصوصیت Display Control موردنظر خود را انتخاب کنید.

۱۸-۸- ایجاد فرم رزرو FormBooking

برای ایجاد فرم از روش ویزارد استفاده می‌کنیم و پس از آن اشیاء دیگر را به فرم افزوده و یا تغییر می‌دهیم.

شکل ۲۱-۸

جابجایی کنترل : می‌توانید آن را انتخاب نموده و به محل مناسب جابه‌جا نمایید و یا اندازه آن را تغییر دهید.

برای انتخاب روی آن کلیک کنید. پس از انتخاب، مربع‌های تغییر اندازه و جابه‌جا کردن ظاهر می‌شوند (شکل ۸-۲۲).



شکل ۸-۲۲

برای انتخاب برچسب متناظر آن باید با Shift، روی برچسب مجدداً کلیک کرد. در این صورت برای آن نیز مربع‌های تغییر اندازه ظاهر می‌شوند (شکل ۸-۲۳).



شکل ۸-۲۳

برای انتخاب برچسب به تنهایی باید با Shift، روی برچسب کلیک کرد و مجدداً روی Combo box کلیک کرد. در این صورت مربع‌های تغییر اندازه فقط برای برچسب ظاهر می‌شود (شکل ۸-۲۴).



شکل ۸-۲۴

با کلیدهای جهت‌ی صفحه کلید و یا درگ خود شیء، همیشه این دو با هم جابه‌جا می‌شوند. در صورتی که بخواهید یکی را فقط جابه‌جا کنید روی مربع جابه‌جایی (مربع بزرگ چپ - بالا) کلیک و درگ می‌کنید. برای حذف برچسب متصل به Combo Box، آن را به تنهایی انتخاب و دکمه حذف را بزنید.



این تکنیک برای تغییر اندازه و جابه‌جایی تمام کنترل‌ها و اشیاء فرم به کار می‌رود. همچنین نحوه حذف برچسب متصل نیز به همین روش صورت می‌گیرد.

افزودن Text Box برای نمایش فیلدهای محاسباتی

– اطمینان یابید که فرم در نمای Design باز باشد. در جعبه ابزار (ToolBox) روی دکمه Text Box کلیک کنید.


– در فضای خالی فرم مربعی رسم کنید. کنترل به همراه برچسب ایجاد می‌شود (شکل ۲۵-۸).

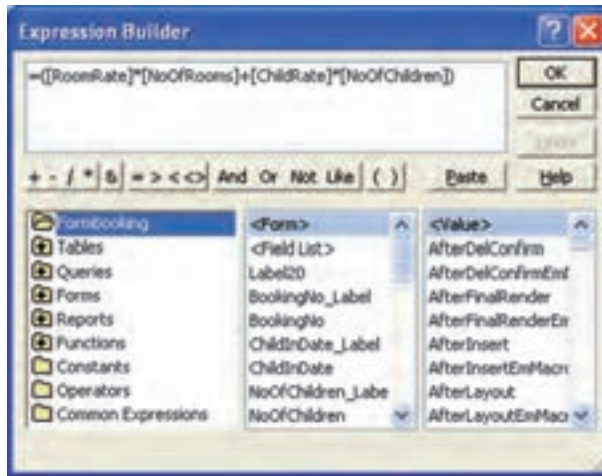


شکل ۲۵-۸

– روی برچسب کلیک کنید. در پنجره Properties، خصوصیات برچسب ظاهر می‌شود. خصوصیت Caption را LabelKenneCost بگذارید.
– در خارج پنجره کلیک کنید تا تغییر را روی فرم مشاهده کنید.
– روی برچسب کلیک کنید تا انتخاب شود. ماوس را در مرز آن قدر جابه‌جا کنید که شبیه فلش دوجبه‌ته شود. سپس مرز را درگ کنید تا جایی که کل متن برچسب مشاهده شود.
– Text Box را انتخاب کنید و در پنجره خصوصیات، خصوصیت Name را به FieldKenneCost تغییر دهید.

