

# دستورات SQL

در کلیه مثالها از پایگاههای زیر استفاده شده است

Term		Students		Courses		Teachers	
StudentCode	*	StudentCode	*	CourseCode	*	TeacherCode	*
CourseCode	*	StudentName		CourseName		TeacherName	
TeacherCode	*	Adrs		Dep		Phone	
Grade		Sex		Unit		Salary	
		City					

فیلدهای ستاره دار ( \* ) کلیدها را مشخص می کنند

## ۱) انتخاب یکسری فیلدها از یک پایگاه

```
Select Field1 [, Field2 , ....] From <Table Name>
```

### Example :

```
Select TeacherCode,TeacherName  
From Teachers
```

```
Select StudentName  
From Students
```

```
Select *  
From Courses
```

## انتخاب همه فیلدها

## ۲) انتخاب یکسری فیلدها با شرط

```
Select Field1 [, Field2 , ....] From <Table Name> Where <Condition>
```

### Example :

```
Select TeacherCode,TeacherName  
From Teachers  
Where TeacherCode>100
```

## در Where Clauses مي توان از And,Or,Not استفاده كرد

```
Select StudentName  
From Students  
Where StudentCode>10 Or StudentName='Ali'
```

```
Select * From Courses Where Not (Unit=4)
```

## براي ايجاد محدوده در يك مورد يك فيلد

```
Select Field1 [, Field2 , ....] From <Table Name>  
Where Field Between StartRange And EndRange
```

### Example

```
Select StudentName  
From Students  
Where StudentCode Between 11 And 19  
Select *  
From Teachers  
Where TeacherName Between 'Ali' And 'Sasan'
```

## براي پيدا كرد كلمات مشابه

```
Select Field1 [, Field2 , ....] From <Table Name>  
Where Field Like Value
```

### Example

```
Select StudentName  
From Students  
Where StudentName Like 'Ali'
```

براي تعميم كلمات مشابه به هرطولي

```
Select *  
From Courses  
Where CourseName Like 'Math%'
```

براي تعميم كلمه مورد نياز به هر محلي

```
Select StudentName  
From Students  
Where StudentName Like '%Ali%'
```

## براي پيدا كردن يكسري كلمات خاص

```
Select Field1 [, Field2 , ....] From <Table Name>  
Where Field In (Valu1, Valu2, Valu3,....)
```

## Example

```
Select StudentName
  From Students
  Where StudentCode In (101,950,342)
```

برای پیدا کردن رکوردهای پرنشده (NULL)

```
Select Field1 [, Field2 , ....] From <Table Name>
Where Field Is Null
```

**تذکر مهم :** NULL با Blank کاملاً متفاوت است . چنانچه در فیلدی Space ریخته شود آن فیلد NULL نیست اما اگر در فیلدی هیچ چیزی ریخته نشود NULL خواهد بود

## (۳) برای مرتب سازی

```
Select Field1 [, Field2 , ....] From <Table Name>
Order By <Field List> [Desc | Asc]
```

## Example

```
Select StudentName
  From Students
  Order By StudentName
```

مرتب سازی برحسب شماره فیلدهای انتخابی

```
Select StudentName,StudentCode
  From Students
  Order By 2
```

مرتب سازی نزولی و صعودی

```
Select StudentName,StudentCode
  From Students
  Order By StudentName Asc ,StudentCode Desc
```

**تذکر مهم :** فیلدهایی که در Order By قید می شوند حتماً می بایستی در Select قید شده باشند

## (۴) استفاده از عبارات ریاضی در Select

### (۴,۱) استفاده از توابع آماده ریاضی

```
Select FunctionName (Field1) [ As AliasName ] From <Table Name>
```

## لیست توابع ریاضی (FunctionName)

<b>Max</b>	حداکثر یک فیلد
<b>Min</b>	حداقل یک فیلد
<b>Sun</b>	حاصل جمع فیلد
<b>Count</b>	شمارش تعداد
<b>Avg</b>	میانگین فیلد

### Example

Select Max(StudentCode)  
From Students

پیدا کردن حداکثر ، حداقل و متوسط حقوق معلمان

Select Max(Salary) As MaxS, Min(Salary) As MinS, Avg(Salary)  
From Teachers

Select Max (Salary) From Teachers Where TeacherCode Between [1000,1500]  
تعداد دروس

Select Count(\*) From Courses

## ٤,٢) استفاده از محاسبات ریاضی

Select AliasName = <Calculate Expretion> From <Table Name>  
Select <Calculate Expretion> **As** AliasName From <Table Name>

