
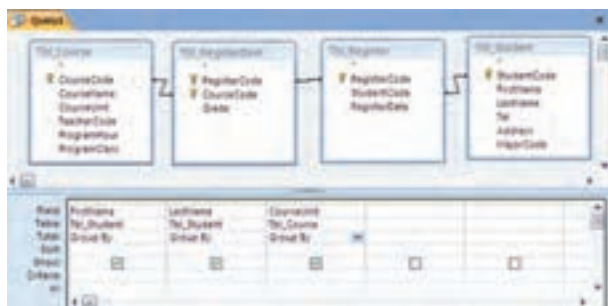



۱۰-۵- فیلدهای محاسباتی تجمعی (توابع Aggregate)

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 جوی 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	مهرزایی	6
سید کاظم	مهرزایی	11
سیدمهران	مهرزایی	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، دکمه 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.GradeByUnit)/ 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

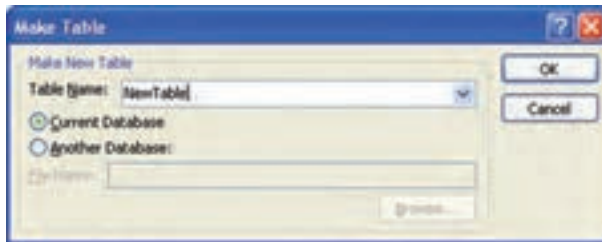
Record: 14 of 4 No Filter

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.

۱-۱۱-۵-۱ 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.

کنجکاو

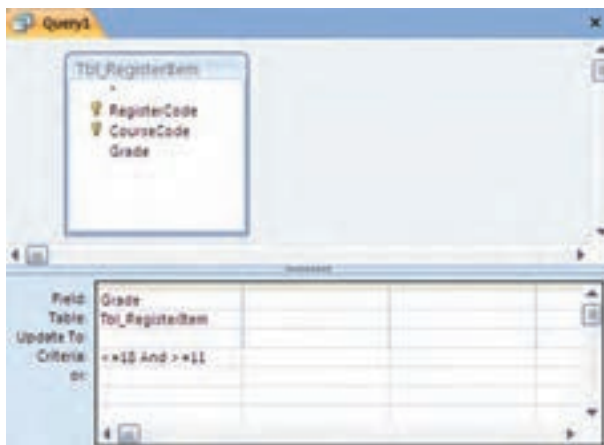
سایر پرس‌وجوهای عملیاتی که می‌توان در Access ایجاد کرد را نام برده، توضیح دهید هر یک برای انجام چه کاری استفاده می‌شوند؟

✓ 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 Tbl_RegisterItem 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 و جو عملیاتی، پیغام تأیید نمایش داده نمی‌شود؟

تمرین ۳-۵ :



پرس و جویی ایجاد کنید که در جدول حاوی معدل، دانش‌آموزانی که معدل آن‌ها زیر ۱۰ است را حذف کند.

راهنمایی: پرس و جویی ایجاد کنید که از جدول Tbl-Average دانش‌آموزانی که معدل زیر ۱۰ دارند را برگرداند، سپس از قسمت QueryType زبانه Design، گزینه Delete Query را انتخاب کرده و پرس و جو را ذخیره و اجرا نمایید.

۲-۱-۵- ایجاد پرس و جوی CrossTab

با این نوع پرس و جو می‌توانیم اطلاعات یک یا چند جدول را در قالب ردیف و ستون خلاصه نماییم. مثلاً محاسبهٔ sum، average، count، یا دیگر انواع مجموع داده که با دو نوع اطلاع (سطری و ستونی) گروه‌بندی می‌شود.

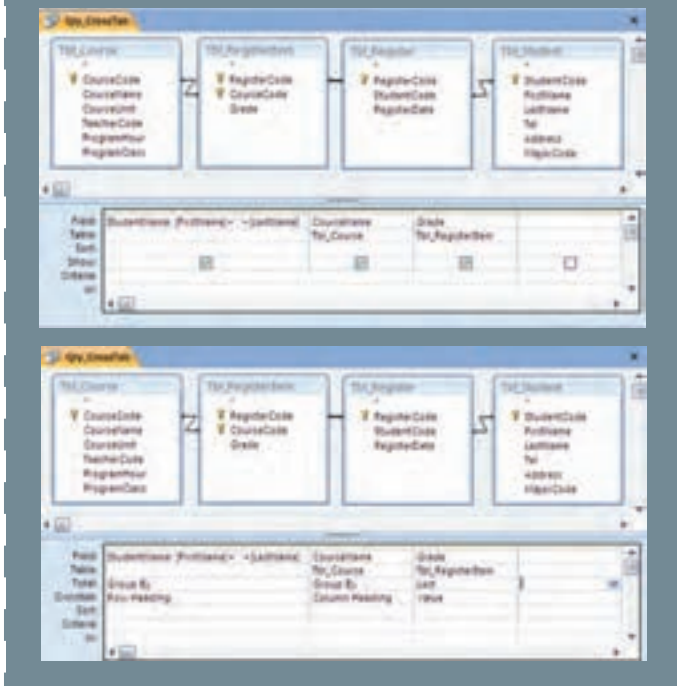
مثلاً یک پرس و جوی ساده، فقط معدل درسی شاگردان کلاس را برمی‌گرداند. در صورتی که اطلاعات زیاد باشد، مقایسه سخت می‌شود.

یک پرس و جوی CrossTab همان اطلاع را نشان می‌دهد. با این تفاوت که اطلاعات سطری با شاگردان و ستونی با درس گروه‌بندی می‌شود و این کار تحلیل را آسان‌تر می‌کند. بنابراین یک نوع پرس و جوی انتخاب رکوردها (Select) است، با این تفاوت که اطلاعات را خلاصه‌تر ارائه می‌کند. در این پرس و جو، داده‌های یک فیلد خاص را به عنوان ستون‌های جدول خروجی و فیلد دیگری را به عنوان ردیف‌های جدول خروجی تعیین می‌کنیم و مثلاً جمع کل مقادیر سطر بر حسب ستون را به عنوان مقدار محل برخورد سطر و ستون (سلول) تعیین می‌کنیم.

برای ایجاد این نوع پرس و جو، 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 برمی‌گزینیم. یکی از توابع را به عنوان داده‌های محل تقاطع سطرها و ستون‌ها انتخاب می‌کنیم.

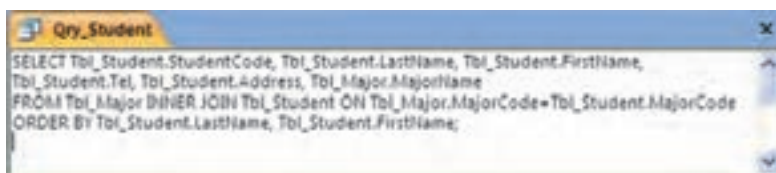
تمرین ۴-۵: ✓

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.



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
```

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
```

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 دوبه‌دو 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

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 ۱۸-۵

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



نکته

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 پرس‌وجوی 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 پرس‌وجوها برای دریافت اطلاعات از بانک استفاده می‌شود. پرس‌وجوها انواع مختلفی دارد که معمول‌ترین آن پرس‌وجوی Select است. به هر پرس‌وجو می‌توان محدودیت و شرط اضافه کرد. می‌توان فیلدهای غیرجدولی از نوع محاسباتی به پرس‌وجو اضافه کرد. با اجرای پرس‌وجو 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.