

استخراج اطلاعات از پایگاه داده

■ دستور استخراج داده ها در MySQL

`SELECT نام_جدول FROM نام_ستون_یا_ستون_ها`;

■ مثال: لیست شناسه اعضاء و نام آنها

`SELECT MemberId, FirstName FROM MemberDetails;`

MemberId	FirstName
1	Katie
3	Sandra
4	Steve
5	John
6	Jenny
7	John
8	Jack
9	Seymour

استخراج اطلاعات از پایگاه داده

- برای نمایش تمام ستون های یک جدول در خروجی به جای آوردن نام تمامی آنها از علامت * استفاده می کنیم

```
SELECT * FROM Location;
```

معادل

```
SELECT LocationId, Street, City, State FROM Location;
```

LocationId	Street	City	State
1	Main Street	Orange Town	New State
2	Winding Road	Windy Village	Golden State
3	Tiny Terrace	Big City	Mega State

استخراج اطلاعات از پایگاه داده

- استخراج داده های بدون تکرار
- خروجی دستور زیر را در نظر بگیرید:

```
SELECT City FROM MemberDetails;
```

City
Townsville
Orange Town
New Town
Orange Town
Orange Town
Big City
Windy Village

استخراج اطلاعات از پایگاه داده

■ استخراج داده های بدون تکرار

■ برای حذف ردیف های تکراری از کلمه کلیدی DISTINCT استفاده می کنیم

```
SELECT DISTINCT City FROM MemberDetails;
```

City
Townsville
Orange Town
New Town
Big City
Windy Village

نکته: اگر این کلمه کلیدی بر روی ستونی که کلید اصلی است به کار برود، نتایج تکراری حذف نمی شوند

استخراج اطلاعات از پایگاه داده

■ استفاده از نام مستعار برای ستون ها

■ می توان برای اینکه عبارت دیگری به جای نام اصلی ستون در خروجی دستور انتخاب نشان داده شود، از AS استفاده کرد

```
SELECT LastName AS Surname FROM MemberDetails;
```

Surname
Smith
Robson

استخراج اطلاعات از پایگاه داده

■ فیلتر کردن نتایج با Where

■ می توان شرطی را برای نمایش نتایج در خروجی اعمال نمود

```
SELECT FirstName + ' ' + LastName AS 'Full Name'  
FROM MemberDetails  
WHERE City = 'Big City';
```

Surname
Smith
Robson

استخراج اطلاعات از پایگاه داده

■ اولویت عملگرها

■ اولویت عملگرها در عبارات شرطی به ترتیب زیر است:

نام عملگر
پرانتز ()
NOT
AND
ALL, ANY, BETWEEN, IN, LIKE, OR, SOME

■ مثال: دستور زیر را تفسیر کنید:

```
SELECT State, DateOfJoining  
FROM MemberDetails  
WHERE State = 'New State' OR State = 'Golden State'  
AND DateOfJoining >= '2005-08-01';
```

استخراج اطلاعات از پایگاه داده

■ عملگرهای منطقی

■ عملگر NOT: برای نقض شرط به کار می رود

■ مثال:

```
SELECT FirstName  
FROM MemberDetails  
WHERE NOT State = 'Golden State' ;
```

```
SELECT City  
FROM MemberDetails  
WHERE NOT (City = 'Townsville' OR City = 'Orange Town'  
OR City = 'New Town') ;
```

■ تمرین

استخراج اطلاعات از پایگاه داده

■ عملگرهای منطقی

■ عملگر BETWEEN: برای تعیین محدوده در شرط به کار می رود

■ مثال:

```
SELECT FilmName, Rating  
FROM Films  
WHERE Rating BETWEEN 3 AND 5
```

■ تمرین

استخراج اطلاعات از پایگاه داده

■ عملگرهای منطقی

■ عملگر LIKE: برای تعیین قالب رشته های متنی در شرط به کار می رود

■ نکته: در تعریف قالب رشته در شرط برای یک کاراکتر تکی از _ و برای یک یا چند

کاراکتر از % استفاده می کنیم

■ مثال:

```
SELECT LastName FROM MemberDetails  
WHERE LastName LIKE 'J%';
```

LastName
Jackson
Jones
Jones
Johnson

استخراج اطلاعات از پایگاه داده

عملگرهای منطقی ■

مثال: ■

```
SELECT LastName FROM MemberDetails  
WHERE LastName LIKE 'D__s';
```

