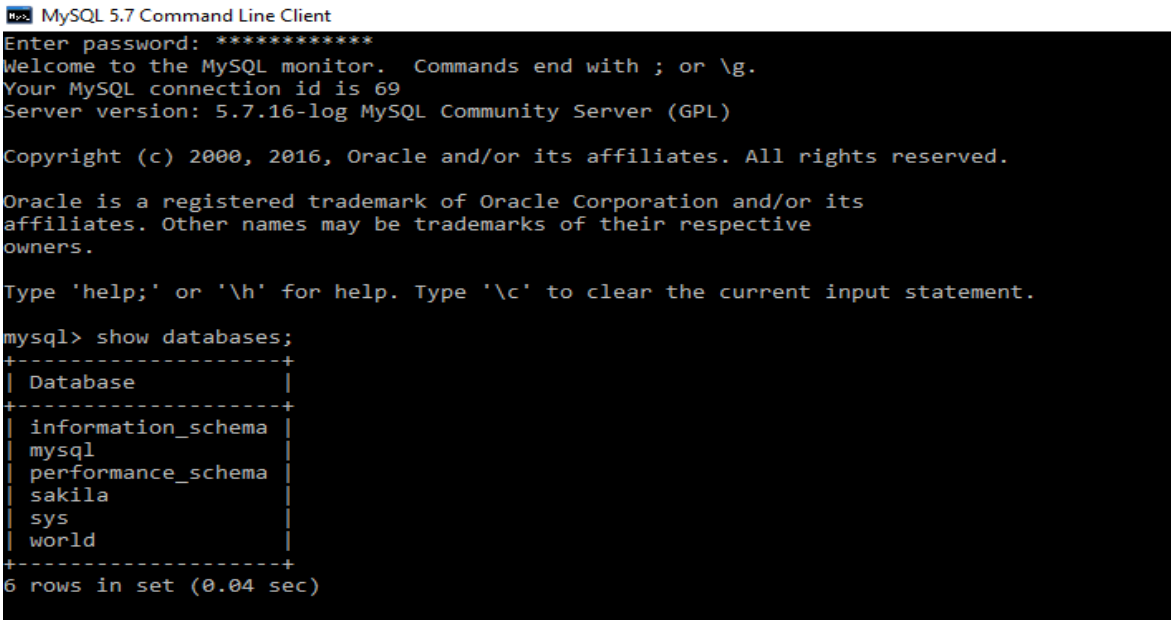
SQL My PRACTICE

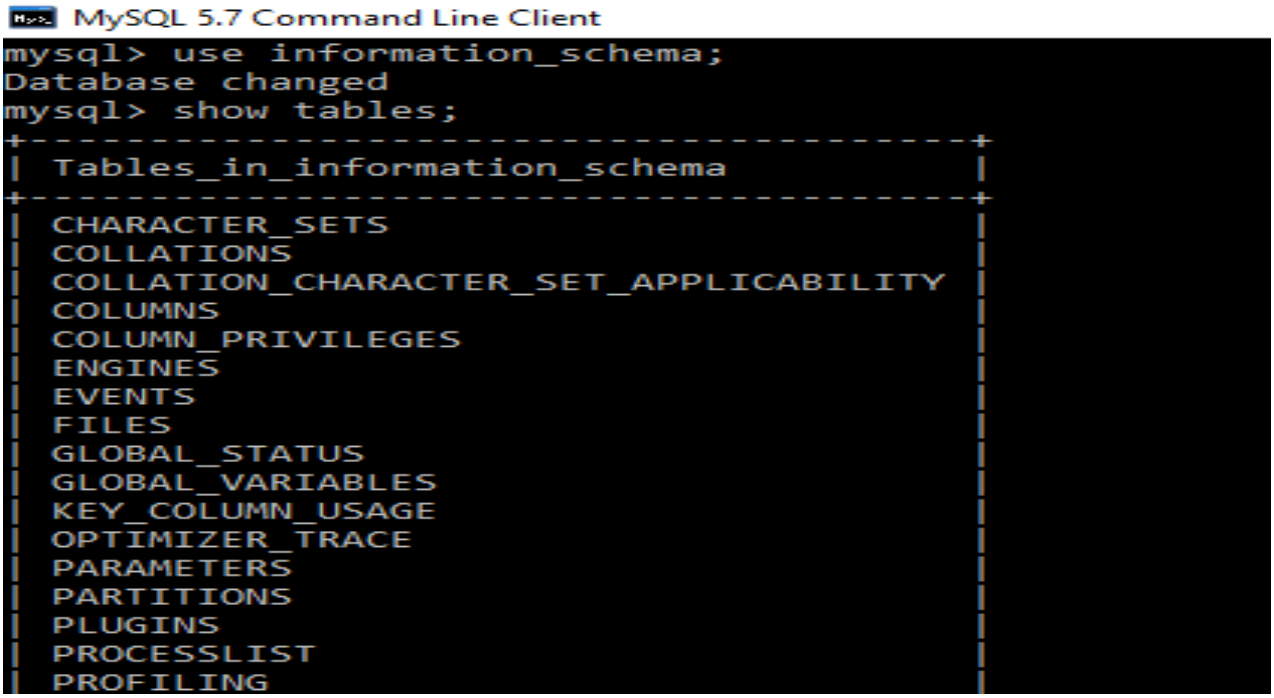
Day I

How to select, choose and use database and tables

Showing databases;



