

۵- پرس و جو یا Query

اصطلاح پرس و جو یا Query به این معنی است که دارای اطلاعاتی هستیم که میخواهیم با آنها کار کنیم. مثلا آنها را بخوانیم، تغییر دهیم و یا مواردی را به آن اضافه کنیم. اکثر Queryهایی که استفاده می‌کنیم از چهار حالت خارج نیستند:

SELECT خواندن.

INSERT درج اطلاعات جدید.

UPDATE به هنگام سازی.

DELETE حذف اطلاعات.

اجرای یک Query بدین صورت است:

```
mysql_query(QueryString, ConnectionID);
```

۵-۱- درج داده‌ها در پایگاه داده INSERT :

قبل از کار با پایگاه داده، باید داده‌ها را در آن قرار دهید. در SQL دستور INSERT این کار را انجام می‌دهد. با استفاده از این دستور سطرهای داده را درون پایگاه داده قرار می‌دهید.

شکل عمومی نگارش این دستور به صورت زیر است :

```
INSERT [INTO] table [ (column1,column2,column3,...) ]
VALUES (value1,value2,value3)
```

برای قرار دادن یک سطر در جدول customers از عبارت

```
INSERT INTO users (username,password,userReallyName)
Values ( 'kaveh','222','Mr.ahangar' )
```

مثال ۹-۱

استفاده کنید. همانطور که ملاحظه می‌کنید به جای table, users و به جای value1,value2,value3 معادل‌هایی را جایگزین کرده ایم. توجه داشته باشید که مقادیری که در دستور العمل وارد می‌شوند به ترتیب در ستون‌های جدول قرار می‌گیرند.

Query فوق یک رکورد جدید در جدول users درج میکند که مقدار فیلد اول kaveh ، دومی

۲۲۲ و آخرین آن Mr.ahangar می باشد.

حتی این امکان نیز وجود دارد که اصلا پرانتز اول را ننویسیم. و میتوان با یک query

چندین فیلد ایجاد کرد.

۵-۲- بازبازی داده از پایگاه داده:

بازبازی داده از پایگاه داده توسط دستور SELECT انجام می شود. در واقع سطرهایی را

که با معیار های مشخص شده مطابقت داشته باشند را انتخاب کرده و آنها را بازبازی می

کنیم.

شکل کلی این دستور عبارتست از :

```
SELECT [option] items
[INTO file_details]
FROM tables
[WHERE conditions]
[GROUP BY group_type]
[HAVING where_definiton]
[ORDER BY order_type]
```

۵-۲-۱- بازبازی داده توسط معیارهای خاص

برای دستیابی به زیر مجموعه ای از سطرهای یک جدول , باید معیارهای خاصی را

تعریف کنید. تعریف این معیار توسط دستور WHERE انجام می شود. مثلا کد :

```
Select *
From orders
Where customerid = 5;
```

مثال ۹-۲

همه ستون های جدول orders و سطرهایی با customerid برابر 5 را انتخاب می

کند. خروجی این کد به شکل زیر است :

orderid	customerid	amount	date
1	5	69.98	2000-04-02
4	5	24.99	2000-05-01

عبارت WHERE معیارهای لازم برای انتخاب سطرها را مشخص می کند. در اینجا از نماد

تساوی (=) استفاده کردیم.

۵-۲-۲- بازبازی داده از جدول های چند تایی

اگر بخواهید مشتریهایی که ماه جاری سفارش خرید داشته اند را مشخص کنید باید هر دو جدول orders , Customers را بررسی کنید. اگر بخواهید اقلام سفارشی آنها را نیز داشته باشید باید اطلاعات جدول order_items را هم در اختیار داشته باشید. این اطلاعات در جدول های مختلفی قرار دارند و برای قرار دادن آنها در کنار هم باید جدول ها را تلفیق کرد (join). در MySQL انواع مختلفی از تلفیق وجود دارد.

۵-۲-۲-۱- تلفیق دو جدول

می خواهیم سفارش هایی که Julie Smith در خواست کرده را بررسی کنیم.

```
Select orders.orderid, orders.amount, orders.date
from customers,order s
Where customers.name = 'Julie Smith'
and customers.customerid = orders.customerid
```

مثال ۹-۳

و خروجی این پرسش عبارتست از :

orderid	amount	Date
2	49.99	2000-04-15

۵-۲-۲-۲- تلفیق بیش از دو جدول

تلفیق بیش از دو جدول مشابه تلفیق دو جدول است. تنها کافی است شرط تلفیق دو جدول را مشخص کنید. فرض کنید می‌خواهیم افرادی که کتاب جاوا را سفارش داده‌اند مشخص کنیم. نقطه اشتراک دو جدول customers, orders همان ستون customerid است. برای دسترسی به جدول order_item از طریق جدول orders, از ordered استفاده می‌کنیم. برای دسترسی به جدول BOOKS از طریق جدول order_items از ISBN استفاده می‌کنیم. پس از یافتن این ارتباطات, کتاب‌هایی را که در عنوان آنها از کلمه java استفاده شده را پیدا می‌کنیم و نام سفارش دهنده آنها را تعیین می‌کنیم.