در ادامه برای تولید عبارت محاسباتی از Expression Builder استفاده می‌کنیم تا هزینه نگهداری را محاسبه کند. می‌توان عبارت را تایپ کرد ولی با کمک Builder احتمال خطا و اشتباه تایی کاهش می‌یابد.

– در پنجره Properties روی زبانه Data کلیک کنید. سپس در خصوصیت Control Source، روی  سمت راست کلیک کنید. پنجره Expression Builder نشان داده می‌شود (شکل ۲۶-۸).



شکل ۸-۲۶

– روی "=" و سپس "(" در قسمت میانی کلیک کنید. این دو در قسمت متنی بالا نمایش داده می‌شوند.

– از پنجره میانی قسمت سه بخشی پایین، RoomRate را دوبار کلیک نمایید (اگر آن را مشاهده نمی‌کنید در لیست به پایین جابه‌جا شوید). در قسمت متنی فیلد در بین [] قرار دارد.

– به این ترتیب عبارت زیر را با کلیک روی عملگرهای میانی و دوبار کلیک روی فیلد موردنظر ایجاد کنید.

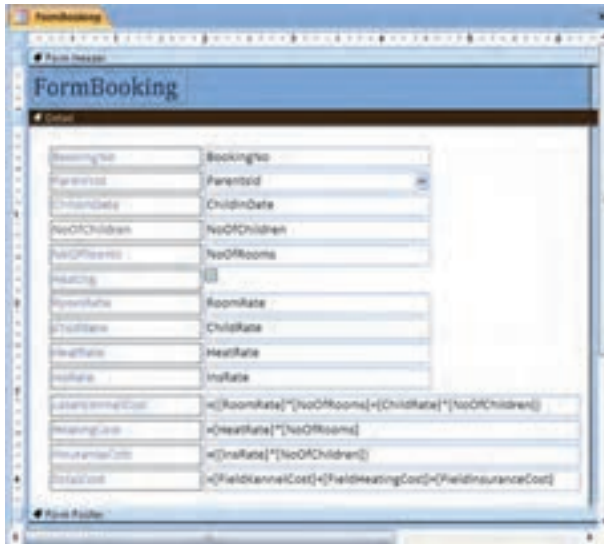
$$=[\text{RoomRate}] * [\text{NoOfRooms}] + [\text{ChildRate}] * [\text{NoOfChildren}]$$

– OK را کلیک کنید تا Expression Builder بسته شود.

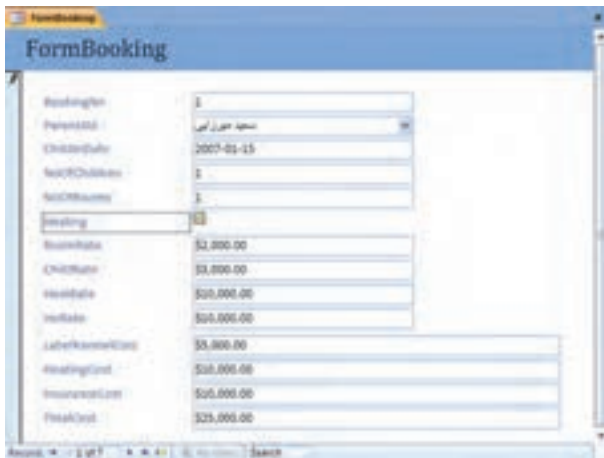
– خصوصیت Format را در پنجره خصوصیات، Currency قرار دهید.

کنترل‌های محاسباتی Insurance Cost، Heating Cost و Total Cost را نیز با افزودن Text Box و با کمک Expression Builder و عبارات تعریف شده در طراحی منطقی به فرم اضافه کنید (شکل ۸-۲۷).

به کمک این فرم اطلاعات رزرو را مشابه شکل ۸-۲۸ ایجاد کنید.