Select database and their inside tables;



Create Database

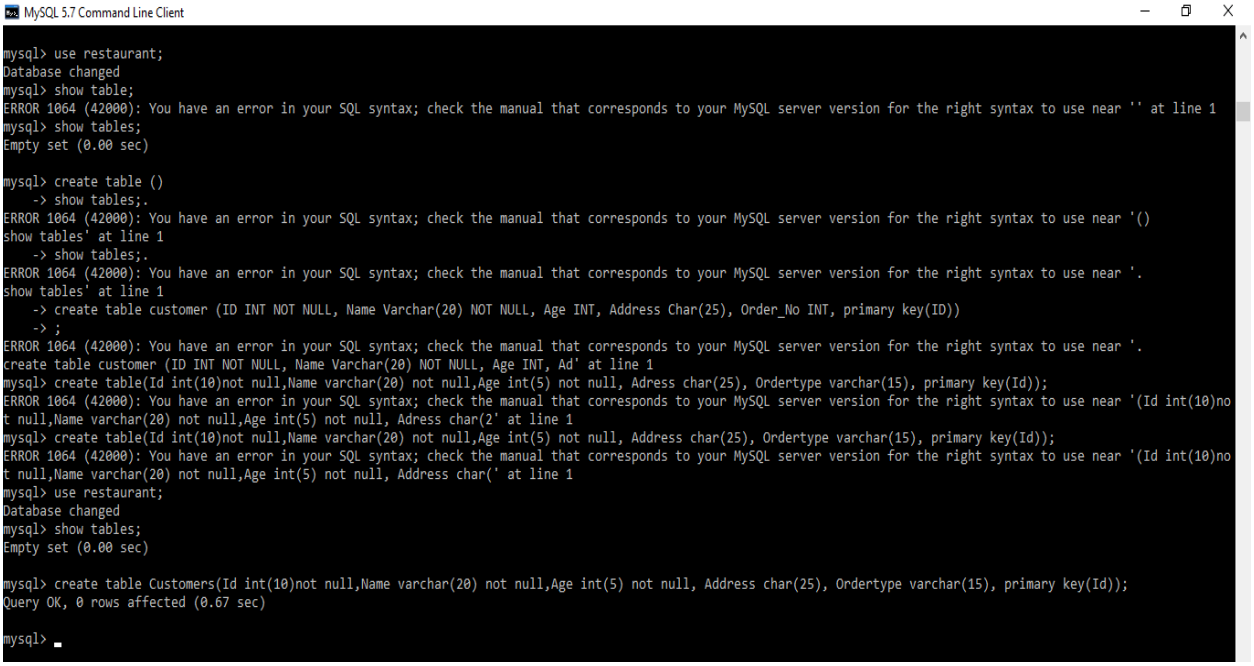
```
mysql> create database test-DB;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manu
1
mysql> create database testDB;
Query OK, 1 row affected (0.07 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sakila             |
| sys                |
| testdb             |
| world              |
+--------------------+
7 rows in set (0.00 sec)

mysql>
```

Drop/delete database:

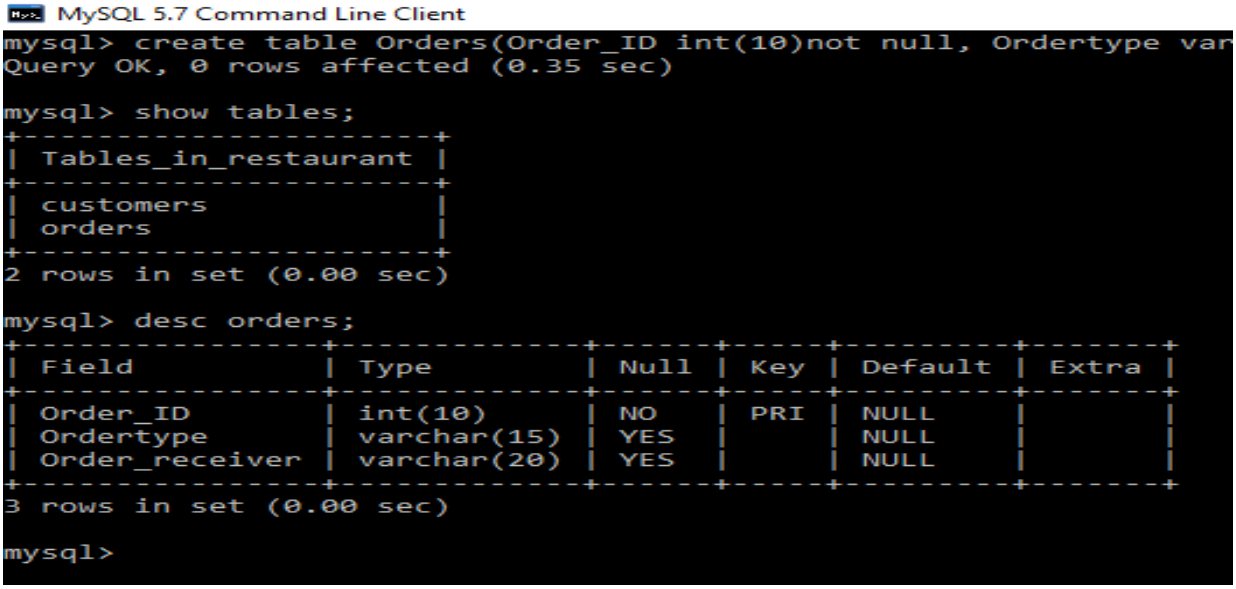
MySQL 5.7 Command Line Client

```
| world                |
+----------------------+
7 rows in set (0.00 sec)

mysql> drop database testdb;
Query OK, 0 rows affected (0.07 sec)

mysql> show databse;
ERROR 1064 (42000): You have an error in your SQL sy
ine 1
mysql> show databases;
+----------------------+
| Database             |
+----------------------+
| information_schema   |
| mysql                |
| performance_schema   |
| sakila               |
| sys                  |
| world                |
+----------------------+
6 rows in set (0.00 sec)
```

How to create a table:



How to view the table:



HOW TO ADD FOREIGN KEY

```
    -> from customers;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'create table
Salary as
from customers' at line 2
mysql> create table Salary as select ID, Salary from customers;
ERROR 1054 (42S22): Unknown column 'Salary' in 'field list'
mysql> create table Employee(Employee_ID int(5) not null, Name varchar(20), Order_ID int(10), primary key(Employee_ID), foreign key(Order_ID));
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ')' at line 1
mysql> create table Employee(Employee_ID int(5) not null, Name varchar(20), Order_ID int(10), primary key(Employee_ID), foreign key(Order_ID) references orders);
ERROR 1215 (HY000): Cannot add foreign key constraint
mysql> create table Employee(Employee_ID int(5) not null, Name varchar(20), Order_ID int(10), primary key(Employee_ID), foreign key(Order_ID) references orders(Order_ID
));
Query OK, 0 rows affected (0.29 sec)

mysql> show tables;
+----------------------+
| Tables_in_restaurant |
+----------------------+
| customers            |
| employee             |
| orders               |
+----------------------+
3 rows in set (0.00 sec)

mysql> desc orders;
+----------------+-------------+------+-----+---------+-------+
| Field          | Type        | Null | Key | Default | Extra |
+----------------+-------------+------+-----+---------+-------+
| Order_ID       | int(10)     | NO   | PRI | NULL    |       |
| Ordertype      | varchar(15) | YES  |     | NULL    |       |
| Order_receiver | varchar(20) | YES  |     | NULL    |       |
+----------------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)

mysql> desc employee;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| Employee_ID | int(5)      | NO   | PRI | NULL    |       |
| Name        | varchar(20) | YES  |     | NULL    |       |
| Order_ID    | int(10)     | YES  | MUL | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

How to insert values in a table

```
mysql> insert into customers values(09, 'Hassan', 25, 'Dhaka', 'Shingara');
Query OK, 1 row affected (0.19 sec)

mysql> show customers;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'customers' at
 line 1
mysql> desc customers;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| Id        | int(10)     | NO   | PRI | NULL    |       |
| Name      | varchar(20) | NO   |     | NULL    |       |
| Age       | int(5)      | NO   |     | NULL    |       |
| Address   | char(25)    | YES  |     | NULL    |       |
| Ordertype | varchar(15) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

```
ustomers values(11, 'Chris', 27, 'Khulna', 'Samucha')' at line 1
mysql> insert into customers values(12, 'Khan', 26, 'Chittagong', 'Puri');
Query OK, 1 row affected (0.07 sec)

mysql> insert into customers values(8, 'Chris', 27, 'Khulna', 'Samucha');
Query OK, 1 row affected (0.05 sec)

mysql> insert into customers values(35, 'Mark', 29, 'Dhaka', 'Nanrooti');
Query OK, 1 row affected (0.08 sec)

mysql> insert into customers values(31, 'Yemmy', 25, 'Sylhet', 'Kabab');
Query OK, 1 row affected (0.06 sec)

mysql> select * from customers;
+----+---------+-----+-------------+------------+
| Id | Name    | Age | Address     | Ordertype |
+----+---------+-----+-------------+------------+
|  8 | Chris   |  27 | Khulna      | Samucha    |
|  9 | Hassan  |  25 | Dhaka       | Shingara   |
| 12 | Khan    |  26 | Chittagong | Puri       |
| 31 | Yemmy   |  25 | Sylhet      | Kabab      |
| 35 | Mark    |  29 | Dhaka       | Nanrooti   |
+----+---------+-----+-------------+------------+
5 rows in set (0.00 sec)

mysql>
```

