

\* اگر از DISTINCT استفاده نکنیم، چون هر درس به وسیله‌ی چندین دانش‌آموز ثبت نام 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 , TeacherLName + ' ' + TeacherFName FROM Tb1_Teacher
```

TeacherCode	Expr1
1030	سیدی نژاد حاکم‌زادگان
1036	مهدی فر محمد
1037	میرزاپور علیرضا
1038	هدایتی فر کاتبین
1039	حسینی امیرعلی
1050	سیورطینت محمد
1050	هدایت نژاد محمد
1051	سماقی پور علیرضا

این پرس و جو را اجرا کنید. نتیجه به صورت روبه‌رو خواهد بود:

حال نام مستعار به فیلد منتسب می‌کنیم، عبارت AS TeacherName:

```
SELECT TeacherCode, TeacherLName + ' ' + TeacherFName AS TeacherName FROM Tb1_Teacher
```

TeacherCode	TeacherName
1030	سیدی نژاد حاکم‌زادگان
1036	مهدی فر محمد
1037	میرزاپور علیرضا
1038	هدایتی فر کاتبین
1039	حسینی امیرعلی
1050	سیورطینت محمد
1050	هدایت نژاد محمد
1051	سماقی پور علیرضا

نتیجه‌ی جدید به صورت روبه‌رو خواهد بود:

### کنجکاو

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 بازیابی کند باشد و مشاهده‌ی نتیجه‌ی پرس و جو طول بکشد.

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

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

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

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 ی INTO را به همراه نام جدول جدید قرار می دهیم. در صورتی که بخواهیم جدول جدید در پایگاه داده ی دیگری قرار گیرد، آدرس آن را در IN قید می کنیم.

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

#### مثال ۵-۱۱

نسخه ی پشتیبانی از جدول معلمان ایجاد کنید :

پرس و جوی 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
```

#### مثال ۵-۱۲

رکوردهای جدول معلمان را حذف کنید.

پرس و جوی 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 ی مقادیر ثابت داده (Single append) و اضافه کردن رکوردهای

از پرس وجوی نوع Select (Multiple append).

• **Multiple-record append query:**

```
INSERT INTO target-TableName [(field1[, field2[,...]])] [IN externaldatabase]
SELECT [source-TableName.]field1[, field2[,...]]
FROM Source_TableName
```

---

---

**مثال ۵-۱۳**

اطلاعات جدول معلمان را از نسخه‌ی پشتیبان برگردانید.

پرس وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
INSERT INTO Tb1_Teacher (TeacherCode, Teacher FName, Teacher
LName, TeacherTe1, TeacherAddress)
SELECT TeacherCode, TeacherFName, TeacherLName, TeacherTel,
TeacherAddress
FROM Tb1_TeacherBackUp
```

---

---

• **Single-record append query:**

```
INSERT INTO target-TableName [(field1[, field2[,...]])]
VALUES (value1[,value2[,...]])
```

---

---

**مثال ۵-۱۴**

اطلاعات معلم جدید را اضافه کنید.

پرس وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
INSERT INTO Tb1_Teacher (TeacherCode, TeacherFName,
TeacherLName, TeacherTe1, TeacherAddress)
VALUES (۱۰۷۰, 'خیابان وزرا کوجه پنجم پلاک ۵' و '۸۸۹۷۱۰۳۴' و 'یعقوبی' و 'صدف' و
```

---

---

**۵-۱۳-۵ دستور UPDATE: ویرایش رکوردها**

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

---

---

### مثال ۵-۱۵

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

```
Update Tb1_Teacher SET TeacherFName='صدف'  
WHERE TecaherCode=1070
```

---

---

### ۵-۱۳-۶ عملگر UNION: برای ایجاد پرس وجوی ترکیب رکوردهاست.

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

در این دستور باید تعداد فیلدهای دو Select که با هم Union می شوند، 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.

---

---

### مثال ۵-۱۶

لیست اسامی معلمان و دانش آموزان مدرسه را به دست آورید.  
پرس وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT StudentCode AS PersonCode, Student FName AS  
PersonFName, StudentLName AS PersonLName  
FROM Tb1_Stuedent  
UNION  
SELECT TeacherCode AS PersonCode, TeacherFName AS  
PersonFName, TeacherLName AS PersonLName  
FROM Tb1_Teacher
```

---

---

### ۵-۱۳-۷ دستور JOIN: دیدیم یکی از روش های انتخاب اطلاعات از دو یا چند

جدول، آن است که در FROM نام جدول ها را با علامت "،" از هم جدا کرده و در WHERE ارتباط آنها را به صورت شرایط دوه دو بیان کنیم. یکی دیگر از راه ها که زمان ایجاد پرس وجو با کمک گرافیک نیز دیده ایم، اتصال جدول ها 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 نتیجه‌ی پرس وجو نمایش داده نخواهند شد.

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

---

### مثال ۵-۱۷

---

اطلاعات معلمان هر درس را به دست آورید.

پرس وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT CourseName, TeacherFName, TeacherLName  
FROM Tb1_Teacher INNER JOIN Tb1_Course  
ON Tb1_Teacher. TeacherCode = Tb1_Course. TeacherCode
```

---

**Right/Left Outer Join**: اطلاعات متناظر دو جدول در نتیجه‌ی پرس وجو 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 انتخاب شده 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 ۵-۱۸

---

اطلاعات معلمان هر درس را به دست آورید. در ضمن معلمان که در این ترم

درسی ارایه نداده‌اند نیز آورده شود.

پرس وجوی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT TeacherFName, TeacherLName, CourseName  
FROM Tb1_Teacher LEFT JOIN Tb1_Course  
ON Tb1_Teacher. TeacherCode = Tb1_Course. TeacherCode
```

---

## مثال ۵-۱۹

دروس ثبت نام شده‌ی دانش‌آموزان را به دست آورید. در ضمن دانش‌آموزانی که در این ترم هنوز درسی ثبت نام نکرده‌اند نیز آورده شود.

```
SELECT StudentFName, StudentLName, 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
```

## کنجکاو

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

## ۸-۱۳-۵ اعمال سایر شرایط به پرس و جوها

● 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 ۴-۷ توضیح دادیم است و از همان قواعد ذکر شده پیروی می‌کند.

## مثال ۵-۲۰

معلمی با نام 'مهدی فر' را حذف کنید.  
پرس و جوی 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 پرس و جو تمامی آن‌ها حذف می‌شوند.

```
DELETE
FROM Tb1_Teacher
WHERE TeacherLName = '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 ذکرشده در عبارت خواهد بود. دو واژه‌ی 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.

## مثال ۵-۲۱

لیست اسامی معلمان براساس نام خانوادگی و در صورت تساوی نام خانوادگی براساس نام مرتب‌شده باشد به دست آورید.  
پرس و جو ی زیر را ایجاد، ذخیره و اجرا نمایید.

```
SELECT Tb1_Teacher. TeacherFName,
Tb1_Teacher. TeacherLName
FROM Tb1_Teacher
ORDER BY Tb1_Teacher. TeacherLName ASC,
Tb1_Teacher. TeacherFName ASC
```

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

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

**تمرین ۵-۵:** SQL پرس و جوی محاسبه‌ی معدل Qry\_Average را مشاهده کنید.

```
SELECT Qry_Register. StudentName,  
Sum (Qry_Register. GradeByUnit)/  
Sum (Qry_Register. CourseUnit) AS GradeAverage  
FROM Qry_Register  
GROUP BY Qry_Register. StudentName
```

این پرس و جو یعنی نام دانش‌آموز را در نظر می‌گیرد و مجموع حاصل ضرب نمره در واحد درسی را به دست آورده، برای محاسبه‌ی معدل تقسیم انجام می‌دهد.

