

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 حامدرضا
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 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 ۵-۱۸

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 ۵-۱۹



**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 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.

## 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.

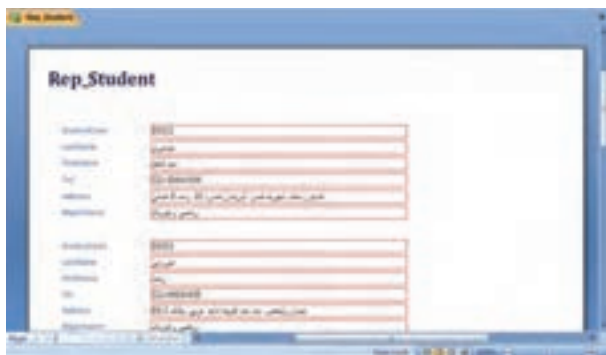
یک گزارش، رکوردها را شبیه نمای Datasheet فرم‌ها و جدول‌ها نشان می‌دهد. تفاوت نمای View در گزارش با Datasheet آن است که در گزارش خلاصه آماری، عناوین، زیرنویس، شماره صفحه، قالب نمایش داده زیباتری فراهم می‌شود. همچنین می‌توان داده‌های قابل نمایش را انتخاب کرد، داده‌ها را گروه‌بندی کرد و جمع کل روی آن گروه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

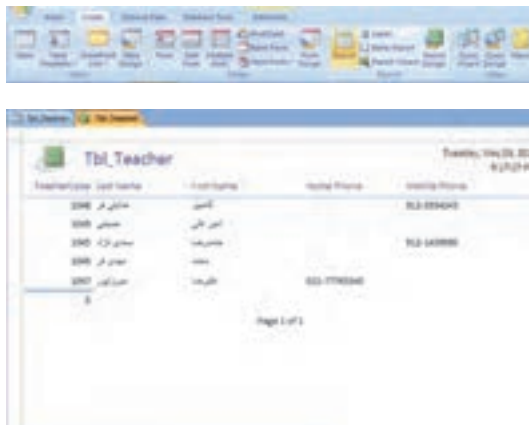
معمولاً نباید تمام داده‌ها در یک گزارش نمایش داده شوند، مگر گزارش برای نمایش جزئیات باشد مثل: لیست موجودی انبار، لیست مشتریان و غیره. به‌طورکل بخشی از اطلاعات و مقادیر جدول‌ها که مفید و برای تصمیم‌گیری مورد نیاز است نمایش داده می‌شوند.

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.

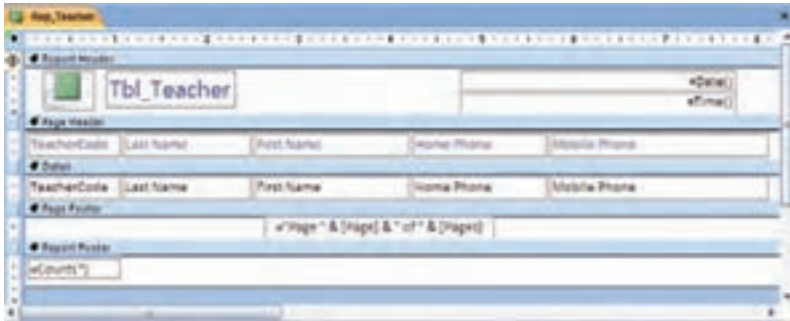
## ۱-۶- ایجاد گزارش ساده به‌وسیلهٔ AutoReports

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 Tb1 - Teacher.  
 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.



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

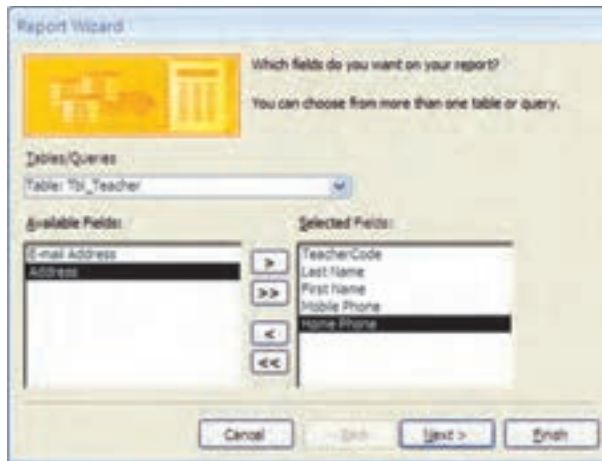
Access معمولاً نمی‌تواند عناوین فیلدها را در بالای گزارش به صورت صحیح قرار دهد. برای تنظیم دقیق همانطور که در ادامه خواهیم دید باید در نمای Design آن را اصلاح نمایید.

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

– پرس و جویی 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 و جویی 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؟

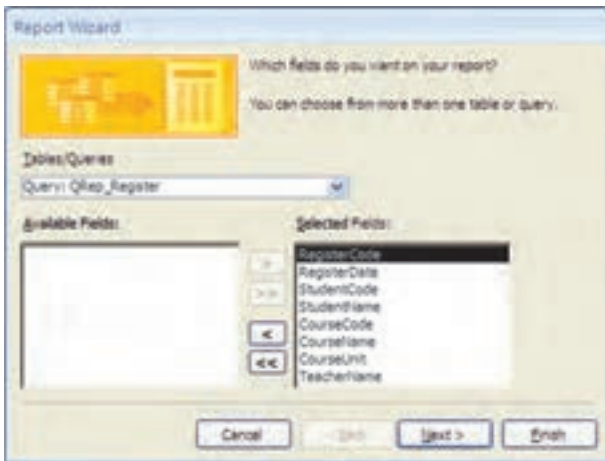
گزارش ثبت‌نام دانش‌آموزان در ترم را همان‌گونه که در طراحی در فصل اول مشخص شده است، ایجاد کنید.

