

فصل پنجم:
SQL

SQL (Structured Query Language)

زبان SQL یک زبان مخصوص کار با بانک اطلاعات (DSL) است که هر سه دسته دستورات DDL، DML و DCL را شامل می شود. انواع داده ای مختلفی توسط SQL پشتیبانی می شود که موجب می شود در طراحی جداول در محیطهایی مثل SQL Server بتوان ستونهایی با نوعهای دلخواه تعریف کرد.

انواع داده ای در SQL

لیستی از مهمترین انواع داده ای که توسط SQL پشتیبانی می شود بشرح زیر است:

انواع عددی

- tinyint : عدد صحیح ۱ بایتی مثبت (۰ تا ۲۵۵)
- smallint : عدد صحیح ۲ بایتی (۳۲۷۶۸- تا ۳۲۷۶۷)
- int : عدد صحیح ۴ بایتی (۲۱۴۷۴۸۳۶۴۷- تا ۲۱۴۷۴۸۳۶۴۸)
- bigint : عدد صحیح ۸ بایتی
- decimal : اعداد حقیقی از 10^{38} - 1 تا 10^{38}
- real : اعداد اعشار ۴ بایتی با دقت ۷ رقم
- float : اعداد اعشار ۸ بایتی با دقت ۱۵ رقم

انواع کاراکتری

- char : رشته تا حداکثر ۸۰۰۰ کاراکتر : ذخیره با طول ثابت
- varchar : رشته تا حداکثر ۸۰۰۰ کاراکتر : ذخیره با طول متغیر
- nchar : رشته تا حداکثر ۴۰۰۰ کاراکتر: ذخیره با طول ثابت و بصورت Unicode
- nvarchar : رشته تا حداکثر ۴۰۰۰ کاراکتر: ذخیره با طول متغیر و بصورت Unicode

انواع دیگر

- bit : مقادیر منطقی ۰ و ۱
- text : متن شامل حداکثر ۲۱۴۷۴۸۳۶۴۷ کاراکتر
- ntext : متن شامل حداکثر ۱۰۷۳۷۴۱۸۲۳ کاراکتر: ذخیره Unicode
- Money : نوع پول (با واحد پولی تنظیم شده در سیستم)
- DateTime : تاریخ (میلادی) و ساعت (با فرمت نمایشی تنظیم شده در سیستم)

- image : برای ذخیره هر نوع داده با حجم زیاد تا ۲۱۴۷۴۸۳۶۴۷ بایت، مثلا برای ذخیره یک فایل یا یک عکس در بانک اطلاعات

دستورات SQL:

دستورات SQL بسیار گسترده و متنوعند. برخی از این دستورات برای کار با ساختارهای کلی هستند و به داده های کاربر ربطی ندارند. به این نوع دستورات، دستورات DDL (Data Definition Language) گفته می شود. دسته دیگری از دستورات، دستورات کار با داده ها و موسوم به DML (Data Manipulation Language) هستند. نوع سوم دستورات، دستورات DCL (Data Control Language) هستند که امکان کنترل داده ها در جهت جامعیت و جلوگیری از دسترسی های غیر مجاز را فراهم می کنند.

کلمات کلیدی مورد استفاده در دستورات SQL

قبل از معرفی دستورات SQL، برای اینکه معمولا دستورات DDL (کار با اسکیمای) و DML (کار با داده ها) بدلیل برخی شباهتها با هم اشتباه گرفته می شوند، لیستی از کلمات کلیدی مورد استفاده در این دو نوع از دستورات (برای عملیات اضافه، حذف و تغییر) بصورت زیر معرفی می شود:

کار با اسکیمای	کار با داده	
جدید	Insert	Create , Add
حذف	Delete	Drop
تغییر	Update	Alter

دستورات کار با اسکیمای (DDL)

تعریف جدول جدید

```
CREATE TABLE "table_name"
("column 1" "data_type",
"column 2" "data_type",
...
)
```

مثال: دستور ساخت جدول اطلاعات دانشجویان:

```
CREATE TABLE Students
( Id int primary key,
```

```
Name Nvarchar(15) not null,  
SurName Nvarchar(20) not null,  
Field Nvarchar(10)  
)
```

مثال: دستور ساخت جدول شماره تلفنها:

```
CREATE TABLE TelNums  
( Id int Identity,  
  TelNum Nvarchar(15),  
  Type Nvarchar(5),  
  SId int,  
  Primary Key (Id),  
  Foreign Key (SId) references Students(Id)  
  جدول دیگر //  
)
```

// ستون خود افزایشی

// کلید خارجی

// مشخص کردن کلید اصلی

برقراری ارتباط بین کلید خارجی با کلید اصلی

از بین بردن یک جدول

```
DROP TABLE "table_name"
```

مثال: از بین بردن جدول دانشجویان:

```
Drop TABLE Students
```

تغییر در اسکیمای یک جدول

```
Alter TABLE "table_name"  
[Drop| Add | Alter] column ...
```

چند مثال:

۱- افزودن یک فیلد جدید

```
Alter TABLE Students Add BirthDate nvarchar(20)
```

۲- تغییر در تعریف یک فیلد

```
Alter TABLE Students Alter column name nvarchar(6) null
```

۳- حذف یک فیلد

Alter TABLE Students Drop column Field

تعریف شاخص (index) روی یک جدول

```
CREATE Index "index_name"  
ON Table_Name( "column 1", "column 2" , ... )
```

این دستور یک جدول شاخص روی جدول مورد نظر ایجاد می کند که مقادیر آن جدول بر اساس فیلد یا فیلدهای مورد نظر مرتب شده است.

مثال:

```
CREATE Index SN on Students(sname)  
CREATE Index SNF on Students(sname, sfamily)
```

شاخص به سرعت عملیات جستجو کمک می کند. مثلا دستور اول باعث می شود اگر دانشجویی با یک نام خاص را بخواهیم پیدا کنیم، با استفاده از شاخص SN سریعا آن را بیابیم.

حذف شاخص:

```
DROP Index "index_name"
```

مثال: حذف شاخص ایجاد شده روی ستون name از جدول Students:

Drop Index SN

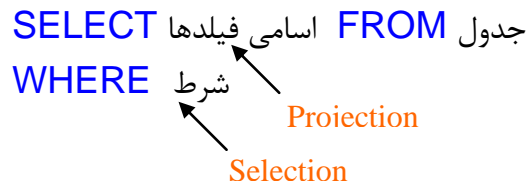
دستورات کار با داده ها(DML)

۱- دستور Select

دستور Select مهمترین و متنوعترین دستور SQL است که کار آن بازیابی داده ها از جداول بانک اطلاعات است. ساختار کلی یک دستور ساده Select بصورت زیر است:

ر ا ب S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry



مثال: لیست شماره و نام تمام دانشجویان.

Select Id, Name from S

مثال: لیست مشخصات تمام دانشجویان رشته های کامپیوتر و شیمی.

Select * from S where Field='Computer' or Field= 'Chemistry'

ممکن بود در این مثال بدلیل کم توجهی به اشتباه بجای عملگر OR بین دو شرط از and استفاده می شد. در آن صورت، مثل این است که بدنبال دانشجویی هستیم که رشته او هم کامپیوتر و هم شیمی است!

مثال: لیست اسامی دانشجویان کامپیوتر که شماره آنها کمتر از ۸۴۱۱۰ است.

Select Name from S where Field='Computer' and Id<84110

مقادیر ممکن بجای اسامی فیلدها:

دستور Select دستوری انعطاف پذیر است. در این دستور غ یر از اسامی فیلدهای جدول جایگزینهای دیگری نیز می توان داشت که برخی از آنها عبارتند از:
۱. یک عدد ثابت:

Select Name, 5 from S

Name	Unknown Column
Ali	5
Reza	5
Hassan	5

۲. یک فرمول:

Select Name, Id/10 as NewID from S

Name	NewID
Ali	8411
Reza	8412
Hassan	8413

۳. ترکیب چند فیلد رشته ای:

مثال: لیست نام و نام خ دانشجویان.

Select Name + ' ' + Family as CompleteName from S

CompleteName
Ali Ahmadi
Reza Rezaei
Hassan Hassani

بررسی خالی بودن یک فیلد

ر ا ب س

SELECT اسامی فیلد ها FROM

جدول

WHERE فیلد IS NULL

Id	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Null
84130	Hassan	Hasani	Chemistry

مثال: لیست شماره و رشته دانشجویانی که رشته آنها وارد شده است.

Select Id,Field from S where not field Is NULL

~~=NULL~~

مثال: لیست شماره دانشجویانی که نه رشته و نه نام خ آنها وارد نشده است.

Select Id from S where Family Is Null and Field Is Null

عملگر Order By :

برای مرتب کردن تاپلهای خروجی بر اساس ستون یا ستونهای مورد نظر می توان از عملگر Order By استفاده کرد.

ر ا ب س

جدول FROM اسامی فیلدها
WHERE شرط
ORDER BY نام یا اندیس فیلدها [desc | asc]
نزولی
صعودی

Id	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

مثال: لیست شماره و نام خ تمام دانشجویان، مرتب شده براساس شماره بطور صعودی

Select Id , family from S order by Id

مثال: لیست مشخصات تمام دانشجویان مرتب بر اساس نام خ و سپس نام.

Select * from S order by family , name

مثال: لیست اسامی دانشجویان کامپیوتر بترتیب نزولی شماره آنها

Select Name from S where Field ='Computer' order by 0 desc

عملگر Between

این عملگر برای بررسی اینکه مقدار یک ستون در یک بازه (بین دو مقدار خاص) است یا نه بکار می رود و برای نوعهای عددی و تاریخ قابل استفاده است. لازم بذکر است که خود دو مقدار ابتدا و انتهای بازه هم در شرط در نظر گرفته می شوند.

ر ا ب س

Id	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

مثال: لیست نام خ تمام دانشجویان که شماره آنها بین ۸۴۱۱۰ و ۸۴۱۲۰ است.

Select Id , Name form S where Id >=84110 and Id <=84120

یا

Select Id , Name form S where Id between 844110 and 84120

عملگر Like

عملگر Like برای مقادیر رشته ای است و برای یافتن مقادیری است که نه الزاما دقیقا یکسان بلکه مشابه کلمه مورد نظر باشند. منظور از مشابه بودن این است که کلمه مورد جستجو جزئی از آن کلمه باشد. قابلیت‌های این عملگر و نحوه استفاده از آن با چند مثال روی جدول زیر نشان داده می شود

ر ا ب S

Id	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

مثال: لیست دانشجویانی که نام آنها با A شروع می شود.

Select * from S where name like 'A%'

مثال: لیست دانشجویانی که نام خ آنها با i تمام می شود.

Select * from S where name Like '%i'

مثال: لیست دانشجویانی که در نام آنها حرف A یا a وجود دارد.

Select * from S where name Like '%A%' or name Like '%a%'

نکته: اگر علامت % را فراموش کنیم، عملگر Like معادل = عمل می کند.

مثال: لیست مشخصات تمام دانشجویان رشته کامپیوتر که نام آنها ۳ حرفی و حرف میانی آن I است.

Select * from S where Field='Computer' and name Like 'I_'

مثال: لیست مشخصات تمام دانشجویان که نام آنها با A شروع و به i ختم می شود .

Select * from S where name Like 'A%i'

مثال: لیست اسامی دانشجویانی که حرف دوم و چهارم نام آنها a و s است.