**نکته:** تمام فیلدهای عبارت Select غیر از آن‌هایی که در توابع تجمعی Min، Sum، Count استفاده شده‌اند، باید در لیست قسمت Group By آورده شده باشند.

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

زمانی که از توابع تجمعی Aggregate و Group by استفاده شود، برای اعمال شرط پس از اجرای پرس و جو، باید از معیار در HAVING استفاده کنیم. اگر پرس و جو دارای عبارت شرط Where باشد، رکوردها ابتدا به وسیله‌ی آن شرط محدود می‌شوند، سپس نتیجه‌ی 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 پرس و جو را محدود کنیم.

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



می‌خواهیم معدل دانش‌آموزانی را به دست 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
```

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

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

### برای مطالعه‌ی بیشتر

● **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، کلیدواژه‌ی 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
WHERW Tb1_RegisterItem. Grade > =10);
```

● **Data Definition**: همان‌طور 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، به‌وسیله‌ی دستورات زبان نیز قابل انجام است. مثل تغییر نام فیلد، تغییر نوع داده‌ی یک فیلد، افزودن فیلد جدید، افزودن کلید اصلی به جدول و ... .  
به‌طور مثال با اجرای دستور زیر که در SQL View یک پرس‌وجوی جدید تایپ می‌نمایید، 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),
[Birthdate] date,
[Phone] text (25),
[Notes] memo,
CONSTRAINT [Index1] PRIMARY KEY ([FriendID]))
```

## خلاصه ی فصل

برای تصمیم‌گیری نیاز به بازیافت اطلاعات ثبت‌شده 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 کنیم. با اجرای پرس‌وجو نتیجه‌ی آن به‌صورت جدولی نشان داده می‌شود که شبیه نمای Datasheet جدول است و همان عملیات را می‌توان روی آن اعمال کرد مثل: مرتب‌سازی، فیلترکردن و ... البته نتیجه‌ی پرس‌وجو در فرم یا گزارش‌ها نیز استفاده می‌شود. نوع دیگر پرس‌وجو، پرس‌وجوهای عملیاتی هستند که از آن‌ها برای تغییر روی داده‌ها استفاده می‌شود و به آن زبان دستکاری داده DML می‌گوییم. البته پرس‌وجوی عملیاتی برای تغییر ساختار بانک نیز وجود دارد که به آن زبان تعریف داده DDL می‌گوییم. از انواع DMLها، می‌توانیم به پرس‌وجوی جدول‌ساز، به هنگام‌سازی، حذف اشاره کنیم که به‌وسیله‌ی محیط گرافیکی Access نیز می‌توانند ایجاد شوند. در پشت‌ظاهر گرافیکی هر یک از این پرس‌وجوها، زبان ساخت‌یافته SQL قرار دارد. یعنی معادل هر یک از این امکانات یک دستور زبانی قرار دارد که با آن‌ها آشنا می‌شویم. پرس‌وجوهای پیچیده‌تر و پیشرفته‌تر به‌سختی به‌وسیله‌ی محیط گرافیکی قابل ایجاد هستند و باید به‌وسیله‌ی برنامه‌نویس و با قواعد زبان SQL نوشته شوند.

- ۱- پرس و جوها چه کاری انجام می دهند و چه زمانی قابل استفاده هستند؟
- ۲- پنجره ی طراحی پرس و جو را به اختصار توضیح دهید.
- ۳- معیارهای پرس و جو به چه منظور استفاده می شوند؟
- ۴- فیلدهای محاسباتی را توضیح دهید. با چه ابزار گرافیکی قابل ایجاد هستند؟
- ۵- پرس و جوهای عملیاتی را شرح دهید.
- ۶- زبان پرس و جوی ساخت یافته SQL چیست؟
- ۷- برخی دستورات SQL را نام ببرید.

### ایجاد و کار با گزارش‌ها

✓ در این فصل ایجاد گزارش و عملیات روی آن را بررسی می‌کنیم. گزارش‌ها یکی از راه‌های مفید نمایش و ارایه 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: در پایان این فصل انتظار می‌رود که هنرجو بتواند:

- تفاوت گزارش با سایر اشیاء را تشخیص دهد.
- نماهای مشاهده‌ی گزارش Preview، ایجاد گزارش و تغییر طراحی را بشناسد.
- گزارش‌ها را به وسیله‌ی AutoReport 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.
  - برحسب‌های 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 ارایه دهد.

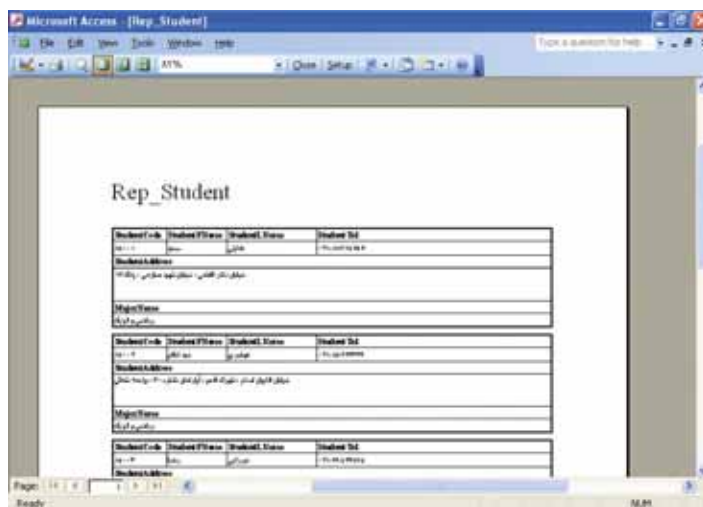
یک گزارش، رکوردها را شبیه نمای Datasheet فرم‌ها و جدول‌ها نشان می‌دهد. تفاوت نمای View در گزارش با Datasheet آن است که در گزارش خلاصه‌ی آماری، عناوین، زیرنویس، شماره‌ی صفحه، قالب نمایش داده زیباتری فراهم می‌شود. همچنین می‌توان داده‌های قابل نمایش را انتخاب کرد، داده‌ها را گروه‌بندی کرد و جمع کل روی آن 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 Print، با مرور داده‌های واقعی در هر صفحه‌ی گزارش می‌توانید گزارش را تایید کنید.
- نمای Layout Preview، با داده‌های نمونه فقط قالب نمایش گزارش را سریع مشاهده کنید.

**نکته:** 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، صورت حساب و ... باشد.

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



شکل ۱-۶- پنجره‌ی 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.

بهرتر 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 به وسیله‌ی AutoReports

با این امکان در بانک اطلاعاتی Access، می‌توان در نمای Datasheet، گزارش‌ها را به‌سرعت ایجاد می‌کند. 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.

### مثال ۱-۶

می‌خواهیم گزارش سریعی از معلمان مدرسه ایجاد کنیم:

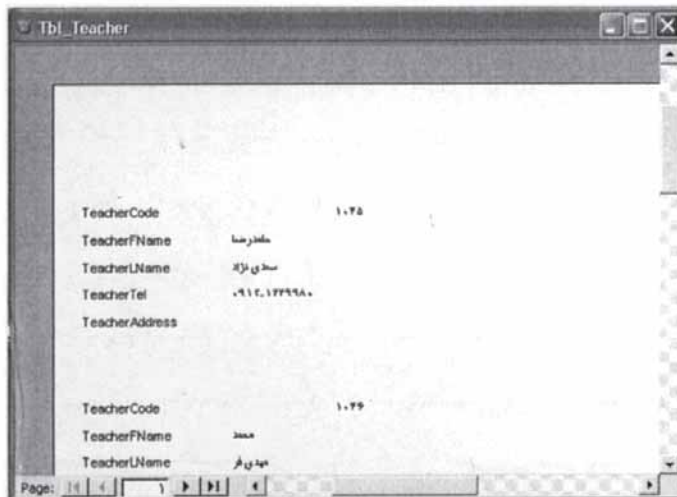
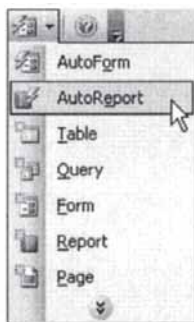
جدول یا پرس‌وجوی مورد نظر را انتخاب نمایید، مثلاً جدول Tbl\_Teacher.

آن را در نمای Datasheet باز کنید.