شکل ۲۷-۸



شکل ۲۸-۸

پروژه‌های نمونه

برای هر یک از محیط‌های عملیاتی پیشنهادی، موجودیت‌ها، فیلدها و ارتباط مناسب بین موجودیت‌ها را شناسایی کرده، بانک اطلاعاتی مربوطه را ایجاد نمایید. (فرم‌ها و گزارشات موردنیاز را ایجاد کنید.)

۱- بانک

۲- باشگاه ورزشی

۳- هتل

۴- دانشگاه

۵- کتابخانه

میانبرهای صفحه کلید در Access

Ctrl + O	باز کردن بانک اطلاعاتی موجود
Ctrl + N	ایجاد بانک اطلاعاتی جدید
Alt + F4	خروج
Ctrl + P	چاپ شیء انتخاب شده
Ctrl + S	ذخیره کردن
Alt + V + P	باز کردن صفحهٔ property برای شیء انتخاب شده
F5	مشاهدهٔ فرم در نمای طراحی
Ctrl + C	کپی شیء انتخاب شده به Clipboard
Ctrl + V	کپی شیء از Clipboard به محیط
Alt + SPACEBAR	مشاهدهٔ منوی کنترل
Ctrl + F4	بستن پنجرهٔ فعال
Alt + TAB	رفتن به پنجرهٔ بعدی در ویزارد
F7	بررسی spelling
F2	تغییر نام شیء انتخاب شده
Shift + F10	نمایش منو به صورت میانبر
F10	فعال کردن نوار منو
F1	راهنمای سیستم Help
Ctrl + Z	لغو عمل

Ctrl + P	باز کردن صفحه تنظیمات چاپ و چاپ
S	باز کردن صفحه Page setup
Z	بزرگنمایی Zoom
Alt + ENTER	نمایش صفحه مشخصات
CTRL + ENTER	ایجاد خط جدید در فیلد از نوع memo یا text
CTRL + '	اضافه کردن فیلد مشابه در رکورد قبلی به رکورد جاری
ESC	برگرداندن تغییرات اعمال شده روی فیلد جاری
ESC ESC (press ESC twice)	برگرداندن تغییرات اعمال شده روی رکورد جاری
F11	نمایش پنجره بانک داده
Ctrl + F	جستجو کردن و جایگزین کردن



واژه نامه تشریحی

Anomaly	آنومالی
	انجام ناپذیری یکی از عملیات در بانک، بروز تبعات نامطلوب و یا فزون کاری برای انجام یکی از عملیات در بانک را گویند.
Cardinality	ماهیت ارتباط
	تناظر بین موجودیت‌ها است که به سه نوع ۱: ۱، ۱: n و n: m وجود دارد.
Data Model	مدل داده‌ای
	مدلی است مفهومی که برای نمایش داده‌ها و ارتباط بین آن‌ها استفاده می‌شود. انواع رایج آن عبارت است از: رابطه‌ای، سلسله مراتبی و شبکه‌ای.
Database	پایگاه داده/ بانک اطلاعاتی/ بانک داده
	مجموعه‌ای از داده‌ها که به صورت یکپارچه با حداقل افزونگی تحت کنترل یک سیستم متمرکز و در چارچوب یک مدل داده‌ای مدیریت می‌شوند.
Database Management system (DBMS)	سیستم مدیریت پایگاه داده
	سیستمی که مدیریت بانک اطلاعاتی را برعهده دارد و هرگونه دستیابی به داده‌ها از طریق آن انجام می‌شود.
Entity	موجودیت
	هر مفهوم یا شیء در محیط عملیاتی است.
Entity-Relationship Diagram (ER)	نمودار ارتباط — موجودیت

نمودار نشان‌دهنده موجودیت‌ها، ویژگی‌های هر موجودیت، ارتباط بین موجودیت‌ها و بیان عملکرد و ماهیت ارتباط است. در این نمودار از شکل‌های استاندارد استفاده می‌شود.

فیلد
Field
یک قطعه داده است، همان واحد معنایی داده. نام فیلد به همراه مقدار آن در حکم یک اطلاع است.

سیستم فایلینگ
Filing system
نرم‌افزار کاربردی است که خود ذخیره و بازیابی داده‌ها را نیز برعهده دارد.