Select name from S where name like '_a_s%'

بازیابی رکوردهای تکراری

گاهی ممکن است خروجی یک دستور Select شامل رکوردهای تکراری باشد. اگر نخواهیم رکورد تکراری بازیابی شود باید از عملگر DISTINCT استفاده کنیم. عملگر دیگر ALL است که برعکس DISTINCT عمل می کند. اگر هیچکدام از آنها نوشته نشود، بطور پیش فرض ALL در نظر گرفته می شود.

اسامی [DISTINCT | ALL]
فیلدها
FROM .

ر ا ب S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

مثال: لیستی از تمام رشته های دانشگاه.

Select Distinct Field from S

مثال: لیست نام و نام خ تمام دانشجویان.

Select all Name , Family from S یا

Select Name , Family from S

عملگر UNION و UNION ALL

عملگر Union اجتماع رکوردهای دو جدول را برمی گرداند. مطابق اجتماع مجموعه ها، رکوردهای تکراری در صورت وجود تنها یکبار در خروجی می آیند. اما اگر مایل باشیم رکوردهای تکراری حذف نشوند، می توانیم از

عملگر

Union

All استفاده کنیم.

S				T		
<u>Id</u>	Name	Family	Field	<u>TId</u>	TName	TFamily
84110	Ali	Ahmadi	Computer	1	Hadi	Ahmadi
84120	Reza	Rezaei	Math	2	Karim	Hassani
84130	Hassan	Hasani	Chemistry	3	Ali	Omidi

مثال: لیستی از نام خ تمام اساتید و دانشجویان.

Select Family from S UNION Select Family From T

Select Family from S UNION ALL Select Family From T

دستور اول دقیقاً اجتماع است یعنی رکورد های تکراری یکبار در خروجی می آیند. اما در دومی تکرار می شوند.

مثال: لیست مشخصات تمام دانشجویان رشته های کامپیوتر و شیمی.

Select * from S where Field='Computer' UNION Select * from S where Field='Chemistry'

عملگر IN

عضویت یک مقدار را در یک مجموعه بررسی می کند.

S				T			ر ا ب STC			
<u>Id</u>	Name	Family	Field	<u>TId</u>	TName	TFamily	<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	Ali	Ahmadi	Computer	1	Hadi	Ahmadi	84110	2	01	12
84120	Reza	Rezaei	Math	2	Karim	Hassani	84110	1	03	8
84130	Hassan	Hasani	Chemistry	3	Ali	Omidi	84130	2	02	18

حالت

(اول)

مجموعه ثابت و مشخص است.

مثال: لیست مشخصات تمام دانشجویان رشته های کامپیوتر، ریاضی و شیمی.

Select * from S where Field IN ('Computer' , 'Math' , 'Chemistry')

حالت دوم) مجموعه ثابت و مشخص نیست. (Nested Query)

مثال: لیست مشخصات تمام دانشجویان که تاکنون درس گرفته اند.

(باید Id آنها عضو مجموعه Sid های موجود در STC باشد)

Select * from S where Id IN (SELECT Sid form STC)

مجموع SID های موجود در STC

~~SELECT * from S where Id IN (select * from STC)~~

یک مثال دیگر

S				C			STC			
Id	Name	Family	Field	CId	CName	Units	SId	TId	CId	mark
84110	Ali	Ahmadi	Computer	01	DB	3	84110	2	01	12
84120	Reza	Rezaei	Math	02	OS	3	84110	1	03	8
84130	Hassan	Hasani	Chemistry				84130	2	02	18

مثال:

لیست

مشخصات

ت

دانشجویانی که تاکنون درس ۳ واحدی نگرفته اند.

(باید Id آنها عضو مجموعه Sid های موجود در STC که درس ۳ واحدی گرفتند نباشد)

SELECT * from S where Id NOT IN
(SELECT Sid from STC where CID IN
(SELECT Cid from c where Units=3))

ر ا ب

رابطه S

رابطه P

رابطه SP

s#	SName	Status	City
S1	فن آوران	30	تهران
S2	پولادین	30	تبریز

P#	PName	Color	Weight	City
P1	AA	Red	10	تهران
P2	BB	Green	20	شیراز
P3	CC	Red	15	تهران

S#	P#	Qty
S1	P1	300
S1	P2	200
S2	P2	400
S2	P3	100

مثال: لیست تهیه کنندگانی که قطعه p2 را تولید کرده اند.

(باید s# آنها عضو مجموعه S# های موجود در SP که P# آنها P2 است، باشد)

**SELECT * from S where S# IN
(SELECT S# from SP Where P# = 2)**

استفاده از '=' بجای 'IN'

مثال: شماره دانشجویانی که درس OS را گرفته اند.

برای پاسخ به این سوال می توان نوشت:

**SELECT Sid from STC Where CId IN (SELECT CId from C Where
CName='OS')**

اما اگر مطمئن باشیم که تنها یک درس بنام OS در جدول C داریم، می توان نوشت:

SELECT Sid from STC Where CId = (SELECT CId from C Where CName='OS')

دستور فوق تنها در صورتی بدرستی اجرا می شود که حاصل دستور داخل پرانتز تنها یک مقدار باشد. در غیر

اینصورت در کامپایل دستور مشکلی ایجاد نمی شود، اما خطای زمان اجرا رخ می دهد

عملگرهای Any و All

استفاده از عملگرهای ANY (حداقل یکی) و ALL (همه) در شرطها با چند مثال نشان داده می شود:

مثال: لیست تهیه کنندگانی که وضعیت آنها عددی فرد بین ۲۸ تا ۳۴ است.

Select * from S Where status= 29 or status = 31 or status=33

یا

Select * from S Where status = ANY (29 , 31 , 33)

مثال: نام قطعاتی که وزن آنها بین تمام قطعات مینیمم نباشد.

Select * from P Where weight > ANY (Select weight from P)

مثال: نام قطعاتی که وزن آنها بین تمام قطعات مینیمم باشد.

Select * from P Where weight <= All (Select weight from P)

نکته: شرط 'ANY(..., ..., ...)' معادل است با شرط 'IN(..., ..., ...)'

عملگر Exists

شرط Exists (Select * from ...) مقدارش true یا false است. اگر نتیجه select داخل پرانتز رکوردی داشته باشد، حاصل true وگرنه false می شود.

کاربرد اصلی: یک رکورد از یک جدول با این شرط انتخاب می شوند که در جدول دیگری رکوردی موجود باشد که با این رکورد ارتباط داشته باشد. بنابراین، بازای هر رکورد از جدول اول، یکبار تمام رکوردهای جدول دوم بررسی می شود.

هر پرس و جو که با In جواب داده شود با Exists نیز قابل جواب است اما برعکس صحیح نیست.

مثال: لیست تمام دانشجویانی که تاکنون درس گرفته اند.

روش اول:

Select * from S where SId IN (Select SId from STC)

روش دوم:

Select * from S Where Exists (Select * from STC where STC.SId = S.SId)

مقایسه دو راه حل (با این فرض که جدول S شامل n تاپل و STC شامل m تاپل باشد):

در روش اول، ابتدا دستور داخل پرانتز اجرا می شود و مجموعه شماره دانشجویانی که درس گرفته اند از جدول STC استخراج شده و جایگزین پرانتز می شود. سپس Select اصلی اجرا می شود و هر دانشجویی که شماره اش جزء مجموعه باشد انتخاب می شود. پس اگر جدول S شامل n تاپل و STC شامل m تاپل باشد، چون تاپلهای هر کدام از این دو جدول تنها یکبار خوانده می شود، مرتبه اجرایی بازیابی اطلاعات در روش اول $O(m+n)$ است.

اما در روش دوم کار با دستور Select اصلی (خواندن تاپلهای رابطه S) آغاز می شود. هر رکوردی از S که خوانده می شود، به این شرط انتخاب می شود که در جدول STC رکوردی مرتبط با آن وجود داشته باشد. پس

بازای هر رکورد از S، یکبار کل رکوردهای STC خوانده می شوند (n بار دستور داخل پرانتز اجرا می شود). بنابراین مرتبه اجرایی در این حالت O(m.n) می شود. نتیجه اینکه عملگر IN از Exists سریعتر است و در موارد ممکن بهتر است بجای Exists از IN استفاده کرد.

مثال: لیست تولیدکنندگان قطعه P2 (با استفاده از عملگر Exists)

راه حل: از جدول S هر رکوردی که شرط زیر را داشته باشد انتخاب کن:

“در جدول SP رکوردی موجود باشد که S# آن برابر با S# رکورد ما بوده و P# آن برابر P2 باشد”

Select * from S Where

Exists(Select * from SP Where S# = S.S# and P# = P2)

عملگر Unique:

کاربرد عملگر Unique مشابه Exists است، با این تفاوت که خروجی عملگر Unique در صورتی true است که حاصل دستور داخل پرانتز دقیقا یک رکورد باشد.

مثال: لیست تهیه کنندگاری که از قطعه P2 یک نمونه تولید کرده اند.

Select * from S Where

Unique (Select * from SP Where S# = S.S# and P# = P2)

پیوندها در SQL

با توجه به اینکه با مفهوم انواع پیوندها در بخش جبر رابطه ای آشنا شده اید، بدون توضیح اضافی نحوه نوشتن انواع پیوندها در SQL را با مثال نشان می دهیم.

ضرب دکارتی

ر ا ب S

Id	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Null
84130	Hassan	Hasani	Chemistry

ر ا ب STC (جدول)

SId	TId	CId	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال:

Select * from S , STC

یا

Select * from S Cross Join STC

می دانیم که در این نوع پیوند، همه رکوردها از دو جدول با هم ترکیب می شوند. بنابراین حاصل ضرب $S \times STC$ شامل رکوردهای بی معنی خواهد بود. پس عملاً کاربرد چندانی ندارد.

پیوند شرطی و طبیعی

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Null
84130	Hassan	Hasani	Chemistry

رابطه STC (جدول)

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

روش اول:

Select * from S , STC where S.ID = STC.SId

شرط اصلی پیوند: به ابري مقدار فيلد هاي مرتبط با هم

روش دوم:

Select * from S Join STC On S.Id = STC.SId

در این نوع پیوند، رکوردهایی از دو جدول با هم ترکیب می شوند که منطقیاً بهم ربط دارند. کار دو روش پیوند نوشته شده یکی است، اما دستور دوم بهینه تر است. شرطی که بجای Where جلوی ON نوشته می شود، شرط پیوند است یعنی در هنگام پیوند کنترل می شود، نه بعد از پیوند. معادل دو دستور فوق در جبر رابطه ای بترتیب بصورت زیر است:
دستور اول:

σ S.Id=STC.SId (S × STC)

دستور دوم:

S ×
S.Id=STC.SId STC

S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Null
84130	Hassan	Hasani	Chemistry

رابطه STC

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال:

نکته: برای پاسخ به پرس و جوهایی که به بیش از یک جدول نیاز دارند از دو روش می توان استفاده کرد:

Join (۲) Nested Query (۱)

مثال: بازیابی لیست دانشجویانی که تاکنون درس گرفته اند

1. SELECT S.* FROM S WHERE Id IN (SELECT SId FROM STC)

2. SELECT S.* FROM S, STC WHERE S.Id = STC.SId

نکته: استفاده از پیوند سبب سرعت بالاتر می شود، ام استفاده از پرس و جوهای تودرتو خوانایی بیشتری دارد.

چند مثال:

رابطه S

رابطه T

رابطه C

جدول

رابطه STC

<u>SId</u>	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

<u>TId</u>	TName	TFamily
1	Hadi	Harnidi
2	Karim	Hassani
3	Ali	Omidi

<u>CId</u>	CName	Units
01	DB	3
02	OS	3

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: لیست دانشجویان به همراه نمرات آنها.