در صورت نیاز برای محدود کردن رکوردها، Filter اعمال نمایید.

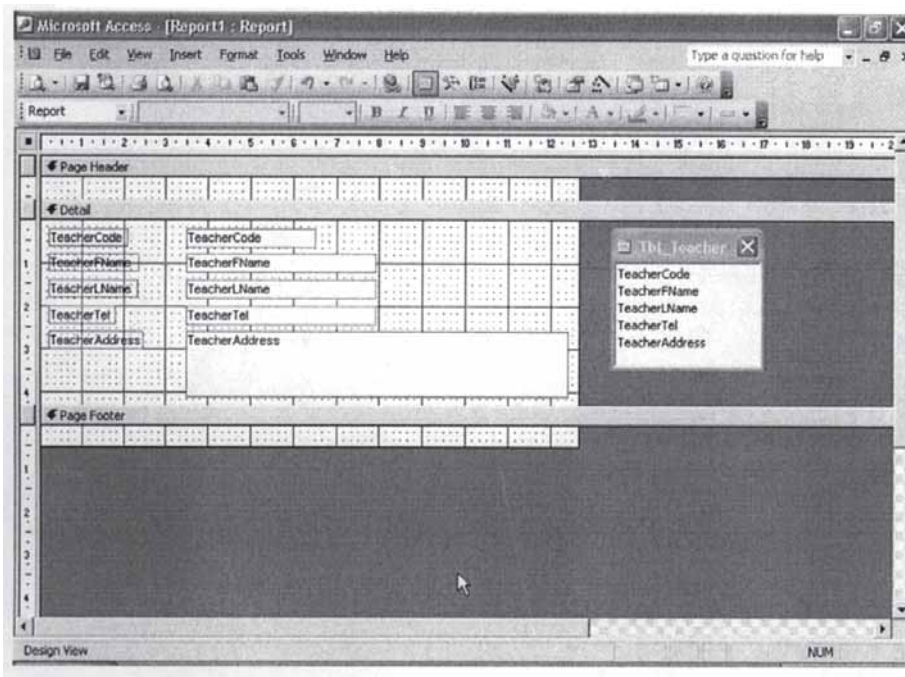
روی فلش سمت پایین در نوار ابزار مطابق شکل روبه‌رو، کلیک کنید.

● از لیست ظاهر شده، گزینه‌ی AutoReport را انتخاب نمایید.



بانک گزارشی از نمای Datasheet ساخته و آن را در نمای Preview نشان می‌دهد.  
● اگر نمایش گزارش مطابق آنچه می‌خواهید باشد، آن را با دکمه‌ی Print چاپ کنید.

● دکمه‌ی Design را کلیک کنید تا طرح گزارش را ببینید (شکل ۴-۶).  
به‌خاطر داشته باشید که شاید درک این نما برای بار اول کمی مشکل به‌نظر رسد.



● نگران اقلام خاص روی صفحه نباشید. AutoReport 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 گزینه‌ی Wizard استفاده می‌شود.



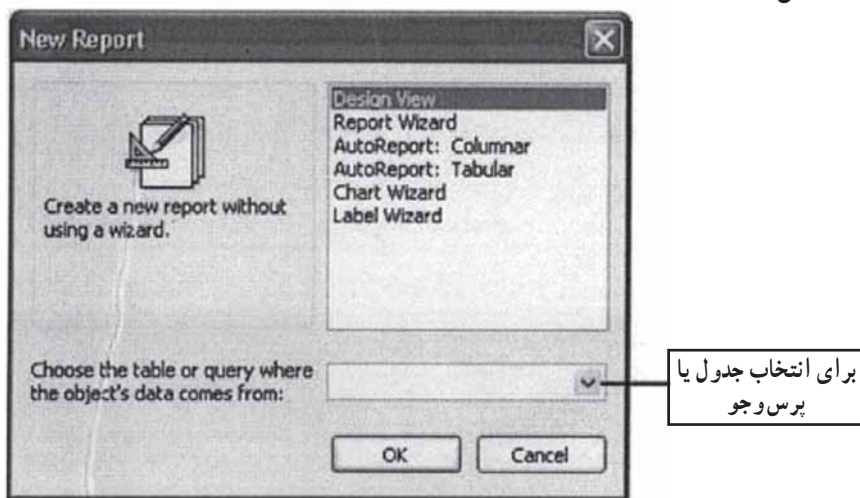
**نکته:** AutoReport ابزارهای طراحی گزارش را برای ایجاد آن به کار می‌برد. زمانی که یاد گرفتید چگونه از این ابزارها استفاده کنید و گزارش را اصلاح کنید و تغییر دهید، می‌توانید از این امکان برای ایجاد گزارش اولیه استفاده کنید و سپس گزارش را مطابق نظر خود تغییر دهید.

## کنجکاو

AutoForm چه کاری انجام می‌دهد، کجا در دسترس است؟

### ۶-۲- ایجاد گزارش به وسیله‌ی Report Wizard

روی پنجره‌ی بانک، گزینه‌ی Reports را انتخاب کنید. از پنجره‌ی باز شده Report Wizard را انتخاب نمایید و در قسمت Choose the table or query، جدول یا پرس‌وجوی مورد نظر را انتخاب کنید (شکل ۶-۲).



شکل ۶-۲- انتخاب جدول یا پرس‌وجو برای ایجاد گزارش

Access، ابزارهای متنوعی برای ایجاد گزارش دارد. همان‌طور که در شکل می‌بینید این روش‌ها به اجمال عبارت است از:


**Design View:** گزارش خالی را در نمای Design باز می‌کند. در این نما می‌توانید عنوان، پانوشته‌ها، جزئیات و خلاصه‌ی 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/ پرس و جوها)، فیلدهای مورد نظر در گزارش و غیره گزارش تولید می‌شود. این روش ایجاد گزارش معمول‌ترین است.

**AutoReport Columnar** : گزارشی را ایجاد می‌کند که تمام فیلدهای جدول یا پرس و جوی انتخاب‌شده را دربر دارد. شبیه AutoReport است. با این تفاوت که از فضای صفحه استفاده‌ی بهتری دارد.

**AutoReport Tabular** : گزارشی تولید می‌کند که رکوردهای جدول/پرس و جو را در یک ردیف نشان می‌دهد و اندازه‌ی قلم را به‌گونه‌ای اختیار می‌کند که صفحه‌ی گزارش تنظیم شود. این نوع گزارش بهتر از نمای Datasheet ساده به نظر می‌آید و اغلب مفیدتر از AutoReport‌های قبلی است (شکل ۳-۶).



TeacherCode	TeacherName	TeacherName	TeacherTel	TeacherAddress
۱-۲۱	سازمان	سازمان	۰۲۱-۱۱۱۱۱۱۱۱	
۱-۲۲	سازمان	سازمان	۰۲۱-۱۱۱۱۱۱۱۱	
۱-۲۳	سازمان	سازمان	۰۲۱-۱۱۱۱۱۱۱۱	
۱-۲۴	سازمان	سازمان	۰۲۱-۱۱۱۱۱۱۱۱	
۱-۲۵	سازمان	سازمان	۰۲۱-۱۱۱۱۱۱۱۱	
۱-۲۶	سازمان	سازمان	۰۲۱-۱۱۱۱۱۱۱۱	
۱-۲۷	سازمان	سازمان	۰۲۱-۱۱۱۱۱۱۱۱	

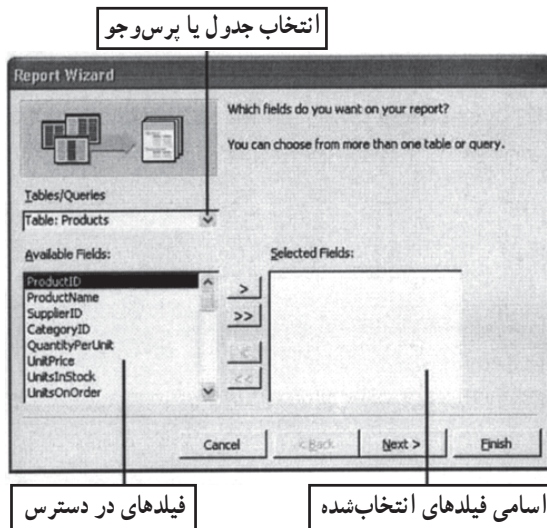
شکل ۳-۶ - پنجره‌ی نمای Preview گزارش ایجادشده به وسیله‌ی Wizard

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.

**Chart Wizard** : گراف داده‌ها را نمایش می‌دهد. گراف این محیط شبیه محیط Excel است و به همان صورت قابل کنترل است.

**Label Wizard** : لیستی از برچسب‌ها ایجاد می‌کند. برچسب پستی، استاندارد شماره‌گذاری خاص خود را دارد که فراهم شده است.

- با کمک ویزارد گزارشی از لیست معلم‌های مدرسه و از نوع Tabular ایجاد کنید.
- در قسمت گزارش‌های بانک، دکمه‌ی New و سپس Report Wizard را انتخاب کنید و OK نمایید.
- پرس‌وجویی ایجاد کنید که تمام رکوردهای جدول معلمان را برگرداند و آن را Qry\_Teachers نام‌گذاری کنید.
- در پنجره‌ی بازشده، در قسمت Tables /Queries جدول Qry\_Teachers را انتخاب کنید.



- تمام فیلدها را با دکمه‌ی >> به لیست انتخاب‌شده اضافه کنید.
- قسمت Selected Field، فیلدهای انتخاب‌شده برای نمایش در گزارش را نشان می‌دهد. فیلدها به وسیله‌ی 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 انتخاب‌شده‌ی خود اضافه کرد. اگرچه گفتیم بهتر است ابتدا پرس‌وجویی 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، دکمه‌ی Next را کلیک کنید.

- سایر مراحل را بدون تغییر تایید کنید و در انتهای Finish را انتخاب کنید.
- گزارش را در حالت Design یا Preview مشاهده کنید. (شکل زیر)

The screenshot shows a Microsoft Access window titled 'RepWizard\_Teacher'. The report displays a table with the following data:

TeacherCode	TeacherFName	TeacherLName	TeacherTel	TeacherAddress
۱۰۲۵	طاهرخا	سادی زاد	۰۲۱۲.۱۳۳۹۹۱۰	
۱۰۲۶	محمد	مهدی فر		
۱۰۲۷	طهرخا	ساموئان	۰۲۱.۷۷۶۵۵۶۰	
۱۰۲۸	کامیل	خلایق فر	۰۲۱۲.۵۵۵۳۳۳	
۱۰۳۹	امیرعلی	سلسان		
۱۰۵۰	محمد	مسورطیت		
۱۰۶۰	محمد	هلیت زاد		
۱۰۶۱	طهرخا	سلفی پور	۰۲۱.۹۹۵۰۹۸۳۳	

### کنجکاو

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

### مثال ۳-۶

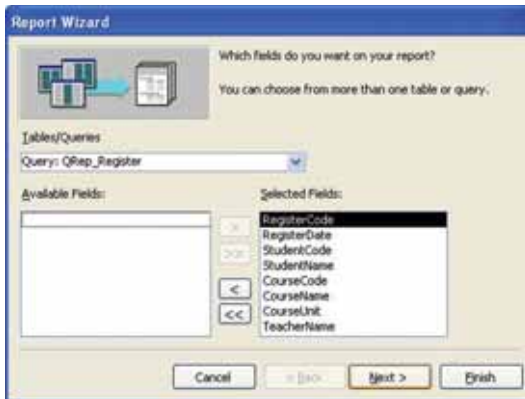
گزارش ثبت نام دانش آموزان در ترم را همان گونه که در طراحی در فصل اول مشخص شده است، ایجاد کنید.

- ابتدا پرس و جویی به نام QRep\_Register ایجاد می کنیم که اطلاعات مورد نیاز را در بر داشته باشد. SQL مورد نظر در زیر آمده است:

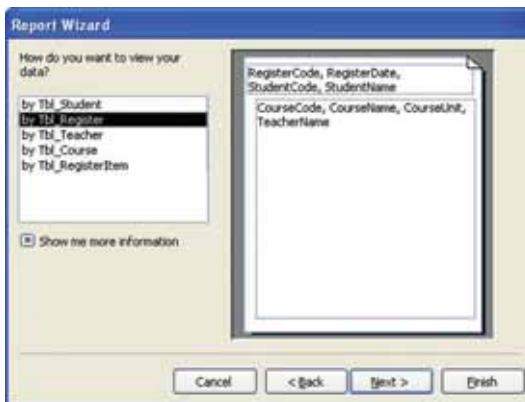
```
SELECT Tb1_Register. RegisterCode, Tb1_Register. RegisterDate, Tb1_Register. StudentCode, [StudentFName] ' '+[StudentLName] AS StudentName,
```