کلید خارجی
Foreign key
کلید اصلی یک جدول است که بنا بر ماهیت ارتباط، به جدول دیگر اضافه می‌شود و در آن جدول با نام کلید خارجی شناخته می‌شود و باعث ارتباط دو جدول می‌شود.

فرم
Form
نمای گرافیکی برای نمایش داده است که امکانات اضافه، ویرایش، حذف و هرگونه عملیات روی داده‌ها را فراهم می‌سازد.

کلید
key
فیلد یا مجموعه‌ای از فیلدها که باعث یکتایی هر یک از رکوردها شود.

فرم نرمال
Normal form
تجزیه‌ای از جدول است که دارای آنومالی نباشد. دارای سطوح نرمال 1NF، 2NF، 3NF، 4NF، BCNF و 5NF است.

محیط عملیاتی
Operational Environment
محیطی است که می‌خواهیم یک سیستم ذخیره و بازیابی برای آن ایجاد کنیم.

کلید اصلی
Primary key
یکی از کلیدهاست که با توجه به ملاحظات به عنوان کلید اصلی جدول در نظر گرفته می‌شود. ممکن است از ویژگی‌های آن موجودیت نباشد و به صورت مجازی برای یکتا شدن رکوردهای جدول به آن اضافه شود.

پرس و جو
Query
ابزار مناسب استخراج اطلاعات از جدول‌ها است. برای ایجاد آن از قواعد یک دستور

زبان استاندارد به نام SQL استفاده می‌شود. اگر چه برخی سیستم‌های مدیریت پایگاه داده، محیطی گرافیکی برای آن ایجاد کرده‌اند ولی سیستم دستوره‌های متناظر را برای آن ایجاد می‌کند.

Record

رکورد

مجموعهٔ فیلدهای مرتبط را گویند.

Relation

رابطه

زیر مجموعه‌ای از حاصل ضرب دکارتی مجموعه‌ها است. مفاهیم مرتبط با نظریه مجموعه‌ها به ترتیب عبارت است از: عضو، مجموعه، حاصل ضرب دکارتی، رابطه و تابع.

Relational Database

بانک دادهٔ رابطه‌ای

سیستم مدیریت بانک داده است که از مدل رابطه‌ای که بر مفهوم ریاضی مجموعه‌ها بنا شده است، پشتیبانی کند.

Relationship

ارتباط

ارتباط بین موجودیت‌ها است. هر ارتباط دارای مفهوم و عملکرد است.

Report

گزارش

یک فرم بیان اطلاعات است در قالبی مشخص و گویا که بر روی صفحه نمایش یا چاپگر نمایش داده می‌شود.

Table

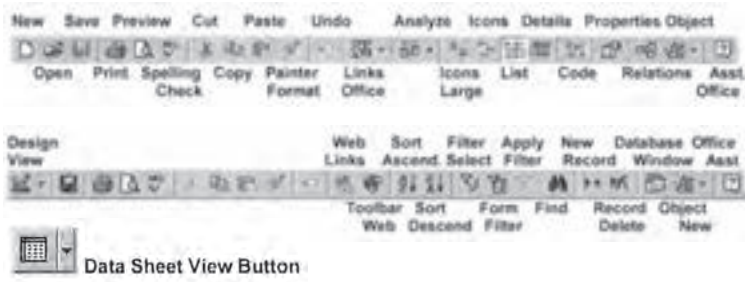
جدول

جدول محلی است که داده‌ها ذخیره می‌شوند و نشان‌دهندهٔ یک موجودیت یا ارتباط بین موجودیت‌ها است.