Select S.* , mark from **S** , **STC** Where S.Id = STC.Sid

مثال: لیست دانشجویان به همراه مشخصات دروس اخذ شده و اساتید مربوطه و نمرات آنها.

Select * from **S** , **C** , **T** , **STC** Where STC.Sid = S.Id
AND STC.Cid = C.Id
AND STC.Tid = T.Id

مشخص است که برای پاسخ به پرس و جوهایی که نیاز به پیوند چند جدول داریم، کافی است اسامی آن جدولها را در دستور Select نوشته و شرطهای اصلی پیوند (برابری ستونهای مرتبط) ذکر شوند.

مثال: لیست تمام دانشجویانی که درس OS را گذرانده اند.

- 1) Select S.* from **S** , **C** , **STC** Where S.Id = STC.Sid AND C.Id = STC.Cid
AND CName = 'OS'
- 2) Select S.* from **S Join STC ON S.Id = STC.Sid**
Join C ON C.Id = STC.Cid
Where CName = 'OS'

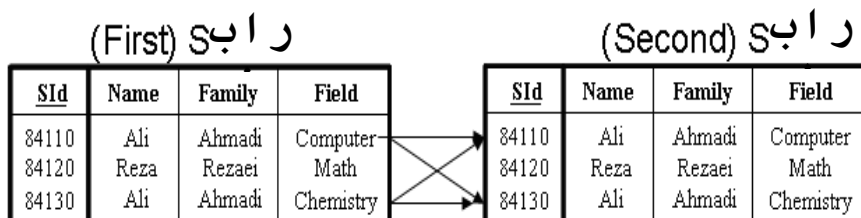
مثال: لیست اساتیدی که در درس DB دانشجوی افتاده دارند.

Select T.* from **C** , **T** , **STC** Where STC.Cid = C.Id
AND STC.Tid = T.Id
AND Cname = 'DB' AND mark < 10

چند مثال:

مثال: لیست شماره دانشجویی تمام جفت دانشجویانی که نام و نام خ یکسان دارند.

نکته: زمانی که نیاز داریم که برای یک پرس و جو از پیوند یک جدول با خودش استفاده کنیم، باید ۲ کپی از آن در حافظه ایجاد کنیم. به این عمل Aliasing (نام مستعار) گفته می شود.



Select First.Id , Second.Id From **S First , S Second**
 Where First.Name = Second.Name AND First.Family = Second.Family
 AND First.Id < Second.Id

اگر شرط آخر فراموش شود، چند مشکل بوجود می آید. اول اینکه، هر تاپل با خودش هم پیوند می خورد . دوم اینکه علاوه بر اینکه مثلا تاپل اول از جدول اول با تاپل سوم از جدول دوم (بدلیل یکسان بودن نام دانشجو) پیوند می خورند، تاپل سوم از جدول اول با تاپل اول از جدول دوم هم پیوند می خورند که در واقع یک پیوند تکراری انجام شده است.

فراپیوند

رابطه

رابطه

رابطه

رابطه STC (جدول)

SId	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

TId	TName	TFamily
1	Hadi	Hamidi
2	Karim	Hassani
3	Ali	Omidi

CId	CName	Units
01	DB	3
02	OS	3

SId	TId	CId	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: لیست تمام دانشجویان به همراه نمرات آنها. اگر دانشجویی درسی نگرفته باز در لیست باشد.

Select * from S Left Outer Join STC On S.Id = STC.Sid

توابع ستونی (آماری)

این توابع عملی را روی یک ستون از جدول انجام می دهند و حاصل یک عدد می شود.

- **SUM**: مجموع مقادیر یک ستون از نوع عددی
- **AVG**: میانگین مقادیر یک ستون از نوع عددی
- **Min**: مینیموم مقادیر یک ستون از نوع عددی
- **Max**: ماکزیموم مقادیر یک ستون از نوع عددی
- **Count**: تعداد عناصر یک ستون (تعداد رکوردها)
 - یا Count(FieldName)
 - Count(*): تعداد رکوردها

چند مثال:

رابطه

رابطه

رابطه

رابطه STC (جدول)

SId	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

TId	TName	TFamily
1	Hadi	Hamidi
2	Karim	Hassani
3	Ali	Omidi

CId	CName	Units
01	DB	3
02	OS	3

SId	TId	CId	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: تعداد دانشجویان رشته کامپیوتر

Select Count (Sid) as Tedad from S Where Field = 'Computer'

یا

Select Count (*) as Tedad from S Where Field = 'Computer'

مثال: تعداد دانشجویانی که تاکنون نمره کمتر از ۱۰ دارند.

Select Count (Distinct Sid) as Tedad from STC Where mark < 10

استفاده از کلمه کلیدی Distinct در داخل تابع Count موجب می شود مقادیر تکراری تنها یکبار شمارش شوند.

مثال: ماکزیموم و مینیموم نمره بین دانشجویان رشته کامپیوتر

Select Max(mark) as mx , Min(mark) as mn from S,STC Where S.Sid = STC.Sid AND Field = 'Computer'

mx	mn
18	8

چند مثال:

رابطه

رابطه

رابطه

رابطه STC (جدول)

SId	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

TId	TName	TFamily
1	Hadi	Hamidi
2	Karim	Hassani
3	Ali	Omidi

CId	CName	Units
01	DB	3
02	OS	3

SId	TId	CId	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: تعداد اساتیدی که تاکنون درس OS را ارائه کرده اند.

<u>CId</u>	Cname	Units	<u>SId</u>	<u>TId</u>	mark
01	DB	3	84110	2	12
02	OS	3	84110	1	8
02	OS	3	84130	1	18

Select Count (**Distinct tId**) as Tedad from C,STC Where C.CId = STC.CId
AND Cname = 'OS'

چند مثال:

<u>SId</u>	SName	SFamily	Field	<u>TId</u>	TName	TFamily	<u>CId</u>	CName	Units	<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	Ali	Ahmadi	Computer	1	Hadi	Hamidi	01	DB	3	84110	2	01	12
84120	Reza	Rezaei	Math	2	Karim	Hassani	02	OS	3	84110	1	03	8
84130	Hassan	Hasani	Chemistry	3	Ali	Omidi				84130	2	02	18

مثال: تعداد اساتیدی که تاکنون درس OS را ارائه نکرده اند.

باید Id آنها در این مجموعه نباشد

<u>CId</u>	Cname	Units	<u>SId</u>	<u>TId</u>	mark
01	DB	3	84110	2	12
02	OS	3	84110	1	8
02	OS	3	84130	1	18

Select Count (tId) from T Where tId **NOT IN**
(Select tId from C,STC Where C.CId = STC.CId AND Cname = 'OS')

مثال: شماره دانشجویانی که نمره ای بالاتر از میانگین تمام نمرات دارند.

Select SId from STC
Where mark > (Select Avg(mark) from STC)

يك عدد
است

~~Select SId from STC
Where mark > Avg(mark)~~

دستور دوم اشتباه است. به این دلیل که دستور مستقلی برای محاسبه میانگین در ابتدا وجود ندارد. پس هر رکورد را که می خواند، میانگین کل از قبل محاسبه نشده تا مقدار نمره این رکورد با آن مقایسه شود. بنابراین مقدار نمره هر رکورد با میانگین نمرات از ابتدای جدول تا اینجا مقایسه می شود (نه میانگین کل نمرات). پس این دستور منجر به خطای منطقی می شود.

مثال: مشخصات دانشجویانی که نمره ای بالاتر از میانگین تمام نمرات دارند.

Select * from S where SId in (دستور بالا)

View ها : جداول مجازی

علاوه بر جداول اصلی (Table) نوع دیگری از جداول بنام جداول مجازی (View) وجود دارند. View ها برخلاف Table ها (که واقعا روی دیسک هستند)، وجود خارجی ندارند. برای کاربران بظاهر و از نظر کاربردی، View ها مانند جدولها هستند، پس می توانیم:

- آنها را باز کرده و داده های آنها را ببینیم.
- بر روی آنها عملیات جبر رابطه ای را انجام دهیم، مثلا یک دستور Select روی یک View بنویسیم یا آن را با یک جدول پیوند دهیم.

ساخت یک View

هرگاه یک دستور Select که جدول حاصل آن برای ما زیاد مورد استفاده است، میتوانیم آن را بصورت یک View ذخیره کنیم تا در مواقع لزوم از آن استفاده کنیم. هنگام ذخیره View داده ها ذخیره نمی شوند. بلکه خود دستور Select است که ذخیره می شود. هنگام رجوع به یک View خود سیستم سریعا دستور مربوط به آن را اجرا کرده و حاصل Select را به ما نشان می دهد بطوریکه بنظر می رسد این داده ها در داخل View ذخیره شده اند.

دستور ساخت یک View:

```
CREATE View viewname as  
Select ....
```

بعنوان مثال برای بسیاری از پرس و جوها نیاز داریم ۴ جدول فوق (یا تعدادی از آنها را پیوند طبیعی دهیم)، و بناچار Query های ما طولانی می شود.

رابطه S

SId	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

رابطه T

TId	TName	TFamily
1	Hadi	Hamidi
2	Karim	Hassani
3	Ali	Omidi

رابطه C

CId	CName	Units
01	DB	3
02	OS	3

جدول

رابطه STC

SId	TId	CId	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

کافی است حاصل پیوند طبیعی ۴ جدول را بصورت یک View ذخیره و در هنگام لزوم از آن استفاده کنیم:

Create View WholeData as

Select * From S,C,T,STC

Where S.Id=STC.Sid AND C.Id=STC.CId AND T.Id= STC.tId

مثال: نام اساتیدی که درس OS را تاکنون تدریس کرده اند.

Select tname from WholeData Where cname = 'OS'

ویرایش داده های یک View

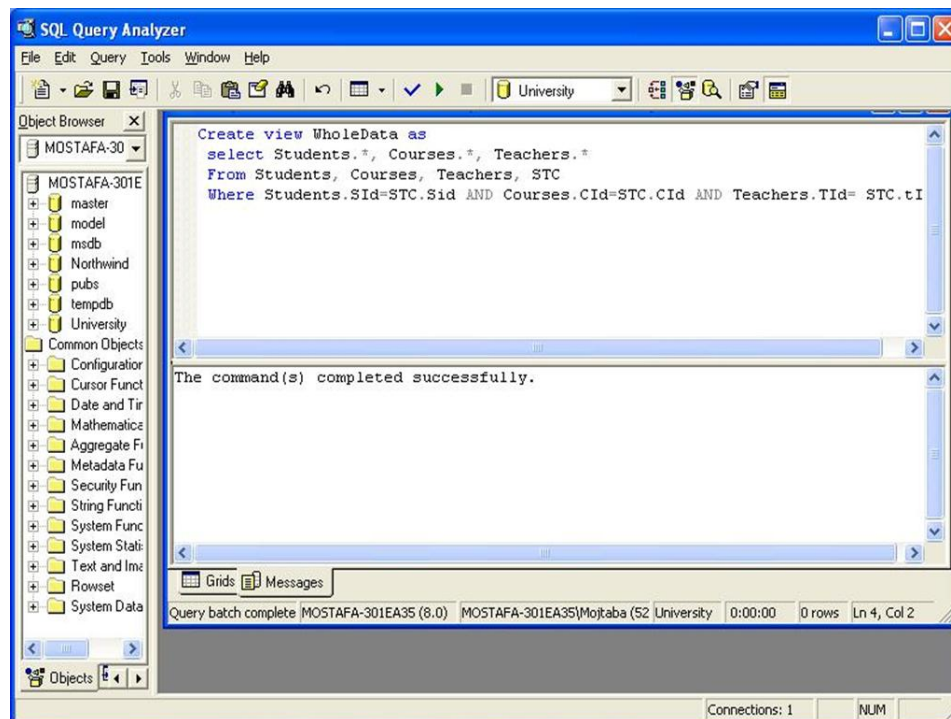
داده های view ها را می توان مانند table ها ویرایش کرد. ام در واقع جدول پایه view ویرایش می شود. درج و ویرایش در صورتی قابل اجراست که view شرایط خاصی داشته باشد. از جمله:

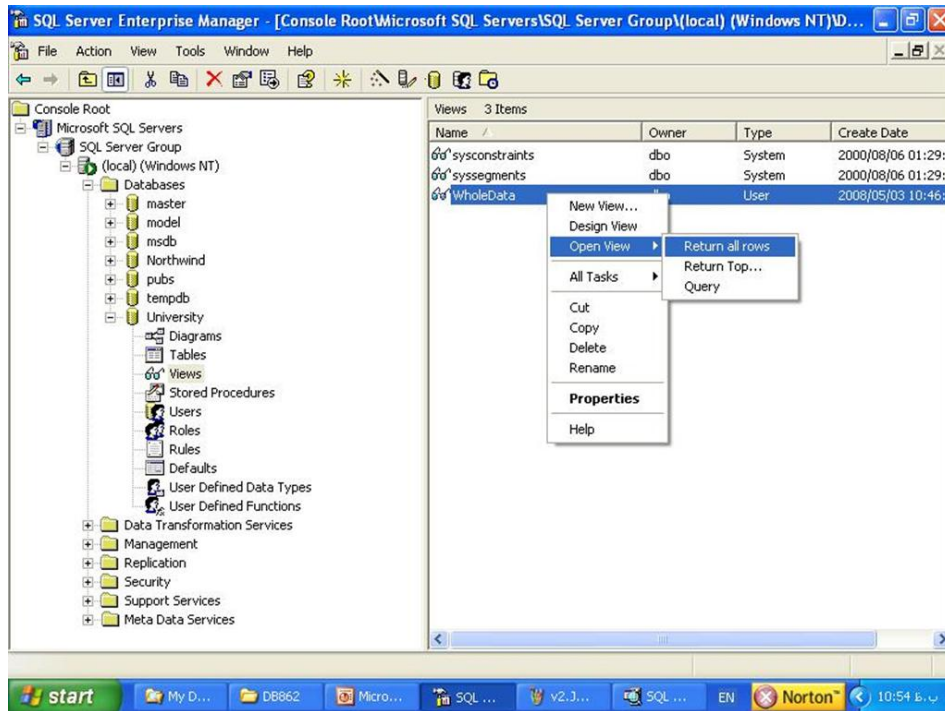
- فقط از روی یک جدول ساخته شده باشد
- کلید جدول در view هم باشد.
- در تعریف آن از select distinct استفاده نشده باشد.

مثال: در view زیر نمی توان رکورد جدیدی درج کرد. چون درج در جدول اصلی (S) انجام می شود و S# نمی تواند مقدار null بگیرد.

-Create view view1 as Select name , city from S

~~-'Ali' , 'Shiraz') → view1~~





The screenshot shows the data in the 'WholeData' table. The table has the following columns: SId, Name, family, CId, Cname, Units, TId, Tname, Tfamily, and Degree. The data is as follows:

SId	Name	family	CId	Cname	Units	TId	Tname	Tfamily	Degree
84110	Ali	Ahmadi	1	DB	3	102	Mohsen	Mohseni	PhD
84110	Ali	Ahmadi	3	C++	2	101	Ahmad	Ahmadi	PhD
84120	Reza	Rezaei	1	DB	3	102	Mohsen	Mohseni	PhD
84140	mahdi	Mahdavi	2	OS	3	103	Akbar	Akbari	Ms
84140	mahdi	Mahdavi	3	C++	2	102	Mohsen	Mohseni	PhD

Stored procedure

- دستور یا دنباله ای از دستورات SQL را می توانیم بصورت یک Procedure در DB ذخیره کنیم تا در زمان نیاز در برنامه های کاربردی بتوانیم آن را صدا بزنیم.

- هر پروسیجر یک نام و می تواند تعدادی پارامتر داشته باشد. همچنین داخل بدنه آن می توان متغیرهای مورد نیاز را تعریف کرد.

دستور ساخت یک SP:

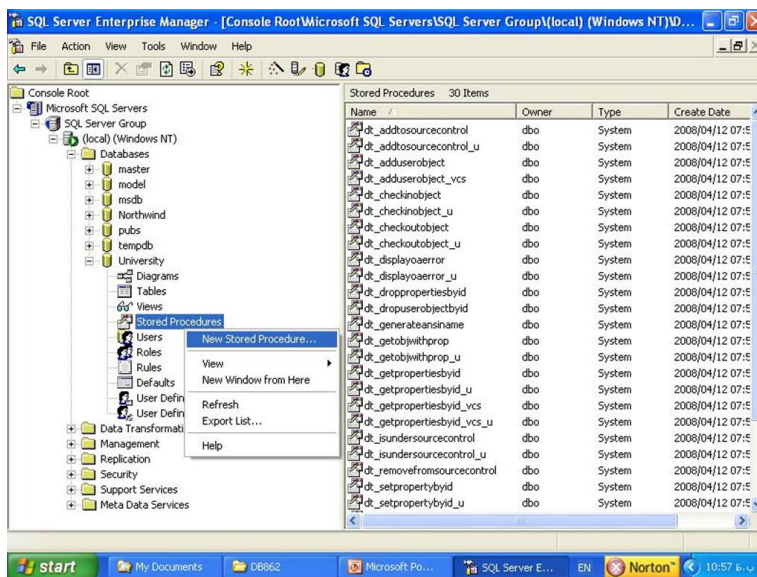
CREATE Procedure SPname

@param1 type1,

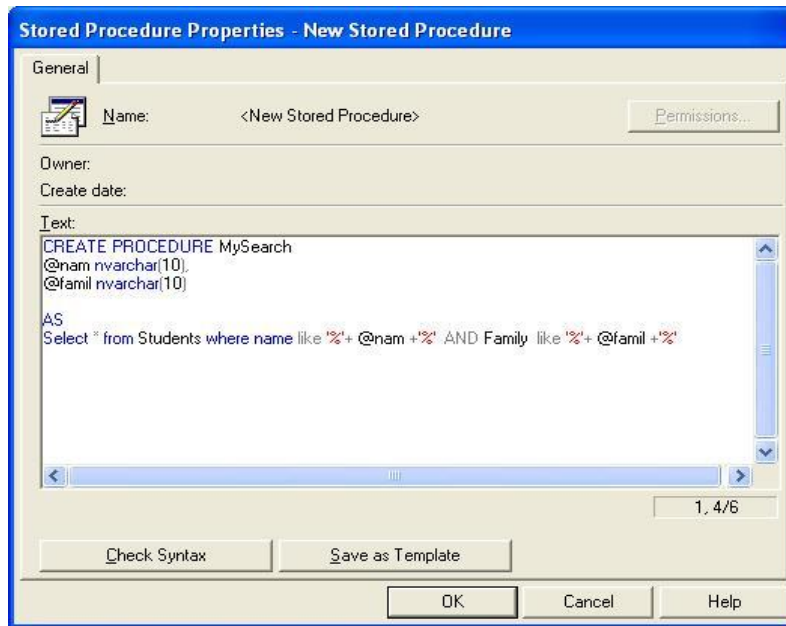
@param2 type2 , ...

as

بدنه پروسیجر



مثال: یک sp برای عمل جستجو



تست (نرم افزار دولتی ۸۶)

دستور SQL زیر در بانک اطلاعاتی تولید کنندگان و قطعات چه بر می گرداند ؟

Select S.SNAME

From S

Where S.S# IN (Select SP.S#

From SP Where SP.P# ='P2';)

- 1) اسامی تولید کنندگانی که قطعه شماره ' P2 ' را تولید می کنند (احتمالاً با تکرار)
- 2) اسامی تولید کنندگانی که قطعه شماره ' P2 ' را تولید می کنند (بدون تکرار)
- 3) اسامی تولید کنندگانی که فقط قطعه شماره ' P2 ' را تولید می کنند .
- 4) اسامی تولید کنندگانی که علاوه بر قطعه شماره ' P2 ' قطعه دیگری نیز تولید می کنند.

گزینه :.....

تست (نرم افزار دولتی ۸۵)

کدام یک از عبارات SQL زیر با بقیه عبارات معادل نمی باشد؟

Select P.* (۱)

from part P
where P.P# IN(select SP.P# from supply SP where SP.charge >40000)

(۲)

Select P.*
from part P
where P.P# = ALL(select S.P# from supply S where S.charge >40000)

(۳)

Select P.*
from part P
where Exists (select * from supply SP where P.P# =SP.P# and SP.charge >40000)

(۴)

Select P.*
from part P
where P.P# =(select SP.P# from supply SP where P.P#=SP.P# and SP.charge >40000)

گزینه :.....

تست (نرم افزار دولتی ۸۷)

نتیجه دستور SQL زیر چیست؟

```
SELECT P.*  
FROM S,P,SP  
WHERE S.CITY='LONDON'
```

AND S.S# =SP.S#

AND P.P# =SP.P#

- (۱) دستور دارای خطای نحوی است .
- (۲) کلیه مشخصات قطعاتی که توسط تمام تولید کنندگان لندن تولید می شود.
- (۳) کلیه مشخصات قطعاتی که حداقل توسط یکی از تولید کنندگان لندن تولید می شود.
- (۴) کلیه مشخصات قطعاتی که دقیقاً توسط یکی از تولید کنندگان لندن تولید می شود.

گزینه :.....

کدام گزینه در مورد دستور زیر صحیح است؟

SELECT * FROM Students WHERE Sid = (SELECT Sid FROM students)

- (۱) دستور صحیح است
- (۲) دستور خطای زمان ترجمه دارد
- (۳) دستور خطای زمان اجرا دارد
- (۴) دستور ممکن است خطای زمان اجرا داشته یا نداشته باشد

گزینه :.....

توضیح: اگر دستور داخل پرانتز یک مقدار برگرداند دستور اجرا می شود. اما اگر تعدادی رکورد برگرداند مقایسه با = بی معنی است.

کدام گزینه در مورد دستور زیر صحیح است؟

SELECT s# FROM S WHERE city = (SELECT city FROM S Where s# = S1)

۱. شماره تهیه کنندگان همشهری تهیه کننده S1
۲. شماره تهیه کنندگان همشهری تهیه کننده S1 و شماره خود S1
۳. این طرز نوشتن در SQL اصلاً مجاز نیست
۴. دستور ممکن است خطای زمان اجرا داشته یا نداشته باشد

گزینه :.....