```
Tb1_RegisterItem.CourseCode,Tb1_Course.CourseName,
Tb1_Course.CourseUnit,
Tb1_Teacher. TeacherFName + ' ' + Tb1_Teacher. Teacher LName 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 دکمه‌ی Next را انتخاب  
می‌کنیم. (شکل روبه‌رو)

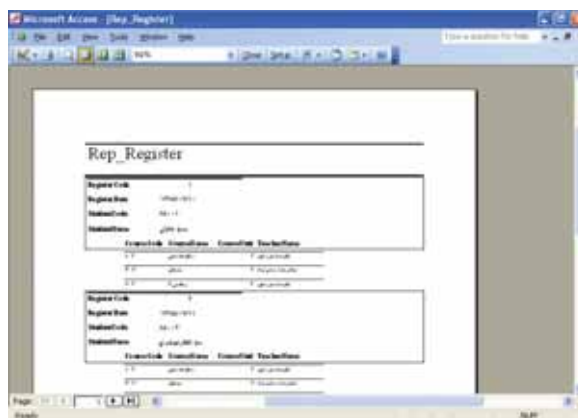


– از پنجره‌ی باز شده  
گزینه‌ی Tb1\_Register را  
انتخاب و دکمه‌ی Next را کلیک  
کنید.

– پیش فرض Group level و Sort Order را بدون تغییر تایید کنید، در پنجره‌ی Layout مطابق شکل زیر گزینه‌ی Align Left<sup>2</sup> را انتخاب کنید و دکمه‌ی Next را کلیک کنید.

– در پنجره‌ی Style، پیش فرض Formal را انتخاب کرده، دکمه‌ی Next را انتخاب کنید.