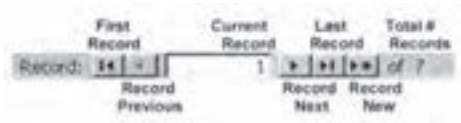
General Concepts

1. Microsoft Access is a database management program. It is intended to manage sets of data. You can enter, edit, and delete data and print reports using this program.
2. The view in which data is entered and edited is called a **table**.
3. The smallest piece of information in a database (e.g. last name) is called a field.
4. All of the pieces of information (all of the fields) about a unit in the database is called a record.
5. In Access, a database consists of all the tables, queries, forms, and reports involved with one set of data.
6. In Access, the Tables View is used to establish the fields and the data types. It is also the view in which data are entered and edited.
7. In Access, the Reports view is used to define printed reports.
8. In Access, the Queries View is used to select records to be viewed or printed.
9. In Access, the Forms View is used to define special data entry forms.
10. In most areas of Access, there are two basic views available, Datasheet View and Design View. In Datasheet View, you see the records in column and row format and can enter, edit, or delete records. Design View allows you to modify the particular item in view, e.g. the table, query, report, or form.

Microsoft Access Buttons and Icons



Records Bar in Table View



1. How do I open a database that already exists?

If Access is not open:

1. Start Access
2. Click the **Open an Existing Database** radio button and press **ENTER**
3. Locate and highlight the existing database filename
4. Click the **OPEN** button

■ OR --

If Access is already open:

1. Click **File, Open**
2. Locate and highlight the existing database filename
3. Click the **OPEN** button

2. How do I close a database?

1. Close all tables, forms, queries or reports that are open (click the **X** box)
2. Close the main database window (click the **X** box)

3. How do I enter data into a database?

1. Open a database
2. If no specialized entry Form exists, open a Table
 - A. Click the **Table** tab

- B. Click a Database filename
 - C. Click the **Open button**
3. The cursor will probably automatically move to the new record. Simply type the new data and press Enter to move to each new field.
 4. Pressing Enter at the last field adds the record and positions the cursor for the next record to be added.

4. How do I enter a new record if I am already in the editing mode?

Press the **New Record button**

The cursor will automatically be taken to a new record

5. How do I edit a record if I am browsing the data?

1. Go to **Datasheet View**
2. Move the cursor to the record to be edited with the **Arrow keys**
3. Move to the field to be edited using the **Enter, Tab, or Arrow keys**
4. If the entire field is highlighted:
 - A. Retype the data
--OR--
 - B. Press the **F2 key** and edit the field using normal editing keys
--OR--
 - C. Click in the desired field and edit using normal editing keys

6. How do I delete a record?

1. Click the record selection button at the front of the record (the entire record is selected).
2. Click the Record Delete button
--OR--
Press the **DELETE key** on the keyboard
3. Answer **Y** to the confirmation question.

7. How do I repeat data from the previous record?

Press **Ctrl-'** when you are in the correct field

8. How do I create a new database?

1. Click **File, New**
2. Click the **General tab**

3. Click the **Blank Database icon** and click **OK**
4. Click Set a **name, and location**
5. Click the **Create Database button**

9. How do I create a new table in a database?

Be sure to have designed your database first so you will know what fields are needed and their type.

1. Open the database
2. Click the **Tables tab**
3. Click the **New button**
4. Click **Design View button**, Click **OK**
5. Type the Field Names, set the Data Types, and set Properties

(**Note:** Field Names can be up to 64 characters long. Do Not use periods, exclamation points, accent graves ('), square brackets ([]) or leading spaces. Keep names short and do not uses spaces unless absolutely necessary.)

6. Click the **Datasheet View button** to preview how the table will look and practice data entry

--OR--

Be sure to have designed your database first so you will know what fields are needed and their type.

1. Open the database
2. Click the **Tables tab**
3. Click the **New button**
4. Click the **Table Wizard option** and click **OK**
5. Select the fields to be in the new table
 - A. Double-click a field in the sample fields column to put in the new table column
 - B. If you can't find a field like you want, choose a different sample table in the left column and then choose a sample field
 - C. Select the new fields in the order you want them in the new database
 - D. If you get the wrong field or get the fields in the wrong order, double-click the names in the new fields column to put them back in the sample fields column
6. Rename any fields necessary
 - A. Click the field name in the New Fields column