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

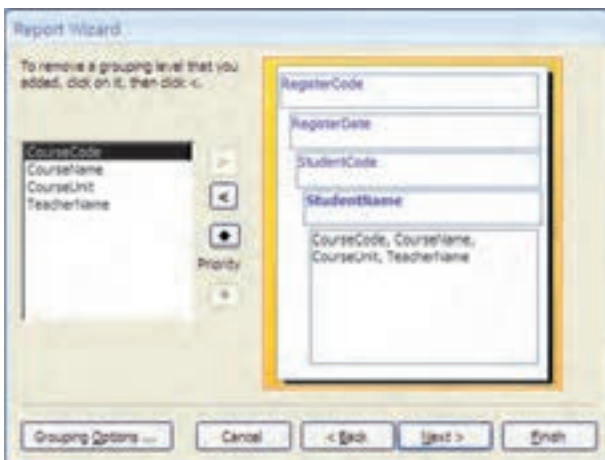
```
SELECT Tb1-Register.RegisterCode,Tb1-Register.RegisterDate,
Tb1-Register.StudentCode,[FirstName]+'+[LastName] AS
StudentName,
Tb1-RegisterItem.CourseCode,Tb1-Course.CourseName,
Tb1-Course.CourseUnit,
Tb1-Teacher.FirstName+'+'+Tb1-Teacher.LastName AS
```

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.



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 پرس‌وجوی مورد نظر را ایجاد می‌کنیم. به نام QRep\_Certificate و با

SQL زیر:

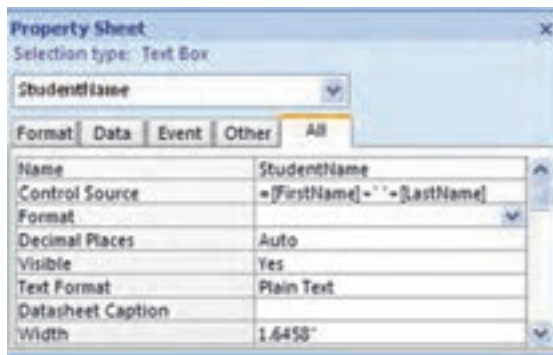
```
SELECT Tb1_Student.StudentCode, FirstName, LastName,
RegisterDate,
MajorName, Tb1_RegisterItem.CourseCode, CourseName, Course
Unit, Grade
FROM Tb1_Course INNER JOIN (Tb1_Major INNER JOIN ((Tb1_
Student INNER JOIN
Tb1_Register ON Tb1_Student.StudentCode= Tb1_Register.
StudentCode) INNER JOIN
Tb1_RegisterItem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

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<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 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<sup>۵</sup>.

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

$R_1$

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

مثال ۷-۱

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.

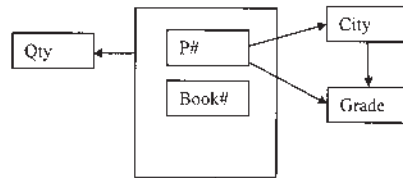
قاعده ۴: هر ناشر دارای یک رتبه صنفی است.

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



## تمرین ۱-۷:

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.

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

## ۳-۷- 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<sup>۲</sup>

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:  $A \nrightarrow C ((a_1, c_1), (a_1, c_2))$

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

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  $6^\circ$ ) 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 رتبهٔ 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، 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 فیلد 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.

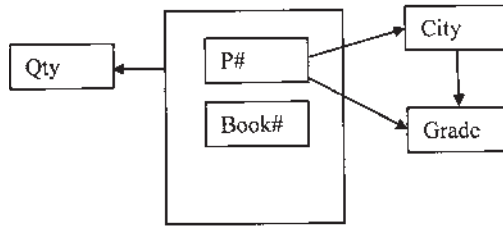
رابطه کارمند را در نظر بگیرید، آیا این رابطه در سطح INF است؟

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

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:

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

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 <sub>1</sub>	c <sub>1</sub>	40	P <sub>1</sub>	b <sub>1</sub>	1000
P <sub>2</sub>	c <sub>2</sub>	60	P <sub>2</sub>	b <sub>4</sub>	3000
P <sub>3</sub>	c <sub>1</sub>	40	P <sub>3</sub>	b <sub>3</sub>	2000
P <sub>4</sub>	c <sub>3</sub>	50	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 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.

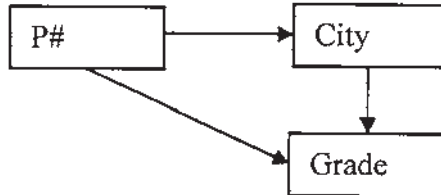
تمرین ۲-۷: آیا معیارهای تجزیه مناسب در تجزیه FIRST لحاظ شده است؟

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# → City
2. City → Grade
3. P# → 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 <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.

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<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 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. 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 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.

$$C \rightarrow D$$

$$A \rightarrow C$$