– نام گزارش را Rep\_Register قرار دهید و دکمه‌ی 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 دکمه‌ی 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 پنجره‌ی مشخصه‌ها و از طریق دکمه‌ی  روی نوار ابزار، این کار را انجام دهیم. با استفاده از زبانه‌ی Format، گزینه‌ی Caption متن 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 گزینه‌ی Text Align در زبانه‌ی Format پنجره‌ی مشخصه‌ها نیز این کار را انجام دهیم.

– برای افزودن امکانات گرافیکی، از منوی View، گزینه‌ی Toolbox را انتخاب می‌نماییم. پنجره‌ی زیر ظاهر می‌شود. این جعبه دقیقاً مشابه ابزار محیط طراحی فرم است.



– با کمک ابزار Rectangle (مستطیل) و Line (خط) طراحی گزارش مورد نظر خود را ایجاد می‌کنیم (شکل زیر).



– دقت کنید که برای وضوح بیشتر بین دو بخش عنوان‌های درس و اقلام داده‌ای فاصله ایجاد کرده‌ایم که در عمل برای زیبایی نباید وجود داشته باشند (شکل زیر).



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



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



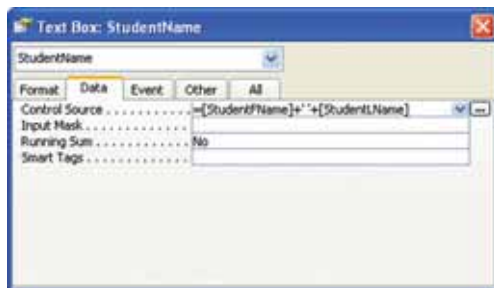
## مثال ۴-۶

گزارش کارنامه دانش‌آموزان ایجاد می‌کنیم.  
– این گزارش مشابه گزارش‌های قبلی ایجاد می‌شود، فقط کمی پیچیدگی در محاسبه‌ی معدل وجود دارد که به آن می‌پردازیم.  
– ابتدا پرس‌وجوی مورد نظر را ایجاد می‌کنیم. به نام QRep\_Certificate و با SQL زیر:

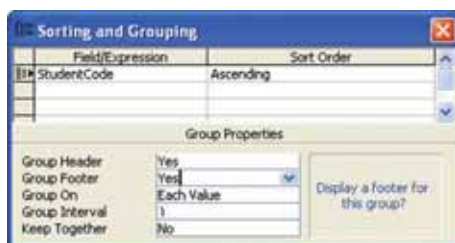
```
SELECT Tb1_Student.StudentCode, StudentFName, StudentLName, RegisterDate, MajorName, Tb1_RegisterItem.CourseCode, CourseName, CourseUnit, Grade FROM Tb1_Course INNER JOIN (Tb1_Major INNER JOIN ((Tb1_Student INNER JOIN Tb1_Register ON Tb1_Student.StudentCode= Tb1_Register.StudentCode) INNER JOIN Tb1_RegisterItem ON Tb1_Register.RegisterCode = Tb1_RegisterItem.RegisterCode) ON Tb1_Major.MajorCode = Tb1_Student.MajorCode) ON Tb1_Course.CourseCode = Tb1_RegisterItem.CourseCode
```

– با کمک ویزارد و این پرس‌وجوی اولیه‌ی مشابه مثال قبلی و به نام Rep\_Certificate ایجاد می‌کنیم.

– دقت نمایید که ما در پرس‌وجوی نام و نام خانوادگی را ترکیب نکرده‌ایم، بنابراین باید در گزارش این کار صورت گیرد. برای این منظور، در پنجره‌ی مشخصه‌ها، زبانه Data، گزینه‌ی Control Source به صورت زیر تنظیم می‌کنیم:



– از منوی View، گزینه‌ی Sorting and Grouping را انتخاب می‌کنیم.



– در این پنجره گزینه‌ی Group Footer را مقدار Yes انتخاب می‌کنیم، تا پانویشت گروه‌بندی نیز نمایش داده شود. جعبه‌متن‌های مربوط به جمع واحدها و محاسبه‌ی معدل باید در این قسمت قرار گیرد. – از جعبه ابزار، دکمه‌ی Text box را انتخاب می‌کنیم و با درگ کردن، آن را در قسمت پانویشت گروه قرار می‌دهیم. مقدار Visible آن را در پنجره‌ی مشخصه‌ها No انتخاب می‌کنیم و مقدار Control Source آن را  $=SUM([CourseUnit]*[Grade])$  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 زبانه‌ی All یا Other، گزینه‌ی Name مقدار SumGradeByUnit قرار می‌دهیم.

– Text box دیگری را در قسمت پانویشت گروه، زیر ستون واحدها قرار می‌دهیم و مقدار Control Source آن را  $=Sum([CourseUnit])$  قرار می‌دهیم. این فیلد جمع کل واحدهای ثبت‌نام‌شده‌ی دانش‌آموز را برمی‌گرداند، زیرا در پانویشت گروه قرار گرفته است و جمع را براساس گروه محاسبه می‌کند. نام آن را به‌وسیله‌ی گزینه‌ی Name در پنجره‌ی مشخصه‌ها، مقدار SumUnit قرار می‌دهیم. – Text box دیگری را در قسمت پانویشت گروه قرار می‌دهیم و مقدار Control Source آن را  $Format(SumUnit, "#.##")$  قرار می‌دهیم. همچنین

گزینه‌ی Border Style را مقدار

Solid قرار می‌دهیم.

– طراحی گزارش را در شکل

روبه‌رو مشاهده می‌کنید.



– نمای Preview گزارش به صورت زیر است.