- B. Click the **Rename Field button**
 - C. Type the new name for the field and press **Enter**
 7. When all fields have been entered and renamed, click the **Next button**
 8. Name the new table, click the **Yes, set the Primary Key for me button**, and click the **Next button**
 9. Click the **Enter data directly into the table radio button** and click the **Finish button**
 10. Click the **Design view button**
 11. Click the leftmost column of the field that contains the key symbol
 12. Click the Primary key icon on the toolbar
 13. Click the **Save icon**
 14. Click the **Datasheet view icon**
 15. Enter data into your new table
- 10. If I am in a table and want to go to a report or query, how do I get to the main database window to select a report or query?**
Press the **Database Window button**

11. How do I set the criteria in a query to tell it which records I want to retrieve?

See the attached handouts for entering queries

12. What does the Input Mask field mean in creating a database table?

The Input Mask is used to limit or control the way information is entered. For example, you can tell Access to convert all characters to uppercase as in the State field in a database. You could also limit a Zip Code field to only numbers. See the associated handout for the characters to use in the Input Mask area and examples of how to use it.

13. How do I create a report?

1. Click the **Reports tab**
2. Click the **New button**
3. Click the **Report Wizard option**, select the table or query for the report using the drop down button, and click the **OK button**
4. Double-click each field to be listed in the new report in the order you want them to appear and click the **Next report**
5. Set grouping levels and click the **Next button**

6. Set the sort fields and sort order and click the **Next button**
7. Select the **Layout** and **Orientation** options and click the **Next button**
8. Select the **Style** and click the **Next button**
9. Give the report a name, click the **Preview the report radio button**, and click the **Finish button**
10. Close the report

14. How do I create a Labels report?

1. Click the **Reports tab**
2. Click the **New button**
3. Click the **Label Wizard option**, select the table or query for the report using the drop down button, and click the **OK button**
4. Set the Label type (Sheet fed or Continuous); select the label size (e.g. Avery 5160); and click the **Next button**
5. Set the Font name, size, weight, color and italic/underline options and click the **Next button**
6. Construct the mailing label and click the **Next button**
7. Set the sort fields in the correct order and click the **Next button**
8. Name the report; click the **See the labels as they will look printed radio button**, and click the **Finish button**
9. Close the report

15. How do I create a query?

1. Click the **Query tab**
2. Click the **New button**
3. Click the **Simple Query Wizard option** and click the **OK button**
4. Select the fields you want in your query by double-clicking each field and click the **Next button**
5. Choose either a Detail or Summary report and click the **Next button**
(**Note:** Detail prints each record in the query. A Summary report allows Sum, Average, Max , Min, and Count options.)
6. Name the query and click the **Open the query to view information radio button** and click the **Finish button**
7. Close the report

16. If I have a report based on a query, how do I get it to update it so I can print another report?

1. If the report is open, **close it**
2. Open the associated query
3. Go to **Design View**
4. Enter the appropriate **query criteria**
5. Go to the **DataSheet View** (this actually retrieves the data)
6. **Save the Query** (you **MUST** save the query before the report will retrieve the updated data)
7. Open the **DataBase Window**
8. **Click the Reports tab**
9. Open the **associated report** (this will retrieve the records from the query)
10. **Print the Report**
11. Close **the Report (if you do not close the report and open it again after the next query, the report information will not be updated)**

منابع

- 1- Date C.J., An Introduction to Database Systems. 7th ed. Addison-Wesley, 2000.
- 2- Codd. E.F., The Relational Model for Database Management. Version 2. Addison-Wesley. 1990.
- 3- Viescas John L., Microsoft Office Access 2003 Inside Out. Microsoft Press. 2004.
- 4 - Office. microsoft. com, “Access 2003 home page”.
- 5 - Microsoft Office Access Help.
- 6 - Sheldon Robert, Microsoft Office Access 2003 A Beginner’s Guide. Edition 1. McGraw-Hill, 2003.
- ۷- روحانی رانکوهی، سید محمد تقی، مفاهیم بنیادی پایگاه داده‌ها، انتشارات جلوه، چاپ اول، پاییز ۱۳۸۰.