LastName
Dales
Doors

استخراج اطلاعات از پایگاه داده

■ عملگرهای منطقی

■ عملگر IN: برای بررسی برابری با مجموعه ای از مقادیر در شرط به کار می رود

■ مثال:

```
SELECT City
FROM MemberDetails
WHERE
City = 'Townsville'
OR
City = 'Windy Village'
OR
City = 'Dover'
OR
City = 'Big City';
```

به جای عبارت بالا از دستور زیر استفاده می کنیم

```
SELECT City
FROM MemberDetails
WHERE
City IN ('Townsville', 'Windy Village', 'Dover', 'Big City');
```

استخراج اطلاعات از پایگاه داده

■ مرتب سازی خروجی:

■ عملگر ORDER BY: برای مرتب سازی مقادیر خروجی به کار می رود

■ نکته: پیش فرض مرتب سازی صعودی (ASC) است و برای نزولی از DESC

استفاده می کنیم

■ مثال:

```
SELECT FilmName, Rating
FROM Films
ORDER BY YearReleased;
```

معادل

```
SELECT YearReleased
FROM Films
ORDER BY YearReleased ASC;
```

برای مرتب سازی نزولی

```
SELECT YearReleased
FROM Films
ORDER BY YearReleased DESC;
```

استخراج اطلاعات از پایگاه داده

■ الحاق ستون ها:

■ عملگر AS: برای تعیین نام مستعار به کار می رود

■ مثال:

```
SELECT FirstName + ' ' + LastName AS FullName FROM  
MemberDetails;
```

FullName
Katie Smith
Susie Simons
John Jackson
Steve Gee
John Jones
Jenny Jones
Jack Johnson
Seymour Botts

استخراج اطلاعات از پایگاه داده

■ الحاق ستون ها:

■ می توان یک رشته را نیز با مقدار ستون الحاق کرد.

■ مثال:

```
SELECT 'First name is ' + FirstName + ', last name is ' +  
LastName FullName
```

```
FROM
```

```
FullName
```

```
First name is Katie, last name is Smith
```

```
First name is Susie, last name is Simons
```

```
First name is John, last name is Jackson
```

```
First name is Steve, last name is Gee
```

```
First name is John, last name is Jones
```

```
First name is Jenny, last name is Jones
```

```
First name is Jack, last name is Johnson
```

```
First name is Seymour, last name is Botts
```

استخراج اطلاعات از پایگاه داده

■ الحاق ستون ها:

■ می توان بیش از یک نام مستعار تعیین نمود

■ مثال:

```
SELECT LastName AS Surname, FirstName AS ChristianName  
FROM MemberDetails;
```

Surname	ChristianName
Smith	Katie
Simons	Susie
Jackson	John
Gee	Steve
Jones	John
Jones	Jenny
Johnson	Jack
Botts	Seymour

استخراج اطلاعات از پایگاه داده

■ الحاق ستون ها:

- اگر نام مستعار حاوی Space یا کاراکترهای دیگری است که نباید در نام ستون یا نام مستعار باشد باید درون براکت قرار گیرد؛

■ مثال:

```
SELECT LastName AS Surname, FirstName AS [Christian Name]  
FROM MemberDetails;
```

دستور زیر خطا دارد:

```
SELECT DateOfBirth AS One***Two FROM MemberDetails;
```

باید به صورت زیر نوشته شود:

```
SELECT DateOfBirth AS One***Two FROM MemberDetails;
```

استخراج اطلاعات از پایگاه داده

■ انتخاب داده ها از بیش از یک جدول:

■ نحوه پیوند جداول:

نام_ستون_جدول ۲ = نام_ستون_جدول ۱ ON INNER JOIN جدول ۲ جدول ۱

■ مثال: نام فیلم، تاریخ انتشار و امتیاز فیلم های تاریخی با کد طبقه فیلم ۶

```
SELECT FilmName, YearReleased, Rating
FROM Films INNER JOIN Category
ON Films.CategoryId = Category.CategoryId
WHERE Category.CategoryId = 6;
```