گزارش کارنامه دانش آموز

شماره دانش آموز: ۸۵۰۰۱

نام دانش آموز: محمد هاشمی

پشته تحصیلی: دانش و فرهنگ

شماره درس	نام درس	نمره	تعداد واحد	نوع
۸۰۱	ریاضیات	۸۰	۳	۳
۸۰۲	علوم	۷۵	۳	۳
۸۰۳	ادبیات	۸۵	۳	۳
۸۰۴	تاریخ	۸۰	۳	۳
۸۰۵	زبان	۸۵	۳	۳
۸۰۶	هنر	۸۰	۳	۳
۸۰۷	موسیقی	۸۵	۳	۳
۸۰۸	فلسفه	۸۰	۳	۳
۸۰۹	فیزیک	۸۵	۳	۳
۸۱۰	شیمی	۸۰	۳	۳
۸۱۱	بیولوژی	۸۵	۳	۳
۸۱۲	صنعت	۸۰	۳	۳
۸۱۳	کامپیوتر	۸۵	۳	۳
۸۱۴	انگلیسی	۸۰	۳	۳
۸۱۵	فرهنگ و هنر	۸۵	۳	۳
۸۱۶	تربیت بدنی	۸۰	۳	۳
۸۱۷	سنگین	۸۵	۳	۳
۸۱۸	سبک	۸۰	۳	۳
۸۱۹	تفکر	۸۵	۳	۳
۸۲۰	خلاقیت	۸۰	۳	۳
۸۲۱	مهارت	۸۵	۳	۳
۸۲۲	توانمندی	۸۰	۳	۳
۸۲۳	خودآگاهی	۸۵	۳	۳
۸۲۴	توانمندی	۸۰	۳	۳
۸۲۵	خودآگاهی	۸۵	۳	۳
۸۲۶	توانمندی	۸۰	۳	۳
۸۲۷	خودآگاهی	۸۵	۳	۳
۸۲۸	توانمندی	۸۰	۳	۳
۸۲۹	خودآگاهی	۸۵	۳	۳
۸۳۰	توانمندی	۸۰	۳	۳
۸۳۱	خودآگاهی	۸۵	۳	۳
۸۳۲	توانمندی	۸۰	۳	۳
۸۳۳	خودآگاهی	۸۵	۳	۳
۸۳۴	توانمندی	۸۰	۳	۳
۸۳۵	خودآگاهی	۸۵	۳	۳
۸۳۶	توانمندی	۸۰	۳	۳
۸۳۷	خودآگاهی	۸۵	۳	۳
۸۳۸	توانمندی	۸۰	۳	۳
۸۳۹	خودآگاهی	۸۵	۳	۳
۸۴۰	توانمندی	۸۰	۳	۳
۸۴۱	خودآگاهی	۸۵	۳	۳
۸۴۲	توانمندی	۸۰	۳	۳
۸۴۳	خودآگاهی	۸۵	۳	۳
۸۴۴	توانمندی	۸۰	۳	۳
۸۴۵	خودآگاهی	۸۵	۳	۳
۸۴۶	توانمندی	۸۰	۳	۳
۸۴۷	خودآگاهی	۸۵	۳	۳
۸۴۸	توانمندی	۸۰	۳	۳
۸۴۹	خودآگاهی	۸۵	۳	۳
۸۵۰	توانمندی	۸۰	۳	۳
۸۵۱	خودآگاهی	۸۵	۳	۳
۸۵۲	توانمندی	۸۰	۳	۳
۸۵۳	خودآگاهی	۸۵	۳	۳
۸۵۴	توانمندی	۸۰	۳	۳
۸۵۵	خودآگاهی	۸۵	۳	۳
۸۵۶	توانمندی	۸۰	۳	۳
۸۵۷	خودآگاهی	۸۵	۳	۳
۸۵۸	توانمندی	۸۰	۳	۳
۸۵۹	خودآگاهی	۸۵	۳	۳
۸۶۰	توانمندی	۸۰	۳	۳
۸۶۱	خودآگاهی	۸۵	۳	۳
۸۶۲	توانمندی	۸۰	۳	۳
۸۶۳	خودآگاهی	۸۵	۳	۳
۸۶۴	توانمندی	۸۰	۳	۳
۸۶۵	خودآگاهی	۸۵	۳	۳
۸۶۶	توانمندی	۸۰	۳	۳
۸۶۷	خودآگاهی	۸۵	۳	۳
۸۶۸	توانمندی	۸۰	۳	۳
۸۶۹	خودآگاهی	۸۵	۳	۳
۸۷۰	توانمندی	۸۰	۳	۳
۸۷۱	خودآگاهی	۸۵	۳	۳
۸۷۲	توانمندی	۸۰	۳	۳
۸۷۳	خودآگاهی	۸۵	۳	۳
۸۷۴	توانمندی	۸۰	۳	۳
۸۷۵	خودآگاهی	۸۵	۳	۳
۸۷۶	توانمندی	۸۰	۳	۳
۸۷۷	خودآگاهی	۸۵	۳	۳
۸۷۸	توانمندی	۸۰	۳	۳
۸۷۹	خودآگاهی	۸۵	۳	۳
۸۸۰	توانمندی	۸۰	۳	۳
۸۸۱	خودآگاهی	۸۵	۳	۳
۸۸۲	توانمندی	۸۰	۳	۳
۸۸۳	خودآگاهی	۸۵	۳	۳
۸۸۴	توانمندی	۸۰	۳	۳
۸۸۵	خودآگاهی	۸۵	۳	۳
۸۸۶	توانمندی	۸۰	۳	۳
۸۸۷	خودآگاهی	۸۵	۳	۳
۸۸۸	توانمندی	۸۰	۳	۳
۸۸۹	خودآگاهی	۸۵	۳	۳
۸۹۰	توانمندی	۸۰	۳	۳
۸۹۱	خودآگاهی	۸۵	۳	۳
۸۹۲	توانمندی	۸۰	۳	۳
۸۹۳	خودآگاهی	۸۵	۳	۳
۸۹۴	توانمندی	۸۰	۳	۳
۸۹۵	خودآگاهی	۸۵	۳	۳
۸۹۶	توانمندی	۸۰	۳	۳
۸۹۷	خودآگاهی	۸۵	۳	۳
۸۹۸	توانمندی	۸۰	۳	۳
۸۹۹	خودآگاهی	۸۵	۳	۳
۹۰۰	توانمندی	۸۰	۳	۳
۹۰۱	خودآگاهی	۸۵	۳	۳
۹۰۲	توانمندی	۸۰	۳	۳
۹۰۳	خودآگاهی	۸۵	۳	۳
۹۰۴	توانمندی	۸۰	۳	۳
۹۰۵	خودآگاهی	۸۵	۳	۳
۹۰۶	توانمندی	۸۰	۳	۳
۹۰۷	خودآگاهی	۸۵	۳	۳
۹۰۸	توانمندی	۸۰	۳	۳
۹۰۹	خودآگاهی	۸۵	۳	۳
۹۱۰	توانمندی	۸۰	۳	۳
۹۱۱	خودآگاهی	۸۵	۳	۳
۹۱۲	توانمندی	۸۰	۳	۳
۹۱۳	خودآگاهی	۸۵	۳	۳
۹۱۴	توانمندی	۸۰	۳	۳
۹۱۵	خودآگاهی	۸۵	۳	۳
۹۱۶	توانمندی	۸۰	۳	۳
۹۱۷	خودآگاهی	۸۵	۳	۳
۹۱۸	توانمندی	۸۰	۳	۳
۹۱۹	خودآگاهی	۸۵	۳	۳
۹۲۰	توانمندی	۸۰	۳	۳
۹۲۱	خودآگاهی	۸۵	۳	۳
۹۲۲	توانمندی	۸۰	۳	۳
۹۲۳	خودآگاهی	۸۵	۳	۳
۹۲۴	توانمندی	۸۰	۳	۳
۹۲۵	خودآگاهی	۸۵	۳	۳
۹۲۶	توانمندی	۸۰	۳	۳
۹۲۷	خودآگاهی	۸۵	۳	۳
۹۲۸	توانمندی	۸۰	۳	۳
۹۲۹	خودآگاهی	۸۵	۳	۳
۹۳۰	توانمندی	۸۰	۳	۳
۹۳۱	خودآگاهی	۸۵	۳	۳
۹۳۲	توانمندی	۸۰	۳	۳
۹۳۳	خودآگاهی	۸۵	۳	۳
۹۳۴	توانمندی	۸۰	۳	۳
۹۳۵	خودآگاهی	۸۵	۳	۳
۹۳۶	توانمندی	۸۰	۳	۳
۹۳۷	خودآگاهی	۸۵	۳	۳
۹۳۸	توانمندی	۸۰	۳	۳
۹۳۹	خودآگاهی	۸۵	۳	۳
۹۴۰	توانمندی	۸۰	۳	۳
۹۴۱	خودآگاهی	۸۵	۳	۳
۹۴۲	توانمندی	۸۰	۳	۳
۹۴۳	خودآگاهی	۸۵	۳	۳
۹۴۴	توانمندی	۸۰	۳	۳
۹۴۵	خودآگاهی	۸۵	۳	۳
۹۴۶	توانمندی	۸۰	۳	۳
۹۴۷	خودآگاهی	۸۵	۳	۳
۹۴۸	توانمندی	۸۰	۳	۳
۹۴۹	خودآگاهی	۸۵	۳	۳
۹۵۰	توانمندی	۸۰	۳	۳
۹۵۱	خودآگاهی	۸۵	۳	۳
۹۵۲	توانمندی	۸۰	۳	۳
۹۵۳	خودآگاهی	۸۵	۳	۳
۹۵۴	توانمندی	۸۰	۳	۳
۹۵۵	خودآگاهی	۸۵	۳	۳
۹۵۶	توانمندی	۸۰	۳	۳
۹۵۷	خودآگاهی	۸۵	۳	۳
۹۵۸	توانمندی	۸۰	۳	۳
۹۵۹	خودآگاهی	۸۵	۳	۳
۹۶۰	توانمندی	۸۰	۳	۳
۹۶۱	خودآگاهی	۸۵	۳	۳
۹۶۲	توانمندی	۸۰	۳	۳
۹۶۳	خودآگاهی	۸۵	۳	۳
۹۶۴	توانمندی	۸۰	۳	۳
۹۶۵	خودآگاهی	۸۵	۳	۳
۹۶۶	توانمندی	۸۰	۳	۳
۹۶۷	خودآگاهی	۸۵	۳	۳
۹۶۸	توانمندی	۸۰	۳	۳
۹۶۹	خودآگاهی	۸۵	۳	۳
۹۷۰	توانمندی	۸۰	۳	۳
۹۷۱	خودآگاهی	۸۵	۳	۳
۹۷۲	توانمندی	۸۰	۳	۳
۹۷۳	خودآگاهی	۸۵	۳	۳
۹۷۴	توانمندی	۸۰	۳	۳
۹۷۵	خودآگاهی	۸۵	۳	۳
۹۷۶	توانمندی	۸۰	۳	۳
۹۷۷	خودآگاهی	۸۵	۳	۳
۹۷۸	توانمندی	۸۰	۳	۳
۹۷۹	خودآگاهی	۸۵	۳	۳
۹۸۰	توانمندی	۸۰	۳	۳
۹۸۱	خودآگاهی	۸۵	۳	۳
۹۸۲	توانمندی	۸۰	۳	۳
۹۸۳	خودآگاهی	۸۵	۳	۳
۹۸۴	توانمندی	۸۰	۳	۳
۹۸۵	خودآگاهی	۸۵	۳	۳
۹۸۶	توانمندی	۸۰	۳	۳
۹۸۷	خودآگاهی	۸۵	۳	۳
۹۸۸	توانمندی	۸۰	۳	۳
۹۸۹	خودآگاهی	۸۵	۳	۳
۹۹۰	توانمندی	۸۰	۳	۳
۹۹۱	خودآگاهی	۸۵	۳	۳
۹۹۲	توانمندی	۸۰	۳	۳
۹۹۳	خودآگاهی	۸۵	۳	۳
۹۹۴	توانمندی	۸۰	۳	۳
۹۹۵	خودآگاهی	۸۵	۳	۳
۹۹۶	توانمندی	۸۰	۳	۳
۹۹۷	خودآگاهی	۸۵	۳	۳
۹۹۸	توانمندی	۸۰	۳	۳
۹۹۹	خودآگاهی	۸۵	۳	۳
۱۰۰۰	توانمندی	۸۰	۳	۳