توضیح: چون مطمئناً دستور داخل پرانتز فقط یک مقدار برمی گرداند دستور بدرستی اجرا می شود

کدام دستور معادل دستورات زیر است؟

Create view V as SELECT * FROM S where city = 'London'

Select Name from V where s# in (Select s# from SP WHERE Qty > 4000)

1. Select Name from S,SP where S.S# = SP.S# AND city = 'London' And Qty>4000
2. Select Name from S join SP on S.S# = SP.S# where city = 'London' And Qty>4000
3. Select Name from S where S# in (select S# from SP Where Qty>4000) AND city = 'London'
4. همه موارد

گزینه:

Select کردن از حاصل یک Select

ر ا ب S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

می دانیم که حاصل دستور Select یک جدول است. پس می توان از آن مجددا اطلاعاتی را select کرد. این کار به دو روش قابل انجام است:

روش اول) ذخیره نتیجه select اول در قالب یک view و بعد Select کردن از View

Create view tmp as

Select * from S Where Field='Computer'

tmp

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84130	Hassan	Hasani	Computer

Select Id from tmp where name like 'A%'

روش دوم) مستقیم و بدون استفاده از view

Select Id from (select * from S where Field = 'Computer') tmp where

نام جدول حاصل را tmp
می گذارد
(Aliasing)

Tmp.name like 'A%'

زمانی که با دستور پیچیده ای مواجه هستیم می توانیم آنرا به این روش به دستورات ساده تر بشکنیم؛ یعنی طی چند مرحله Select را اعمال کنیم.

Select TOP(n)

زمانی که بخواهیم از لیست حاصل از Select فقط n رکورد اول را نشان دهد.

مثال: لیست ۱۰ نفر اول لیست دانشجویان

Select Top(10) * from S

مثال: لیست ۱۰ نفر اول لیست دانشجویان کامپیوتر

Select Top(10) * from S Where Field = 'Computer'

مثال: لیست ۱۰ نفر اول از لیست دانشجویانی که تاکنون درس نگرفته اند.

Select Top(10) * from S where Sid Not in (Select Sid from STC)

مثال: لیست ۱۰ نفر سوم (۲۱ تا ۳۰ ام) از لیست دانشجویان.

(یعنی اولین ۱۰ نفری که Id آنها جزء ۲۰ شماره اول لیست نیست)

Select Top(10) * from S where Sid Not in
(Select Top(20) Sid from S)

20 شماره اول
لیست

عملگر Group By

- زمانی کاربرد دارد که بخواهیم رکوردها را بر اساس مقدار یک یا چند فیلد گروه بندی کنیم
- بعد از اعمال این عملگر، رکورد های جدول حاصل، هر یک نماینده یک گروه از رکوردها هستند.
- بیشتر اوقات زمانی استفاده می شود که بخواهیم توابع ستونی (آماری) را روی دسته های رکوردها بصورت مجزا اعمال کنیم.
- فقط عملگر Order By می تواند بعد از Group By بیاید.

Id	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Null
84130	Hassan	Hasani	Chemistry

SId	TId	CId	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: لیست شماره دانشجویی هر دانشجو به همراه میانگین نمرات او.
در این مثال بجای اعمال تابع Avg روی کل رکوردها باید بصورت زیر عمل شود:

SId	CId	term	mark
84110	01	2	12
84110	02	1	8
84130	02	2	18
84130	01	2	5

→ Avg →

SId	average
84110	10
84130	11.5

رکورد در واقع نماینده یک گروه از رکوردها است

Select SId,Avg(mark) as average from STC Group By SId

SId	CId	term	mark
84110	01	2	12
84110	02	1	8
84130	02	2	18
84130	01	2	5

→ Avg

→ Avg

SId	average
84110	10
84130	11.5

سوال: آیا در مثال قبل می توانستیم دستور زیر را بنویسیم؟

Select SId,term,Avg(Mark) from STC Group By SId

پاسخ: خیر، چون گروه بندی فقط بر اساس شماره دانشجویی انجام شده، میانگین بدست آمده بین نمرات مربوط به ترمهای مختلف است. پس ترم مشخصی وجود ندارد که آنرا در رکورد نماینده گروه قرار دهیم

SId	term	average
84110	?	10
84130	?	11.5

سوال: دستور زیر را چگونه بنویسیم؟

Select SId,term,Avg(Mark) from STC Group By SId

پاسخ: هدف از این دستور ظاهرا این است که شماره دانشجو و معدل او در هر ترم جداگانه نشان داده شود. بنابراین باید گروه بندی رکوردها را بر اساس فیلدهای (ش دانشجویی و ترم) انجام دهیم (نه فقط ش دانشجویی) تا معدل هر دانشجو در هر ترم مجزا حساب شود:

SId	CId	term	mark
84110	01	2	12
84110	02	1	8
84130	02	2	18
84130	01	2	5

→ Avg

→ Avg

→ Avg

SId	term	average
84110	2	12
84110	1	8
84130	2	11.5

Select SId,term,Avg(Mark) from STC Group By SId, term

نتیجه مهم: تنها فیلدهایی را می توانیم select کنیم که جلوی Group By آمده باشند. اما اجباری نیست هر فیلدی که در Group by آمده حتما جلوی Select هم بیاید.

چند مثال:

رابط S

رابط T

رابط C

جدول

رابط STC

SId	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

TId	TName	TFamily
1	Hadi	Harnidi
2	Karim	Hassani
3	Ali	Omidi

CId	CName	Units
01	DB	3
02	OS	3

SId	TId	CId	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: شماره تمام دروس به همراه ماکزیموم نمره ای که در هر درس تاکنون گرفته شده است.

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	02	8
84130	2	02	18

→ Max
→ Max

Select CId , Max(mark) from STC Group By CId

چند مثال:

مثال: مشخصات هریک از اساتید بترتیب الفبای نام خ، به همراه تعداد دانشجویانی که هر استاد در هر درس

داشته است.
(جدول)

<u>SId</u>	<u>TId</u>	TName	TFamily	<u>CId</u>	mark
84110	2	Karim	Hassani	01	12
84110	1	Hadi	Hamidi	02	8
84130	1	Hadi	Hamidi	02	18
84110	1	Hadi	Hamidi	01	15

→ Count
→ Count
→ Count

Select tId, tname, tfamily , Count(SId) from T, STC
Where T.tId = STC.tId
Group By tId, tname, tfamily, CId
Order by tfamily

Having: اعمال شرط روی گروهها

- همانطور که دیدید Group By رکوردها را دسته بندی می کند و در جدول حاصل هر رکورد در واقع یک گروه است.
- برای اینکه شرط بگذاریم که چه گروههایی در نهایت نمایش داده شوند، بجای Where از Having استفاده می کنیم.

مثال: بازیابی شماره دانشجویی و میانگین نمرات دانشجویانیکه میانگین نمرات آنها کمتر از ۱۱ است.

<u>SId</u>	<u>CId</u>	term	mark
84110	01	2	12
84110	02	1	8
84130	02	2	18
84130	01	2	5

→ Avg
→ Avg

<u>SId</u>	average
84110	10
84130	11.5

Select SId,Avg(Mark) from STC Group By SId

S				T			C			STC			
<u>SId</u>	SName	SFamily	Field	<u>TId</u>	TName	TFamily	<u>CId</u>	CName	Units	<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	Ali	Ahmadi	Computer	1	Hadi	Hamidi	01	DB	3	84110	2	01	12
84120	Reza	Rezaei	Math	2	Karim	Hassani	02	OS	3	84110	1	02	8
84130	Hassan	Hasani	Chemistry	3	Ali	Omidi				84130	1	02	18
										84110	1	01	15

Having Avg(Mark)<11

اگر بجای Having در این مثال Where نوشته می شد، لازم بود هر رکورد ستونی بنام Avg(mark) داشته باشد تا شرط برای آن چک شود که اینطور نیست.

مثال: بازیابی شماره دانشجویی و میانگین نمرات دانشجویانی که تا کنون ۳ تا از دروس را گرفته اند.

SId	CId	term	mark
84110	01	2	12
84110	02	1	8
84130	02	2	18
84130	01	2	5
84130	03	2	16

→ Avg

→ Avg

SId	average
84110	10
84130	11.5

Select SId,Avg(Mark) from STC Group By SId Having Count(distinct CId)>1

مثال: شماره هریک از اساتید که نام خ آنها با H شروع می شود و تاکنون بیش از یک دانشجو داشته اند، به همراه تعداد دانشجویانی که هر استاد ^{جدول} داشته است.

SId	TId	TName	TFamily	CId	mark
84110	2	Karim	Hassani	01	12
84110	1	Hadi	Hamidi	02	8
84130	1	Hadi	Hamidi	02	18
84110	1	Hadi	Hamidi	01	15

→ Count

→ Count

شرط اول به
where مربوط
می شود و شرط
having به

Select tId, Count(distinct Sid) from T, STC Where T.tId = STC.tId and tname like 'A%' Group By tId Having Count(distinct Sid) > 1

چند مثال:

رابط S				رابط T			رابط C			رابط STC (جدول)			
SId	SName	SFamily	Field	TId	TName	TFamily	CId	CName	Units	SId	TId	CId	mark
84110	Ali	Ahmadi	Computer	1	Hadi	Hamidi	01	DB	3	84110	2	01	12
84120	Reza	Rezaei	Math	2	Karim	Hassani	02	OS	3	84110	1	03	8
84130	Hassan	Hasani	Chemistry	3	Ali	Omidi				84130	2	02	18

مثال: شماره تمام دانشجویانی که تاکنون ۳ تا از دروس را گرفته اند.

Select Sid from STC Group By Sid
Having Count(distinct CId) = 3

پرس و جوهای دارای شرط 'همه':

در جبر رابطه ای، اینگونه پرس و جوها را با عملگر تقسیم پاسخ می دهیم. اما در SQL عملگر تقسیم نداریم. بجای آن سه روش برای پاسخگویی به این نوع پرس و جوها وجود دارد.

روش اول پاسخ به پرس و جوهای دارای "همه" (بجای عملگر تقسیم)

مثال: شماره تمام دانشجویانی که تاکنون همه دروس را گرفته اند
با الگوبرداری از مثال قبل:

Select Sid from STC Group By Sid
Having Count(distinct CId) = (select Count (CId) from C)

یک عدد است
در SQL عملگر تقسیم نداریم. پس به این روش
عمل می کنیم.

روش دوم پاسخ به پرس و جوهای دارای "همه"

رابطه S

رابطه T

رابطه C

جدول

رابطه STC

SId	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

TId	TName	TFamily
1	Hadi	Hamidi
2	Karim	Hassani
3	Ali	Omidi

CId	CName	Units
01	DB	3
02	OS	3

SId	TId	CId	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: شماره تمام دانشجویانی که تاکنون همه دروس را گرفته اند

جواب: هر دانشجویی که بازای آن دانشجو درسی وجود نداشته باشد که به همراه آن دانشجو در جدول STC

نیامده باشد.

Select Sid from S Where Not Exists
 (Select * from C Where Not Exists
 (Select * from STC
 where S.Id = Stc.Sid And C.Id = STC.CId))

S				T			C			STC			
SId	SName	SFamily	Field	TId	TName	TFamily	CId	CName	Units	SId	TId	CId	mark
84110	Ali	Ahmadi	Computer	1	Hadi	Hamidi	01	DB	3	84110	2	01	12
84120	Reza	Rezaei	Math	2	Karim	Hassani	02	OS	3	84110	1	03	8
84130	Hassan	Hasani	Chemistry	3	Ali	Omidi				84130	2	02	18

چند مثال پیشرفته تر از Select های تودرتو:

مثال: لیست کد هریک از اساتید به همراه ماکزیموم تعداد دانشجویانی که آن استاد تاکنون در یک ترم داشته است.

تفاوت این مثال با مثالهای قبل این است که بدنبال محاسبه تابع ماکزیموم روی مقادیر یک ستون جدول نیستیم. بلکه بایستی ابتدا تعداد دانشجویان هر استاد در هر ترم محاسبه شود و در یک جدول موقتی (مثلا tmp) قرار گیرد. سپس، ماکزیموم این تعدادها برای هر استاد محاسبه شود. چنین پرس و جوهایی در دو مرحله و بصورت زیر پاسخ داده می شوند.