### Example

Select Salary , NewSal = Salary +1000  
From Teachers

Select TeacherName , Salary\*1.2 As PayIt , Phone  
From Teachers

## ٥) خلاصه سازی (عدم نمایش رکوردهای مشابه)

Select **Distinct** Field From <Table Name>

### Example

```
Select Distinct City  
From Students
```

نمایش کلیه شهرها بدون تکرار

```
Select City  
From Students
```

نمایش کلیه شهرها با تکرار

```
Select Count (Distinct City )  
From Students
```

شمارش تعداد شهرها بدون تکرار

### ٦ گروه بندی (دسته بندی)

```
Select Field1 [,Field2,....] From <Table Name> Group By <Field List>
```

### Example

```
Select StudentName  
From Students  
Group By StudentCity , StudentName
```

تعداد دانش آموزان به تفکیک شهر

```
Select City , Count(*)  
From Students  
Group By City
```

تعداد دروس ثبت نام شده هر دانش آموز و میزان شهریه دانش آموزان

```
Select StudentCode, Count(*) As TerCourse , Count(*) * 45000 As MustPay  
From Term  
Group By StudentCode
```

**تذکر مهم :** فیلدهایی (و نه توابع یا محاسبات) که در Select قید می شوند حتما می بایستی در Group By نیز قید شده باشند.

### ٧ شرط گذاری روی گروه بندی (دسته بندی)

```
Select Field1 [,Field2,....] From <Table Name> Group By <Field List>  
Having <Condition>
```

### Example

```
Select City , Count(*)  
From Students  
Group By City  
Having Sex = 'M'
```

تعداد دانش آموزان مرد به تفکیک شهر

نام شهر هايي كه بيش از ۳ دانش آموزدارند

```
Select City , Count (*)  
From Students  
Group By City  
Having Count(*) >3
```

**تذکر مهم:** Where و Having هيچ تفاوتی ندارند . اما having روی اطلاعاتی که دسته بندی شده اند عمل می کند .

## ۸) ترکیب پایگاهها

```
Select Table1.Field1 ,Table2.Field2 [ ,....] From <Table1,Table2,...>
```

### Example

دروس هر استاد در ترم

```
Select Teachers.TeacherName, Teachers.TeacherCode , Term.CourseCode  
From Teachers, Term
```

**تذکر مهم:** این روش در ۹۹% موارد کاربرد ندارد. فرض کنید پایگاه Term دارای ۱۰۰ رکورد باشد و پایگاه Teachers ۴ رکورد . دستور فوق باعث می گردد تا به تعداد حاصل ضرب رکورد ها (۴۰۰) رکورد جواب پیدا کنید که غلط است . اما در موارد که نیاز به نمایش تمامی حالات امکان پذیر می باشید، ممکن است مفید باشد.

دروس هر استاد در ترم

```
Select Teachers.TeacherName, Teachers.TeacherCode , Term.CourseCode  
From Teachers, Term  
Where Teachers.TeacherCode= Term.TeacherCode  
نمرات دروس دانش آموزان (۱)
```

```
Select Students.StudentName, Term.CourseCode , Term.Grade  
From Students,Term  
Where Students.StudentCode= Term.StudentCode
```

**تذکر مهم:** چنانچه فیلدی در پایگاهها مشترک نیستند می توانید نام پایگاه را ذکر نکنید .

نمرات دروس دانش آموزان (۲)

```
Select StudentName, Term.CourseCode , Grade  
From Students,Term  
Where Students.StudentCode= Term.StudentCode
```

**تذکر مهم :** توصیه می شود همیشه برای پایگاههای خود Alias (نام مجازی) تعریف کنید

نمرات دروس دانش آموزان (۳)

```
Select StudentName, D1.CourseCode , Grade
      From Students D2 ,Term D1
      Where D2.StudentCode= D1.StudentCode
```

نام و نمره دانش آموزان خانم که بالاتر از در یک درس بیشتر از ۱۵ گرفته اند

```
Select StudentName, CourseName , Grade
      From Students D1, Courses D2 , Term D3
      Where D1.Sex='F' And D3.Grade>=15      And
            D1.StudentCode=D3.StudentCode    And
            D3.CourseCode=D2.CourseCode
```

نمرات دروس دانش آموزان به ترتیب (۴)

```
Select StudentName, CourseName , Grade
      From Students D1 ,Term D3, Courses D2
      Where D1.StudentCode=D3.StudentCode And
            D3.CourseCode=D2.CourseCode
      Order By CourseName , StudentName
```