**نکته:** 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.

## خلاصه‌ی فصل

برای چاپ اطلاعات با قالبی زیبا و مناسب از گزارش استفاده می‌کنیم. گزارش نمای قابل چاپ را روی صفحه‌ی 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 گزارش‌ساز 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- روش‌های ایجاد گزارش را توضیح دهید.
- ۳- به نظر شما کدامیک از روش‌های ساخت گزارش مناسب‌تر است و چرا؟
- ۴- کنترل‌های گزارش را بررسی کنید، آیا اختلافی بین آن‌ها و کنترل‌های فرم وجود دارد؟

### نرمال سازی

✓ در پس از شناسایی موجودیت‌ها و رسم ER، صفات مورد نیاز و ارتباط بین موجودیت‌ها، می‌باید جدول‌ها (روابط) مورد نیاز طراحی شوند. یک تصور نادرست این است که هرچه صفات جدول بیشتر باشد، از اکمال و جامعیت بالاتری برخوردار است، حال آن‌که گزینش درست صفات، از بروز برخی مشکلات در بانک‌های اطلاعاتی اجتناب می‌کند. نارسایی‌ها در طراحی و یک طراحی نامناسب، باعث ایجاد پدیده‌ی «آنومالی»<sup>۱</sup> می‌گردد.

آنومالی، در سه محور قابل طرح و بررسی است :

الف - «انجام ناپذیری» یکی از عملیات در بانک.

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

ج - «فزون کاری» 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.

هدف‌های رفتاری: در پایان این فصل انتظار می‌رود که هنرجو بتواند :

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

۱- Anomaly

## ۱-۷- ویژگی‌های رابطه

رابطه که در بانک‌های اطلاعاتی رابطه‌ای<sup>۱</sup>، 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<sup>۲</sup> نظم ندارد.

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

ج - جدول، سطرهای تکراری ندارد.

## ۲-۷- وابستگی تابعی<sup>۳</sup>

شبهه تعریف تابع در ریاضیات، در رابطه‌ی  $R$ ، صفت  $y$  با صفت  $x$  وابستگی تابعی دارد اگر به‌ازای هر مقدار  $x$ ، تنها یک مقدار  $y$  وجود داشته باشد.

وابستگی تابعی  $A \rightarrow B$  را به صورت  $A \rightarrow B$  نشان می‌دهیم.<sup>۵</sup>

### مثال ۱-۷

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

$R_1$		
A	B	C
$a_1$	$b_1$	$c_1$
$a_2$	$b_2$	$c_3$
$a_1$	$b_1$	$c_2$
$a_3$	$b_4$	$c_2$
$a_5$	$b_1$	$c_1$

پاسخ: وابستگی  $A \rightarrow B$  برقرار است. زیرا به‌ازای هیچ مقدار مساوی از  $A$ ، دو

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 خاصی ساده یا مرکب باشند.

۵-  $A$  را اصطلاحاً دترمینان گویند.

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.

وابستگی تابعی بیان کننده‌ی قواعد محیط عملیاتی است. این قواعد را می‌توان در نمودار «وابستگی تابعی» یا نمودار FD نمایش داد.

## مثال ۲-۷

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

قاعده‌ی ۱: هر ناشر، تعدادی کتاب منتشر می‌کند.

قاعده‌ی ۲: هر ناشر از یک کتاب، شمارگان مشخصی منتشر می‌کند.

قاعده‌ی ۳: هر ناشر در یک شهر دفتر دارد.

قاعده‌ی ۴: هر ناشر دارای یک رتبه‌ی صنفی است.

قاعده‌ی ۵: ناشران یک شهر دارای یک رتبه‌ی صنفی هستند.

پاسخ: با توجه به قواعد بالا داریم:

$P\# = \text{شماره ناشر}$ ،  $Book\# = \text{شماره کتاب}$ ،  $Qty = \text{شمارگان}$ ،  $City = \text{شهر}$  و

$Grade = \text{رتبه صنفی}$  (ناشر  $\rightarrow$  ناشر)

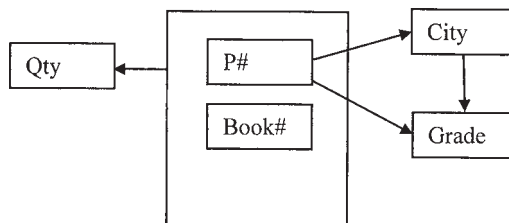
1)  $P\# \rightarrow Book\#$

2)  $(P\# \rightarrow, Book\#) \rightarrow Qty$

3)  $P\# \rightarrow City$

4)  $P\# \rightarrow Grade$

5)  $City \rightarrow Grade$



FIRST

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

**تمرین ۷-۱:** در یک محیط عملیاتی چهار قاعده بیان کنید و نمودار FD متناظر را رسم کنید.

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

### مثال ۷-۳

اگر  $A \rightarrow B$  و  $B \rightarrow C$  برقرار باشد، آیا  $A \rightarrow C$  برقرار است؟

پاسخ: فرض می‌کنیم  $A \not\rightarrow C$  در این صورت داریم:

دو حالت روبه‌رو قابل تصور است:

در هر دو حالت با تناقض در فرض مواجه می‌شویم. در قسمت (الف)  $B \rightarrow C$  و در قسمت (ب)  $A \rightarrow B$  در نتیجه طبق برهان خلف داریم:  $A \rightarrow C$ .