پرسش زیر کلیه عملیات‌های ذکر شده را انجام می‌دهد:

```
Select customers.name
From customers, orders, order_items, books
Where customers.customerid = orders.customerid
and orders.orderid = order_items.orderid
and order_items.isbn = books.isbn
and books.title like '%java%';
```

مثال ۹-۴

خروجی این پرسش برابر است با:

Name
Michelle
Arthur

همانطور که می‌بینید, بین داده‌های سه جدول ارتباطی برقرار می‌کنیم و با استفاده از

سه شرط تلفیق, سه جدول را با هم تلفیق می‌کنیم. تلفیق از نوع متعادل است.

۵-۲-۳- یافتن سطرهای غیر متناظر و غیر مشترک

در مثال مقابل تنها سطرهای مشترک و متناظر در تشکیل جدول نهایی نقش داشتند. گاهی اوقات ممکن است به دنبال سطرهای متناظر بر گردید. مثلا بخواهید مشتری هایی که هیچ سفارشی ندارند یا کتابهایی که هرگز سفارش خرید نداشته اند را مشخص کنید. روش حل چنین مسائلی در MySQL استفاده از تلفیق چپ (left join) است. در این نوع تلفیق، سطرهای متناظر بدست می آیند و اگر هیچ انطباقی بین دو جدول وجود نداشته باشد، سطرهای شامل تصاویر NULL است در ستون های سمت راست اضافه می شود.

```
Select customers.customerid, customers.name, orders.orderid
From customers left join orders
On customers.customerid = orders.customerid;
```

مثال ۹-۵

در تلفیق چپ، شرایط تلفیق را با عبارت ON مشخص می کنیم.

خروجی این پرسش برابر است با:

Customerid	Name	Ordered
1	Mellisa Jones	NULL
2	Michael Archer	NULL
3	Julie Smith	2
4	Alan Wong	3
5	Michelle Arthur	1
6	Michelle Arthur	4

در این خروجی Mellisa Jones و Michael Archer از orderid متناظر بر خوردار نیستند

و orderid آنها با NULL مشخص شده است.

اگر بخواهید مشتری‌هایی که هیچ چیز سفارش نکرده اند، مشاهده کنید، در قسمت راست

جدول NULL های موجود در کلید اصلی (در اینجا ordered) را جستجو کنید:

```
select customers.customerid, customers.name
from customers left join orders
using (customerid)
where orders.orderid is null;
```

مثال ۶-۹

خروجی برابر خواهد بود.

customerid	name
1	Mellisa Jones
2	Michael Archer

در استفاده از using دقت کنید که ستون‌های هر دو جدول باید از نام‌های یکسان

برخوردار باشند.

۵-۲-۴- نام‌های مستعار جدول‌ها

استفاده از نام مستعار (alias) برای جدول‌ها بسیار مفید و سودمند است با استفاده از نام

مستعار، پرسش مربوط به تلفیق بیش از دو جدول را به صورت زیر خلاصه کرده ایم:

```
select c.name
from customers as c, orders as o, order_item as oi, books as b
where c.customerid = o.customerid
and o.orderid = oi.orderid
and oi.isbn = b.isbn
and b.title like '%java%';
```

مثال ۷-۹

با استفاده از عبارت AS نام‌های مستعار به کار رفته در پرسش را تعریف می‌کنید. حتی

می‌توانید برای ستون‌ها نام مستعار در نظر بگیرید.

در هنگام تلفیق یک جدول با خودش از اسم مستعار آن استفاده می شود. مثلا اگر بخواهید سطرهایی از جدول را پیدا کنید که مقادیر آنها مشترک یا برابر است (مثلا مشتری هایی که در یک شهر زندگی می کنند) برای جدول دو نام مستعار متفاوت تعریف می کنید :

```
select c1.name, c2.name, c1.city
from customers as c1, customers as c2
where c1.city = c2.city
and c1.name != c2.name;
```

مثال ۸-۹

در واقع وانمود کرده اید که جدول customers از دو جدول متفاوت C1 و C2 تشکیل شده و تلفیق را به ستون City اعمال کرده اید.

۵-۲-۵- انتخاب سطر

یکی از شرط هایی که با دستور SELECT به کار می رود شرط LIMIT است. این شرط سطرهایی از جدول که در خروجی ظاهر می شوند را مشخص می کند. این شرط از دو پارامتر تشکیل شده است : شماره سطر ابتدایی و تعداد سطر ها در این پرسش نحوه استفاده از LIMIT را می بینید :

```
select name
from customers
limit 2, 3;
```

مثال ۹-۹

در این پرسش , سه سطر از ستون name که از سطر دوم شروع می شود یعنی سطرهای ۲, ۳ و ۴ در خروجی ظاهر می شوند. توجه داشته باشید که LIMIT در استاندارد ANSI SQL تعریف نشده است و استفاده از آن ممکن است باعث شود SQL با تعدادی از RDBMS ها ناسازگار شود.