Select tId , Max(cnt) from
 (Select tId, term , Count(SId) as cnt from STC Group by tId, term) tmp
 group by tId

مثال: لیست شماره تمام دانشجویانی که تاکنون ۳ ترم مشروط شده اند.
 (یعنی معدل آنها در ۳ تا از ترمها کمتر از ۱۲ شده است)

Select Sid from
 (Select Sid , term , Avg(mark) as average from STC Group by Sid, term
 Having Avg(mark)<12) tmp
 Group by Sid Having Count(average)=3

تست (دولتی ۸۴)

دربانک زیر کدام گزینه پاسخ پرس و جوی " نام ملوان هائی که همه قایق ها را رزرو کرده اند " می باشد ؟
Sailor (sid , sname , rating , age) ملوان
Boats (bid , bname , color) قایق
Reserves (sid , bid , day) رزرو

Select S.name from sailor S where not exists((select B.bid from Boats B) EXCEPT (Select R.bid from Reserves R) AND S.sid = R.sid) (۱)

Select S.sname from sailor S where not exists (Select B.bid from Boats B where not exists (Select R.bid from Reserves R where R.bid = Bbid and R.sid=S.sid)) (۲)

(۴) هیچکدام

(۳) موارد ۱ و ۲

گزینه :.....

تست (دولتی ۸۵)

در مورد عبارات SQL زیر کدام یک از گزینه ها صحیح است ؟

الف: select avg(balance) from account where branch-name = " Melli"

ب: select branch-name , avg (balance) from account group by branch-name

having avg(balance) > 1200

ج: select branch-name from branch where assets > ANY (select assets from branch where branch-city = 'ISFAHAN')

(۱) فقط عبارت (الف) صحیح می باشد .

(۲) فقط عبارات (الف) و (ب) صحیح می باشد .

(۳) هر سه مورد صحیح می باشد.

(۴) لااقل یکی از عبارات (الف) و (ب) و (ج) و غلط می باشد .

گزینه :.....

تست (دولتی ۸۶)

در آنالیز رابطه ای فرص کنید :

Range of SX is S
Range of PX is P
Range of SPX is SP

آنگاه عبارات زیر چه چیزی بر می گرداند ؟

- ۱) اسامی تولید کنندگانی که همه قطعات را تولید می کنند.
- ۲) اسامی تولید کنندگانی که هیچ قطعه ای را تولید نمی کنند.
- ۳) اسامی تولید کنندگانی که حداقل یک قطعه را تولید می کنند.
- ۴) اسامی تولید کنندگانی که حداقل یک قطعه را تولید نمی کنند.

گزینه ۱ صحیح است.

کدام گزینه شماره قطعاتی را می دهد که توسط بعضی از تولید کنندگان (نه همه) تولید نشده اند؟

1. Select P# from P where P# Not In (Select P# from SP)
2. Select P# from P Where Not Exists (Select S# from S Where Not Exists (Select * from SP where S.S# = SP.S# AND P.P# = SP.P#))
3. Select P# from P Where Not Exists (Select S# from S Where Exists (Select * from SP where S.S# = SP.S# AND P.P# = SP.P#))
4. Select P# from P Where Exists (Select S# from S Where Not Exists (Select * from SP where S.S# = SP.S# AND P.P# = SP.P#))

گزینه ۴ صحیح است. قطعه ای باید انتخاب شود که تهیه کننده ای در جدول S وجود داشته باشد که این تهیه کننده و آن قطعه در کنار هم در جدول SP وجود نداشته باشند.

دستور زیر چه کار می کند؟

Select S# from SP Group By S# Having Sum(Qty) <= Count(Qty)

- ۱) تهیه کنندگانی که کمترین تعداد تولید را دارند
- ۲) تهیه کنندگانی که از هر قطعه فقط یک نمونه تولید کرده اند

۳) تهیه کنندگانی که از هر قطعه بیش از یک نمونه تولید کرده اند
 ۴) خروجی مشخصی ندارد.
 گزینه ۲ صحیح است.

دستور Insert

کار این دستور، درج یک رکورد جدید در یک جدول است.

Insert into (نام جدول) (فیلد ۱, فیلد ۲, ...) **values** (مقدار ۱, مقدار ۲, ...)

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

Insert into s (Id , Name , Family , Field)

Values ('84140', 'Hadi','Karimi','Math')

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry
84140	Hadi	Karimi	Math

زمانی که بخواهیم در تمام فیلدها مقدار وارد کنیم ذکر نام فیلدها ضرورتی ندارد، پس:

Insert into (نام جدول) **values** (مقدار ۱, مقدار ۲, ...)

مثال:

Insert into S Values ('84140', 'Karim', 'Karimi', 'Math')

اما اگر فقط به تعدادی از فیلدها بخواهیم مقدار دهیم، ذکر نام آن فیلدها لازم است:

Insert into S (Id, Name, Family) Values ('84140', 'Karim', 'Karimi')

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Computer
84140	Hadi	Karimi	Null

برای جداولی که کلید Auto Increment دارند، هنگام درج مجبوریم نام فیلدها را ذکر کنیم چون به فیلد کلید نمی توانیم مقدار

دهیم.

رابطه STC

<u>Mykey</u>	SId	CId	TId	Mark
1	84110	101	1	18.5
2	84110	102	2	15
3	84120	102	1	16

Insert into STC (Sid, Cid, Tid, Mark) values('84120', '103', 3, 13)

<u>Mykey</u>	SId	CId	TId	Mark
1	84110	101	1	18.5
2	84110	102	2	15
3	84120	102	1	16
4	84120	103	3	13

تست (دولتی ۸۷)

با اجرای دستور SQL زیر روی بانک اطلاعاتی تولید کنندگان قطعات:

```
INSERT INTO S ( S# , SNAME , CITY )  
VALUES ( 'S10' , 'SMITH' , 'NEW YORK');
```

کدام یک از گزاره های زیر صحیح است ؟

- ۱) دستور با خطای اجرا مواجه می شود .
- ۲) دستور با خطای ترجمه مواجه می شود.
- ۳) اعتبار (STATUS) تولید کنندگان Smith مقدار NULL خواهد شد.
- ۴) اعتبار (STATUS) تولید کنندگان Smith مقدار قبلی خود را حفظ می کند .

گزینه :

درج رکوردهای حاصل از یک Select در یک جدول

بجای مجموعه ثابتی از مقادیر می توانیم مجموعه مقادیر حاصل از یک دستور Select را در یک جدول درج کنیم.

Insert into values (select (فیلدها) نام جدول دستور)

مثال: شماره، نام و نام خ اساتید را در جدول دانشجویان نیز درج کنید

<u>SId</u>	SName	SFamily	Field	<u>TId</u>	TName	TFamily
84110	Ali	Ahmadi	Computer	1	Hadi	Hamidi
84120	Reza	Rezaei	Math	2	Karim	Hassani
84130	Hassan	Hasani	Chemistry			

Insert into S (Sid, SName, Sfamily) values (Select TId,Tname,TFamily from T)

<u>SId</u>	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry
1	Hadi	Hamidi	Null
2	Karim	Hassani	Null

رابط

دستور Update

کار این دستور، تغییر مقادیر رکوردها است:

Update نام جدول set تغییرات Where شرط

تغییرات روی تمام رکوردهایی که شرط را دارند اعمال می شود.

مثال: دستوری که عنوان رشته دانشجویان رشته Computer را به Software تغییر دهد.

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

Update S Set field = 'software' Where field = 'computer'

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	software
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	software

چند مثال:

رابطه S

رابطه T

رابطه C

رابطه STC (جدول)

<u>SId</u>	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

<u>TId</u>	TName	TFamily
1	Hadi	Hamidi
2	Karim	Hassani
3	Ali	Omidi

<u>CId</u>	CName	Units
01	DB	3
02	OS	3

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: نام دانشجویی با شماره ۸۶۱۱۰ را به Reza و رشته او را به software تبدیل کنید.

Update S Set Sname = 'Reza', field = 'software' Where Sid = 86110

مثال: تعداد واحد دروسی را که تاکنون اخذ نشده اند به ۴ تبدیل کنید.

Update C Set Units = 4 Where Cid Not in (Select Cid from STC)

مثال: نمرات کمتر از ۱۰ دانشجویان در درس OS را ۱ نمره اضافه کنید.

Update STC Set mark = mark+1

Where mark<10 and Cid in (Select Cid from C where Cname= 'OS')

دستور Delete

کار این دستور، حذف رکوردهای مورد نظر از یک جدول است.

Delete from نام جدول **Where** شرط

تمام رکوردهایی که شرط را دارند حذف می شوند.

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

مثال: دستوری که دانشجویان رشته Computer را از لیست حذف کند.

Delete from S Where field = 'computer'

<u>Id</u>	Name	Family	Field
84120	Reza	Rezaei	Math

چند مثال:

رابطه

رابطه

رابطه

جدول

رابطه STC

<u>SId</u>	SName	SFamily	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

<u>TId</u>	TName	TFamily
1	Hadi	Harnidi
2	Karim	Hassani
3	Ali	Omidi

<u>CId</u>	CName	Units
01	DB	3
02	OS	3

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: دروسی را که تاکنون اخذ نشده اند از لیست دروس حذف کنید.

Delete from C Where Cid Not in (Select Cid from STC)

مثال: دانشجویانی را که تاکنون بیش از ۱۰ بار افتاده اند را از لیست حذف کنید.

Delete from S Where Sid in

(Select Sid from STC where mark<10 group by Sid Having Count(mark)>10)

دستورات DCL:

دستورات DCL دسته سوم دستورات SQL هستند که امکان برقراری کنترل‌های امنیتی را روی داده‌ها فراهم می‌کنند. مهمترین دستورات DCL در SQL دو دستور GRANT و REVOKE هستند.

دستور GRANT:

با استفاده از دستور GRANT می‌توان مجوز عملیات دلخواهی را روی داده‌های خاصی به یک یا چند کاربر اعطا کرد.

GRANT operations

ON datasource

TO users

[With Grant Option]

مثال: دستور زیر مجوز عملیات حذف از جدول S و همچنین بازیابی تنها ستونهای نام و نام خانوادگی از این جدول را به دو کاربر Ali و John اعطا می‌کند.

GRANT Delete, Select(Name, Family)

ON S

TO Ali, John

With grant option

نوشتن عبارت With Grant Option در پایان دستور Grant اختیاری است. اضافه کردن این عبارت به مثال بالا به Ali و John اجازه می‌دهد که مجوزهای دریافتی را بتوانند به دیگران هم بدهند. مجوزی که برای انجام عملیاتی روی یک جدول به یک کاربر داده می‌شود، می‌تواند محدود به بخشی از داده‌های جدول (بخشی از رکوردها) باشد.

مثال: دستور زیر مجوز بازیابی (مشاهده) تنها رکوردهای مربوط به دانشجویان رشته کامپیوتر را به Ali می‌دهد.