### ۷-۳-۱ وابستگی تابعی کامل<sup>۱</sup>

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 زیرمجموعه‌ی  $x$  وابستگی تابعی نداشته باشد.

### مثال ۷-۴

با توجه به  $R_2$  تحقیق کنید، آیا وابستگی تابعی کامل بین  $C$  و  $(A$  و  $B)$  برقرار است؟

R2		
A	B	C
a <sub>1</sub>	b <sub>1</sub>	c <sub>1</sub>
a <sub>2</sub>	b <sub>1</sub>	c <sub>3</sub>
a <sub>1</sub>	b <sub>2</sub>	c <sub>2</sub>



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

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

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

در نتیجه:  $(A, B) \Rightarrow C$

**نکته:** 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 رابطه‌ی 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 رتبه‌ی صنفی ناشران شهر  $c_2$  (مقدار  $6^\circ$ ) 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 رتبه‌ی صنفی ناشران ساکن شهر  $c_3$  از  $5^\circ$  به  $7^\circ$  تغییر کند. این عمل موجب به‌هنگام‌سازی منتشرشونده<sup>۱</sup> (فزون‌کاری) در سیستم می‌شود، یعنی در بیش از یک سطر جدول باید تغییر صورت گیرد.

همان‌طور که مشاهده می‌شود، رابطه‌ی FIRST (مثال ۷-۲) 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.

## ۷-۵- 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 BCNF<sup>۱</sup>

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.

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

رابطه‌ی R در سطح نرمال اول (1NF) است، اگر تمام فیلدهای آن در هر سطر جدول، تک‌مقداری باشد.

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

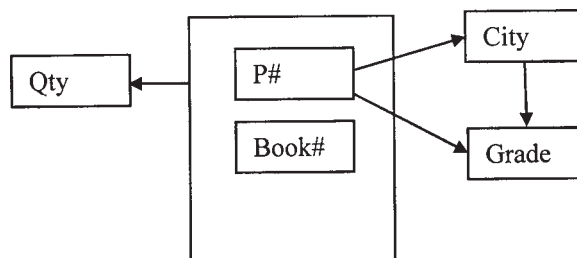
رابطه‌ی کارمند را در نظر بگیرید، آیا این رابطه در سطح INF است؟

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 رابطه‌ی 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:

$P\# \rightarrow City$

$Book\# \rightarrow City$

اما همانطور که در نمودار FD قابل مشاهده است، داریم:

$P\# \rightarrow City$

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

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

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

برای رفع آنومالی و افزایش سطح نرمال رابطه، می‌باید رابطه‌ی FIRST را تجزیه نماییم. برای این کار FIRST را به صورت زیر تجزیه می‌کنیم:

SECOND (P#, City, Grade)

PB (P#, Book#, Qty)

SECOND

P#	City	Grade
P <sub>1</sub>	c <sub>1</sub>	40
P <sub>2</sub>	c <sub>2</sub>	60
P <sub>3</sub>	c <sub>1</sub>	40
P <sub>4</sub>	c <sub>3</sub>	50

PB

P#	Book#	Qty
P <sub>1</sub>	b <sub>1</sub>	1000
P <sub>2</sub>	b <sub>4</sub>	3000
P <sub>3</sub>	b <sub>3</sub>	2000
P <sub>1</sub>	b <sub>2</sub>	3000
P <sub>4</sub>	b <sub>3</sub>	4000
P <sub>4</sub>	b <sub>5</sub>	3500

ملاک تجزیه‌ی رابطه: تجزیه‌ی رابطه R به رابطه‌های  $R_1$  و  $R_2$  می‌باید به نحوی باشد که پیوند<sup>۱</sup> دو رابطه‌ی  $R_1$  و  $R_2$  رابطه‌ی R را ایجاد کند و تاپلی (رکورد یا سطر) کم و زیاد نشود. از طرف دیگر تجزیه‌ی R می‌باید، وابستگی‌های تابعی را حفظ کند.

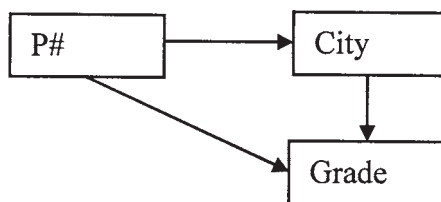
**تمرین ۲-۷:** آیا معیارهای تجزیه‌ی مناسب در تجزیه‌ی FIRST لحاظ شده است؟

رابطه‌ی 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: رابطه‌ی 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.

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

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

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

PC(P#,City)

CS (City, Grade)

PS		CS	
P#	City	City	Grade
P <sub>1</sub>	c <sub>1</sub>	c <sub>1</sub>	40
P <sub>2</sub>	c <sub>2</sub>	c <sub>2</sub>	60
P <sub>3</sub>	c <sub>1</sub>	c <sub>3</sub>	50
P <sub>4</sub>	c <sub>3</sub>		

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.

### تمرین ۶-۷

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

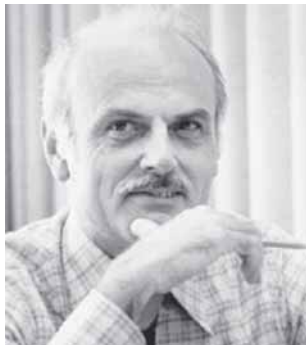
اگرچه نرمال‌سازی، آنومالی‌ها را از بین می‌برد اما در پرس‌وجوها، به علت نیاز به پیوند رابطه‌ها (جدول‌ها) سرعت انجام پرس‌وجو کاهش می‌یابد.

### کنجکاو

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

## ادگار فرانک تد کاد (۲۳ اگوست 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 دانشکده‌ی 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 مقاله‌ی خود را در مورد مدل رابطه‌ای برای بانک‌های داده‌ی مشترک و بزرگ<sup>۱</sup> منتشر کرد.

کاد بر توسعه و بسط مدل رابطه‌ای خود ادامه داد و با کریس دیت<sup>۲</sup> 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 گانه‌ی وی شد که یک پایگاه داده‌ی 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 هم‌راهانش، شرکتی مشاوره‌ای راه‌اندازی کرد.

<sup>۱</sup>—“A Relational Model of Data for Large Shared Data Banks”

<sup>۲</sup>—Chris Date

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

### کریستوفر جی دیت (۲۳ آگوست ۱۹۲۳ – ۱۸ آوریل ۲۰۰۳)



دیت یکی از نویسندگان، محققان، مشاوران و متخصصان مستقل در زمینه‌ی فن‌آوری بانک داده‌ی رابطه‌ای است. زمانی که برای شرکت IBM کار می‌کرد، در زمینه‌ی 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 به‌عنوان مقدمه‌ای بر سیستم‌های بانک داده، تاکنون به هشتمین ویرایش خود رسیده است و کتابی مرجع در زمینه‌ی بانک‌های داده‌ای است و صدها دانشکده و دانشگاه در سراسر دنیا تدریس می‌شود. وی همچنین نویسنده‌ی کتاب‌های بسیاری در زمینه‌ی مدیریت داده است.

### خلاصه‌ی 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. تجزیه شدن جدول به جدول‌های جدید که حافظ اطلاعات جدول قبلی باشد، آنومالی‌ها را از بین می‌برد ولی در مواردی سرعت بازیافت اطلاعات در پرس‌وجوها را افزایش می‌دهد.

## خودآزمایی

- 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 خاصی غیرکلید و کلید اصلی در FIRST، نقض شده است.