FilmName	YearReleased	Rating
Sense and Insensitivity	2001	3
15th Late Afternoon	1989	5
Gone with the Window Cleaner	1988	3
The Good, the Bad, and the Facially Challenged	1989	5

استخراج اطلاعات از پایگاه داده

■ انتخاب داده ها از بیش از یک جدول:

■ به جای استفاده از عبارت JOIN می توان شرط تساوی روی ستون های همنام به همراه نام دو جدول استفاده کرد

■ مثال:

```
SELECT FilmName, YearReleased, Rating  
FROM Films, Category  
WHERE Films.CategoryId = Category.CategoryId AND  
Category.CategoryId = 6;
```

FilmName	YearReleased	Rating
Sense and Insensitivity	2001	3
15th Late Afternoon	1989	5
Gone with the Window Cleaner	1988	3
The Good, the Bad, and the Facially Challenged	1989	5

استخراج اطلاعات از پایگاه داده

■ انتخاب داده ها از بیش از یک جدول:

■ برای انتخاب ستون از یک جدول در هنگام پیوند دو یا بیش از دو جدول می توان از عملگر نقطه استفاده نمود

■ مثال:

```
SELECT Category.Category, FavCategory.MemberId  
FROM FavCategory INNER JOIN Category  
ON FavCategory.CategoryId = Category.CategoryId;
```

Category	MemberId
Thriller	3
Thriller	5
Thriller	10
Thriller	3
Romance	1

استخراج اطلاعات از پایگاه داده

- انتخاب داده ها از بیش از یک جدول:
- تمرین ۱: پرس و جویی برای جدول خروجی زیر بنویسید. (دقت کنید که نام خانوادگی ها مرتب شده هستند)

Category	FirstName	LastName
Thriller	Stuart	Dales
War	Stuart	Dales
Historical	Stuart	Dales
Horror	William	Doors
Sci-fi	William	Doors
Sci-fi	Steve	Gee
War	Jamie	Hills
Horror	Jamie	Hills
Sci-fi	Jamie	Hills

استخراج اطلاعات از پایگاه داده

■ انتخاب داده ها از بیش از یک جدول:

■ تمرین ۲: پرس و جویی برای جدول خروجی زیر بنویسید. (دقت کنید که نام خانوادگی و نام ها مرتب شده هستند)

FirstName	LastName	Category	FilmName	YearReleased
Stuart	Dales	Thriller	The Maltese Poodle	1947
Stuart	Dales	Thriller	Raging Bullocks	1980
Stuart	Dales	Thriller	The Life Of Bob	1984
William	Doors	Horror	The Lion, the Witch, and	1989
William	Doors	Horror	Nightmare on Oak Street	1977
Steve	Gee	Sci-fi	The Wide Brimmed Hat	2005
Jamie	Hills	War	The Dirty Half Dozen	1987
Jamie	Hills	War	Planet of the Japes	1967
Jamie	Hills	Horror	Nightmare on Oak Street	1977

استخراج اطلاعات از پایگاه داده

- انتخاب داده ها از بیش از یک جدول:
- تمرین ۳: لیست اعضای که در شهری که باشگاه گردهمایی برگزار میکند زندگی نمی کنند.
- تمرین ۴: لیست اعضای که در گردهمایی ها شرکت می کنند به همراه تاریخ گردهمایی و محل برگزاری آن.

استخراج اطلاعات از پایگاه داده

■ داده NULL:

■ مقدار نامشخص در پایگاه داده با NULL مشخص می شود.

■ مثال: دستور زیر مقدار تاریخ تولد را مشخص نکرده است، بنابراین در ستون مربوط

مقدار NULL قرار می گیرد.

```
INSERT INTO MemberDetails  
(MemberId, FirstName, LastName, Email, DateOfJoining)  
VALUES (15, 'Catherine', 'Hawthorn', 'chawthorn@mailme.org',  
'2005-08-25')
```

■ به طور کلی بهتر است از ورود داده NULL اجتناب کنیم و به جای آن مقادیر پیش فرض قرار داده شود.

■ مثال: اگر یک ستون عددی را ایجاد کرده اید، عددی را به کار ببرید که هرگز در آن وارد نمی شود مانند عدد ۱- برای ستون سن افراد. یا برای ستون متنی، از یک رشته تهی

۲۴ استفاده کنید.