GRANT Select

ON S Where Field= 'Computer'
TO Ali

سوال: فرض کنید یک View روی جدول S بصورت زیر تعریف شده باشد:

```
CREATE View V as  
Select * from S where Field='Computer'
```

همچنین فرض کنید مجوز خواندن از این View را به Ali داده باشیم:
(یعنی Ali مجاز خواهد بود لیست دانشجویان کامپیوتر را ببیند)

```
GRANT Select  
ON V  
TO Ali
```

حال این سوال مطرح است که آیا دستور زیر می تواند توسط Ali صادر شود؟

```
Ali: Select * from S where Field='Computer'
```

جواب:

بظاهر بنظر می آید که Ali درخواست غیر مجازی ندارد. او قصد بازیابی لیست دانشجویان کامپیوتر را دارد که به او مجوز این کار را داده ایم. اما این درخواست از سوی DBMS رد می شود. دلیل آن است که مجوز عملیات Select روی V به Ali داده شده، نه روی S. پس تنها مجاز به خواندن از V است نه از S.

دستور REVOKE:

کار این دستور پس گرفتن مجوز عملیاتی خاص از یک یا چند کاربر می باشد.

```
REVOKE operations  
ON datasource  
FROM users  
[Restrict/Cascade]
```

مثال: دستور زیر مجوز عملیات حذف از جدول S را از Ali بازپس می گیرد.

```
REVOKE Delete  
ON S  
FROM ALI
```

در ارتباط با مثال فوق، ممکن است Ali در زمانی که این مجوز را داشته آنرا به کاربران دیگری هم اعطا کرده باشد و چه بسا به همین ترتیب بطور زنجیری کاربران دیگری این مجوز را گرفته باشند. اگر بخواهیم زمان پس گرفتن مجوز از Ali بخواهیم بطور زنجیری (آبشاری) این مجوز از آن افراد هم گرفته شود، بایستی در پایان

دستور Revoke از کلمه کلیدی Cascade استفاده کرد. در غیر اینصورت، می توان کلمه Restrict را بکار برد و یا اینکه هیچکدام از این دو کلمه را ننوشت.

پرسشهای چهارگزینه ای

کدام یک از عبارات زیر نا درست است؟ (مهندسی IT دولتی ۸۵)

- (۱) استفاده از Correlated Query بجای Join موجب پائین آمدن کارایی (Performance) می شود.
- (۲) هر Query که قابل بیان بصورت یک Natural Join می باشد می تواند بصورت یک Subquery نیز بیان شود .
- (۳) هر Query که قابل بیان بصورت یک Subquery می باشد می تواند بصورت یک Equi Join نیز بیان شود .
- (۴) هر Query که قابل بیان بصورت یک Equi Join می باشد می تواند بصورت یک Natural Join نیز بیان شود .

پرس و جوی ذیل در یک پایگاه داده مطرح است : (مهندسی IT دولتی ۸۵)

« اسامی کارمندانی که مدیر پروژه نیز می باشند را لیست نمایید »

کدام یک از عبارات ذیل برای فرموله کردن پرس و جوی فوق نا درست است ؟

SELECT ename FROM Emps e WHERE
SOME Projs p
SATLSFIES p.pmgr = e.empno (۱)

SELECT ename FROM Emps e WHERE
ONE Projs p
SATLSFIES p.pmgr = e.empno (۲)

SELECT ename FROM Emps e WHERE
e.empno = ANY(
SELECT p.pmgr FROM Projs p) (۳)

```
SELECT
ename
WHERE exists(
SELECT * FROM Projs p
WHERE p.pmgr = e.empno)
```

(۴)

روش مناسب برای تعریف محدودیت های جامعیتی عمومی در SQL چیست ؟
(مهندسی IT دولتی ۸۵)

Create Domain (۲)	Create Table (۱)
Triggered Procedure (۴)	Create Assertion (۳)

درمورد عبارات SQL زیر کدام یک از گزینه ها صحیح است ؟
(کارشناسی ارشد _ دولتی ۸۵)

الف -

Select avg (balance) from account where branch –name ="melli"

ب -

**Select branch-name avg (balance) from account
group by branch-name having avg (balance) > 1200**

ج -

**Select branch-name from branch where assents > ANY (select assets from
branch where brach – city)='ISFAHAN'**

- ۱) فقط عبارت (الف) صحیح می باشد .
- ۲) فقط عبارات (الف) ، (ب) صحیح می باشند .
- ۳) هر سه عبارت (الف) ، (ب) و(ج) صحیح هستند .
- ۴) لااقل یکی از عبارات (الف) ،(ب) ، (ج) غلط می باشد .

کدام یک از عبارات زیر با بقیه عبارات معادل نمی باشد ؟
(کارشناسی ارشد _ دولتی ۸۵)

(۱)

Select P.*
From part P
Where p.p# in (select SP.P# from supply SP where SP.charge > 40000)

(۲)

Select P.*
From part p
Where P.P# = ALL (select S>P# from supply S where S.charge > 40000)

(۳)

Select P.*
From part P
Where Exists (select * from supply SP where P>P# = SP.charge > 40000)

(۴)

Select P.*
From part P
Where P.P# = (select SP.P# from supply SP where P.P# = SP.P# and SP.charge > 40000)

پایگاه داده ای با دستورات زیر ایجاد شده است : (مهندسی IT دولتی ۸۴)

```
CREATE TABLE Books(  
isbn CHAR(20) PRIMARYKEY ,  
bname CHAR(50) ,  
type CHAR(9) CHECK ( typeIN('technical','fiction','self-help'))  
);  
CREATE TABLE Authos(  
Ssn CHAR(9) ,  
isbn CHAR(20)  
PRIMARYKEY(ssn,isbn)  
);
```

(۱)

```
SELECT ssn FROM Authors
EXCEPT
(SELECT ssn
FROM Authors A , Books B
WHERE A.isbn = B.isbn AND B.type='fiction')
```

(۲)

```
SELECT ssn
FROM Authors , Books
WHERE Authors.isbn = Books.isbn
AND type <> 'fiction'
```

(۳)

```
SELECT ssn
FROM Authors A
WHERE NOT EXISTS
(SELECT *
FROM Books B
WHERE A.isbn = B.isbn AND B.type ='fiction')
```

(۴)

```
SELECT ssn
FROM Authors
WHERE ssn NOT IN
(SELECT ssn
FROM Authors A,Books B
WHERE A.isbn =B.isbn
AND B.type ='fiction')
```

فرض کنید V یک View باشد که به صورت زیر مجموعه سطری و ستونی (Row and Column Subset) از رابطه پایه ای R تعریف شده است. برای آنکه هرگونه تغییر (Update) روی V مجاز باشد، کدامیک از محدودیتهای زیر کافی است؟
(کارشناسی ارشد _ دولتی ۷۳)

(۱) V بایستی شامل کلید اصلی رابطه R باشد.

(۲) V بایستی شامل تمام خصیصه های (Attributes) غیر Null در R باشد.

(۱)

```
Select name
  From Person
  Where no = father_no;
```

(۲)

```
Select p1.name
  From Person P1, Person P2
  Where P1.father_no = P2.father_no;
```

(۳)

```
Select name
  From Person
  Where name IN( select name from Person where no = father_no);
```

(۴)

```
Select P1.name
  From Person P1 , Person P2
  Where P1.no = P2.father_no;
```

جدول A دارای ستون **Price** و **Pcode** است. کدام گزینه در مورد عبارت زیر صحیح است ؟
(کارشناسی ارشد _ دولتی ۷۵)

```
Select Pcode , Price
  From A
  Where Price = Max(Price);
```

(۱) معادل عبارت مقابل است :

```
Select Pcode , Price
  From A
  Where Price =(select Max(Price) from A);
```

(۲) اشکال منطقی دارد زیرا محاسبه ماکزیمم در مرحله جداگانه ای انجام نمی شود .

(۳) همواره یک سطر چاپ می کند که غلط است .

(۴) معادل عبارت مقابل است :

```
Select Pcode , Max(Price)
  From A
  Where Max(Price) = ( select Price from A);
```

جداول زیر برای یک بانک مفروض است: (کارشناسی ارشد _ دولتی ۸۲)

مشتری **CUSTOMER** (CUSTNAME , ADDRESS)

حساب **ACCOUNT** (ACCNO , CUSTNAME)

LOAN(LOANNO , CUSTNAME)

کدام دستور SQL زیر برای مشتریانی که وام نگرفته اند کافی است ؟

(۱)

```
SELECT * FROM CUSTOMER , LOAN
WHERE CUSTOMER .CUSTNAME < >
LOAN.CUSTNAME
```

(۲)

```
SELECT * FROM CUSTOMER , LOAN
WHERE CUSTOMER .CUSTNAME =
LOAN.CUSTNAME
```

(۳)

```
SELECT * FROM CUSTOMER
WHERE CUSTNAME NOT IN
(SELECT CUSTNAME FROM LOAN)
```

(۴) هیچکدام

ضد نیم اتصال دو رابطه R و S بنا بر تعریف شامل تاپل های آویزان R می شود. به این معنا که تاپلهایی که متناظری در S و همچنین در هیچ رابطه دیگری ندارند. روابط R(A,B,C) و S(A,B,D) را در نظر بگیرد که بر اساس صفات مشترک A و B متصل می شوند و فوض کنید که <A,B> کلیدی برای هر دو رابطه می باشد. کدام گزینه یک ضد نیم اتصال را بطور صحیح پیاده می کند ؟

(مهندسی IT_ دولتی ۸۴)

(۱)

select R.* from R,S where R.A <> S.A AND R.B <> S.B

(۲)

Select * from R where not exists (select * from S where S.A = R.A OR S.B = R.B)

(۳)

Select * from R where not exists (select * from S where S.A = R.A AND S.B = R.B)

(۴)

Select * from R where A <> any (select A from S) AND B <> any (Select B from S)

در بانک زیر ، کدام گزینه پاسخ پرس و جوی « نام ملوان هائی که همه قایق ها را رزرو کرده اند » می باشد ؟ (کارشناسی ارشد _ دولتی ۸۴)

Sailor (sid , sname , rating , age) ملوان

Boats (bid , bname , color) قایق

Reserves (sid , bid , day) رزرو

(۱)

Select S.name from sailor S where not exists ((select B.bid from Boats B) EXCEPT

(Select R.bid from Reserves R) AND S.sid)

(۲)

Select S.name from sailor S where not exists (select B.bid from Boats B where not exists (Select R.bid from Reserves R where R.bid =B.bid and R.sid = S.sid))

(۳) موارد ۱ و ۲

(۴) هیچکدام

کدامیک از موارد زیر در رابطه با یک DBMS رابطه ای (مثل INGRES) مصداق می کند ؟

(کارشناسی ارشد _ آزاد ۷۱)

- (۱) کاربر می بایست QUERY خود را بصورت بهینه (OPTIMIZED) به سیستم وارد کند .
 - (۲) کاربر باید بتواند QUERY خود را به هر صورت که می خواهد فرموله به سیستم وارد کند .
 - (۳) استراتژی اجرای یک QUERY بر اساس متدهای دستیابی (ACCESS METHODS) تعریف شده روی رابطه های پایه سیستم تعیین و به اجرا گذاشته می شود .
- (۴) ۲۰۳

رابطه تولید کننده $S(S\#,City,Status)$ مفروض است . برای بازیابی $S\#$ و $Status$ مربوط به تولید کنندگان شهر کرمان بواساس ترتیب نزولی $Status$ کدام یک از تستورات SQL زیر صحیح می باشد ؟

(کارشناسی ارشد _ دولتی ۷۴)

(۲) `Select # , Status
where City = 'Kernam'
From S
Order By Status;`

(۱) `Select S# , Status
From S
Where City = 'Kernam'
Order By Status Desc;`

(۴) `Order by Status
From S
Select Status , S#
Where City ='Kernam'`

(۳) `Select Status , S#
Where City = 'Kernam'
Order By Status Desc
From S`

دو رابطه $R(A,B,C,D)$ و $R'(A,B,C',D')$ در بانک اطلاعاتی رابطه ای مفروض است و دیدگاه (View) به نام RR' را به شکل زیر تعریف می کنیم : (کارشناسی ارشد _ دولتی ۷۴)