## ۸) ارتباط پایگاهها

```
Select Field1 ,Field2 [ ,....] From <Table1> Inner join <Table2> On
      Table1.Field = Table2.Field
```

نمرات دروس دانش آموزان (۵)

```
Select StudentName, D3.CouseCode, Grade
      From Students D1 Inner Join Term D3 On
      D1.StudentCode=D3.StudentCode
```

## ۹) Select های تودرتو

```
Select Field1 ,Field2 [ ,....] From <Table1,Table2,...>
      Where Field Operand (Select Field1 ,Field2 [ ,....] From
      <Table1,Table2,...>)
```

## لیست عملگرها (Oprand)

IN / Not IN	درمجموعه هست
Exists / Not Exists	اگر وجود دارد
Any	حداقل یکی
All	حتما همه
> , < , = , <= , >= , <>	میانگین فیلد

### Example

نام دانش آموزانی که نمره ۱۵ گرفته اند

```
Select StudentName
  From Students
  Where StudentCode IN
    (Select StudentCode
     From Term
     Where Grade>15)
```

**نکته مهم:** چرا از Select های تودر تو استفاده می کنیم . زیرا اینکار باعث افزایش سرعت و راحتی خواندن و نوشتن دستورات SQL می گردد. علاوه برآن بعدا خواهید ترکیب این قابلیت با دیگر دستورات بسیار مفید خواهد بود.

نام دانش آموزانی که نمره ۱۵ گرفته اند

```
Select StudentName
  From Students
  Where StudentCode =
    (Select StudentCode
     From Term
     Where Grade>15)
```

**تذکر مهم:** شما می توانید يك پایگاه را با خودش ترکیب کنید نام معلمانی که کمتر از حد متوسط حقوق دریافت می کنند

```
Select TeacherName
  From Teachers
  Where Salary < (Select Avg(Salary)
                  From Teachers)
```

نام دانش آموزانی که حداقل در يك درس ۱۶ گرفته اند

```
Select StudentName
  From Students
  Where StudentCode = Any (Select StudentCode
                           From Term
                           Where Grade=16)
```

نام دانش آموزاني که داراي معدلبيش از متوسط معدل است

```
Select StudentName , Avg(Grade)
  From Students D1, Term D2
 Where D1.StudentCode = D2.StudentCode
 Group By StudentName
 Having Avg(Grade) > Any (Select Avg(Grade)
                          From Term
                          Group By StudentName)
```

نام دانش آموزاني که در هيچ درسي ثبت نام نکرده اند

```
Select StudentCode, StudentName
  From Students D1
 Where Not Exists ( Select *
                   From Term
                   Where StudentCode = D1.StudentCode)
```

نام کلاسهايي که فقط شاگرد مرد دارند

```
Select Distinct CourseName
  From Course D1
 Where 'M' = All ( Select Sex
                  From Term D2 , Students D3
                  Where D2.StudentCode = D3.StudentCode
                  And
                  D2.CourseCode = D1.CourseCode)
```

نام شاگرداني که بيش از ۲ درس گرفته اند

```
Select StudentName
  From Students S
 Where Exists (Select Count(*)
              From Term
              Where StudentCode= S.StudentCode
              Having Count(*)>2)
```

**تذکر مهم :** ممکن است Having را بدون استفاده از Group By در Select هاي داخلي بکار ببريد . اينکار باعث مي شود تا Having جواب را بصورت يك گروه بزرگ نگاه کند و سپس شرط را اعمال کند. نام دانشکده هايي که بيش از ۴ دانش آموز دارند

```
Select Dep
  From Courses C
 Where Exists (Select Count(*)
              From Term
              Where CourseCode= C.CourseCode
              Having Count(*)>4)
```

## در پایان فرم کلي دستور **Select**

```
Select <Field List >
      From <Table List >
      Where Condition
      Group By <Field List >
      Having Condition
      Order By <Field List >
```

## ۱۰) ایجاد جدول

```
Create Table <Table Name> (Columns)
```

### Example

```
Create Table Students ( StudentCode SmallInt Not Null Unique ,
                        StudentName Char (20) ,
                        Adrs Char (50) ,
                        City Char (10) ,
                        .....
                        )
```