Create Vies RR' As

(Select A,B from R)
Union
(Select A,B from R')

کدام گزینه با تعریف فوق صادق است ؟

RR'(A,B,A,B) (۲)
RR'(C',c) (۴)

RR'(C',D',C,D) (۱)
RR'(A,B) (۳)

بافرض داشتن پایگاه داده رابطه ای زیر:

S(S# , SNAME , STATUS , CITY) : تهیه کننده

P(p# , PNAME , COLOR,WEIGHT,CITY) : قطعه

J(J# ,JNAME ,CITY) : پروژه

SPJ : (S#,P#,J#,QTY)

رویه پاسخگویی زیر ار در SQL در نظر می گیریم:

```
SELECT DISTINCT P#  
FROM SPJ SPJX  
WHERE NOT EXISTS  
(SELECT *  
FROM J  
WHERE CITY ='C2'  
AND NOT EXISTS  
(SELECT *  
FROM SPJ SPJY  
WHERE SPJY.P# = SPJX.P#  
AND SPJY.J# = J.J#))
```

این رویه پاسخگو ، پاسخ کدام پرسو جو رابه دست می آورد؟(کارشناسی ارشد _ دولتی ۷۲)

- (۱) شماره قطعاتی را بیابید که برای تمام پروژه های شهر C2 تهیه شده است .
- (۲) شماره قطعاتی را بیابید که برای تمام پروژه های شهر C2 تیه نشده باشند .
- (۳) شماره قطعاتی را بیابید که اقلأ برای یک پروژه شهر C2 تهیه شده باشند .
- (۴) شماره قطعاتی را بیابید که برای هیچ پروژه شهر C2 تهیه نشده باشند .

اگر داشته باشیم :

(حقوق : Salary و واحد : dept و نام کارمند : empname و شماره کارمند : empno employee)

آنگاه پاسخ پرسو جوی زیر کدام گزینه می باشد ؟ (کارشناسی ارشد _ دولتی ۸۳)

پرسو جو : نام حقوق کارمندانی که متوسط حقوقشان از متوسط حقوق واحدشان کمتر ولی از متوسط حقوق کل کارکنان سازمان بیشتر است .

(۱)

Select empname , salary
From employee
Group By dept having salary > avg (salary)

(۲)

Select empname , salary
From employee
Having salary > select avg (salary) from employee

(۳)

Select empname , salary
From (select * from employee where salary > (select avg (salary) from employee))
Group By dept Having salary < avg (salary);

(۴)

Select empname salary
From employee where salary > avg (salary)
Group By dept Having salary > avg (salary)

با توجه به جدول employee در سؤال بالا ، دستور زیر چه کاری انجام می هد ؟

(کارشناسی ارشد _ دولتی ۸۳)

Grant select , update (empno , empname) on employee to Roya

- (۱) این دستور مشکل ساختاری دارد .
- (۲) اجازه به روز در آوردن و نیز انتخاب ستون های جدول employee را به Roya می دهد .
- (۳) اجازه به روز در آوردن و نیز انتخاب ستون های جدول را از Roya پس می گیرد .
- (۴) اجازه به روز در آوردن empname و empno و نیز انتخاب ستون های جدول را به Roya می دهد .

در صورتی که مفهوم رابطه $SPJ \langle S\#,P\#,J\#,QTY \rangle$ به شکل ذیل بیان گردد :

« تهیه کننده S# ، قطعه P# را برای پروژه به تعداد QTY تهیه نموده است » و کلید اصلی رابطه S#,P#,J# باشد : برای فرموله کردن پرس و جوی زیر به زبان SQL کدام قسمت تکمیلی باید به دستور SQL زیر ملحق شود؟ (مهندسی IT دولتی ۸۳)

پرس و جو : دستور SQL : شماره پروژه هایی را معین کنید که تمام قطعات آنها را یک تهیه کننده ، تهیه کرده باشد . Select J# from SPJ	
--	--

(۱)

Group by S#

Having Count (*) = 1;

(۲)

Group by P#

P# , Having Count (*) = 1;

(۳)

Group by J#,P#

Having Coutn (S#) = 1;

(۴)

Group by J#

Having Count (Distinct S#) = 1;

جدول محصول به صورت زیر مفروض است :

goods (P#, Pname , Price)

خروجی دستور زیر کدام گزینه است؟ (مهندسی IT دولتی ۸۳)

Select Prince from goods

Where Price >= Max(Price);

- (۱) Null
(۲) کل جدول
(۳) بستگی به داده ها دارد
(۴) ماکزیمم قیمت در جدول

کدام یک از گزینه ها نا درست است؟ (مهندسی IT دولتی ۸۳)

- (۱) قانون اول و دوم جامعیت داده ای بوسیله تعریف دامنه (domain) برای ویژگی ها قابل انجام است .
(۲) برای اعمال قانون اول جامعیت داده ای در SQL کافی است در تعریف ویژگی که بعنوان کلید اصلی است not null و unique قید شود .
(۳) برای اعمال قانون اول جامعیت داده ای در SQL می بایست در ایجاد جدول کلید اصلی بوسیله primary key تعریف شود .
(۴) برای اعمال قانون دوم جامعیت داده ای در SQL می بایست در ایجاد جدول کلید خارجی بوسیله foreign key تعریف شود .

اگر **employee(empno,empname,dept,salary)** شماره کارمند : **empno** نام

کارمند: **empname** ، واحد: **dept** و حقوق : **salary** باشد ، آنگاه جواب پرسش زیر کدام یک از

گزینه ها می باشد؟ (مهندسی IT دولتی ۸۳)

(۱)

```
Select *  
From employee  
Where salary > avg ( salary );
```

(۲)

```
Select *  
From employee  
Having salary > avg ( salary );
```

(۳)

```
Select *  
From employee  
Where salary > ( select avg ( salary ) from employee )
```

(۴)

```
Select *  
From employee  
Group By dept Having salary > avg ( salary ) ;
```

دستور SQL زیر در بانک اطلاعاتی تولید کنندگان و قطعات چه کار می کند ؟
(مهندسی IT دولتی ۸۶)

```
UPDATE P  
SET COLOR = 'Orange'  
WHERE COLOR = 'Red'
```

- ۱) تمام قطعات قرمز رنگ را به رنگ نارنجی در می آورد.
- ۲) تمام قطعات نارنجی را به رنگ قرمز در می آورد.
- ۳) اولین قطعه قرمز رنگ را به رنگ نارنجی در می آورد.
- ۴) آخرین قطعه قرمز رنگ را به رنگ نارنجی در می آورد.

در بانک اطلاعاتی تولید کنندگان و قطعات ، دستور SQL زیر باعث: (مهندسی IT دولتی ۸۶)

```
CREATE VIEW LONDONSUP  
AS SELECT *  
FROM S  
WHERE CITY = 'LONDON'
```

- ۱) افزایش سرعت پاسخگویی به برخی از پرس و جوها می شود .
- ۲) خطای زمان اجرا می شود .
- ۳) خطای نحوی (Syntax) می شود .
- ۴) سهولت بیان برخی از پرس و جوها می شود .

کدام دستور با معنای داده شده در SQL وجود ندارد؟ (مهندسی IT دولتی ۸۶)

۲) Delete برای ازبین بردن جداول

۱) Update برای تغییر View

۳) Revoke برای لغو اجازه دسترسی

۴) Update برای تغییر جداول پایه

اگر رابطه های $S(S\#,SNAME,STATUS,CITY)$ و $SP(S\#,P\#,QTY)$ مفروض باشد ، معادل پرسش زیر کدام یک می باشد؟ (مهندسی IT دولتی ۸۶)

```
SELECT SNAME , CITY
FROM S
WHERE S# IN ( SELECT S#
FROM SP
WHERE P# = 'P1')
```

(۱)

```
SELECT SNAME , CITY FROM S
WHERE S# = ANY ( SELECT S# FROM SP WHERE P# = 'P1')
```

(۲)

```
SELECT SNAME , CITY
FROM S
WHERE EXISTS ( SELECT * FROM SP WHERE SP.P# = 'P1' AND
SP.S#=S.S#)
```

(۳)

```
SELECT SNAME , CITY
FROM S, SP
WHERE S.S# = SP.S# AND SP.P# = 'P1'
```

(۵) هر سه معادل می باشند .

دو جدول x و y مفروض هستند ، در صورت اجرای دستور SQL زیر کاردینالیتی جدول حاصل کدام است؟ (مهندسی IT آزاد ۸۵)

x

F	B
f1	b1
f2	b2

y

A	B	C
a1	b1	MK
a2	b4	MTK

Select *
 From y Right
 ON
 Where y.c Like 'M%';

a1	b2	LN
a2	b1	RP

outer join x
y.B=x.B

3 (۴)

4 (۳)

1 (۲)

2 (۱)

**

پایگاه داده ای دارای سه جدول S و L و g به صورت زیر است :

S(S# , Sname) , L(L# , Lname) , G(S#,L#,Mark)

که در آن جدول S جدول دانشجویی با خصیصه های شماره دانشجویی و نام دانشجو. جدول L جدول درسی با خصیصه های شماره و نام درس و جدول G نمره با خصیصه های شماره دانشجویی شماره درس و نمره درس است با توجه به این پایگاه داده به ۲ سؤال بعدی پاسخ دهید .

**

با توجه به جدول پایگاه داده فوق خروجی دستور SQL زیر کدام است؟ (مهندسی IT_آزاد ۸۶)

**SELECT DISTINCT Sname
 FROM S
 WHERE NOT EXISTS (SELECT S# FROM G WHERE L#='۱'**

- (۱) نام تمام دانشجویانی را مشخص می کند که از درس ۱ نمره دارند.
- (۲) نام بعضی از دانشجویانی را مشخص می کند که از درس ۱ نمره ندارند .
- (۳) نام تمام دانشجویانی را مشخص می کند که از درس ۱ نمره ندارند.

۴) نام تمام دانشجویانی را مشخص می کند که از درس # حداکثر یک نمره دارند .

با توجه به جدول پایگاه داده سؤال قبلی پرس و جوی عبارت SQL زیر کدام است ؟
(مهندسی IT_آزاد ۸۶)

```
SELECT DISTINCT S#  
FROM G  
WHERE Mark = ( SELECT MAX ( Mark ) FROM G)
```

- ۱) شماره دانشجویانی را مشخص می کند که بیشترین نمره را اخذ نموده اند .
- ۲) شماره بعضی از دانشجویانی را مشخص می کند که بیشترین نمره را اخذ نموده اند .
- ۳) شماره دانشجویانی را مشخص می کند که نمره بیشتری اخذ نموده است .
- ۴) این دستور مشکل گرامری دارد .

در عبارت : (مهندسی IT_آزاد ۸۶)

```
SELECT State , COUNT ( State )  
FROM Customer_T  
GROUP BY State HAVING COUNT ( State ) >1
```

- ۱) می توانیم از یک Subquery استفاده کنیم .
- ۲) می توانیم از WHERE بجای HAVING استفاده کنیم .
- ۳) می توانیم شرط $COUNT (State) < 10$ را با شرطی فعلی ترکیب کنیم .
- ۴) می توانیم از $COUNT (*)$ بجای $COUNT (State)$ استفاده کنیم .