ایجاد پایگاه دانش آموزان

## ۱۱) حذف جدول

```
Drop Table <Table Name>
```

### Example

```
Drop Table Students
```

حذف پایگاه دانش آموزان

## ۱۲) اضافه کردن رکورد

```
Insert Into Table <Table Name> [(Columns Name)] Values <Values>
```

```
Insert Into Table <Table Name>  
Select <Fields>  
From <Table Name>  
Where <Condition>
```

### Example

رکورد جدید در پایگاه دروس (۱)  
Insert Into Courses (CourseCode,CourseName,Dep,Unit)  
Values (1005 , 'Math 2' , 'Mt' , 4)

رکورد جدید در پایگاه دروس (۲)  
Insert Into Courses  
Values(1005,'Math 2','Mt',4)

## ۱۳) تغییر مقدار رکورد

```
Update <Table Name> Set <Column Name> = <Value> Where  
<Condition>
```

### Example

تغییر واحد درس ریاضی  
Update Course  
Set Unit = 3  
Where CourseCode=1005

تغییر حقوق معلمان زیر مقدار متوسط  
Update Teachers  
Set Salary=1.2 \* Salary -100  
Where Salary < (Select Avg(Salary)  
From Teachers)

## ۱۴) حذف رکورد

```
Delete From <Table Name> Where <Condition>
```

### Example

حذف درس ریاضی  
Delete From Course  
Where CourseCode=1005

## (۱۴) ایجاد يك نما (View)

```
Create View <View Name> (<Column Names>) As  
Select <Column Names>  
From <Table Names>  
Where <Condition>
```

### Example

ایجاد يك جدول موقت از معلمان و حقوق و ۲ برابر حقوق  
Create View TeachLook (TCode , TName , MainSal , Sal2)  
Select (TeacherCode , TeacherName , Salary , Salary \*2)  
From Teachers

Create View Resome (StName , Ccode , Grade )  
Select StudentName, D3.CouseCode, Grade  
From Students D1 Inner Join Term D3 On  
D1.StudentCode=D3.StudentCode

**تذکر مهم :** پس از ساخت يك View مي توان با آن همچون يك Table رفتار کرد و در آن Insert, Update و Delete انجام داد.

## (۱۵) حذف نما

```
Drop Veiw <View Name>
```

### Example

حذف نماي دانش آموزان  
Drop View Students

## (۱۶) ایجاد ایندکس

```
Create [Unique] Index <Index Name> ON <Table Name> (<Field  
Names>)
```

### Example

ایجاد يك ایندکس از نلم بروي پایگاه معلمان (۱)  
Create Index TNameIDX ON Teachers (TeacherName)  
ایجاد يك ایندکس از نلم بروي پایگاه معلمان (۲)  
Create Index TNameIDX ON Teachers (TeacherCode,TeacherName)

## (۱۷) حذف ایندکس

```
Drop Index <Index Name>
```

### Example

Drop Index TNameIDX

(۱۸) Union (ادغام ۲ پایگاه با فیلدهای مشابه )

```
Select <Fields> From <Table Name>  
Union  
Select <Fields> From <Table Name>
```

### Example

```
Select StudentCode , StudentName  
Form Students  
Union  
Select TeacherCode , TeacherName  
Form Teachers
```

## جدول انواع داده ها در SQL Table

نوع فیلد	توضیح
Bigint	صحیح $2^{63} \sim -2^{63}$
Int	صحیح $2^{31} \sim -2^{31}$
Smallint	صحیح $2^{15} \sim -2^{15}$
Tinyint	0 ~ 255
Bit	0 ~ 1
Float	اعشاری $1.79^{308} \sim -1.79^{308}$
Real	اعشاری $3.4^{38} \sim -3.4^{38}$
DateTime	1/1/1753 ~ 31/12/9999
SmallDateTime	1/1/1900 ~ 6/6/2079
Char	طول ثابت ، حداکثر ۸۰۰۰ کارکتر
VarChar	طول متغیر ، حداکثر ۸۰۰۰ کارکتر
Text	طول متغیر ، حداکثر $2^{31}$ کارکتر
Nchar	طول ثابت ، حداکثر ۴۰۰۰ کارکتر Unicode
Nvarchar	طول متغیر ، حداکثر ۴۰۰۰ کارکتر Unicode
Ntext	طول متغیر ، حداکثر $2^{30}$ کارکتر Unicode
Binary	طول ثابت حداکثر ۸۰۰۰ بایت
VarBinary	طول متغیر حداکثر ۸۰۰۰
Image	طول متغیر حداکثر $2^{31}$
Uniqueidentifier	شناسه منحصر به فرد جهانی

پایان  
وکیلی ۸۲/۰۶/۱۶