Show the data inside the table

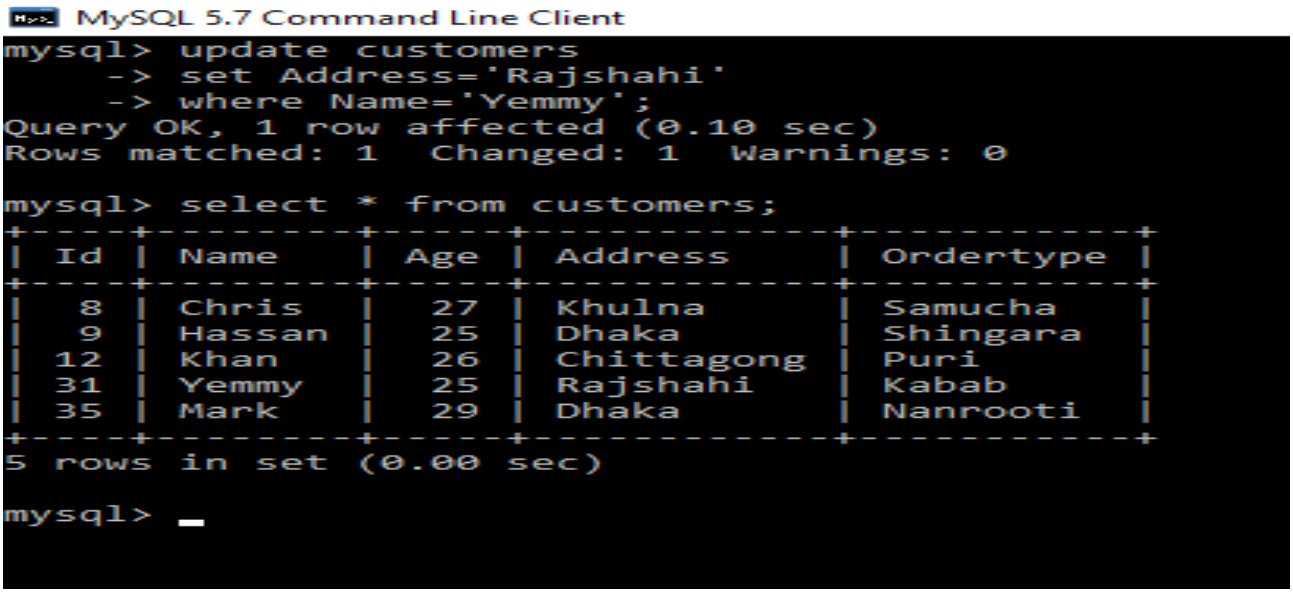
```
mysql> show table customer;
ERROR 1064 (42000): You have an error in your SQL syntax; c
line 1
mysql> show table customers;
ERROR 1064 (42000): You have an error in your SQL syntax; c
 line 1
mysql> select * from customers;
+----+--------+-----+---------+-----------+
| Id | Name   | Age | Address | Ordertype |
+----+--------+-----+---------+-----------+
|  9 | Hassan |  25 | Dhaka   | Shingara  |
+----+--------+-----+---------+-----------+
1 row in set (0.00 sec)
```

```
mysql> select Name, Age from table customers;
ERROR 1064 (42000): You have an error in your SQL syntax; che
rs' at line 1
mysql> select Name, Age from customers;
+---------+-----+
| Name    | Age |
+---------+-----+
| Chris   |  27 |
| Hassan  |  25 |
| Khan    |  26 |
| Yemmy   |  25 |
| Mark    |  29 |
+---------+-----+
5 rows in set (0.00 sec)

mysql> select Id from customers;
+----+
| Id |
+----+
|  8 |
|  9 |
| 12 |
| 31 |
| 35 |
+----+
5 rows in set (0.00 sec)

mysql> select Address from customers;
+-------------+
| Address     |
+-------------+
| Khulna      |
| Dhaka       |
| Chittagong  |
| Sylhet      |
| Dhaka       |
+-------------+
5 rows in set (0.00 sec)
```

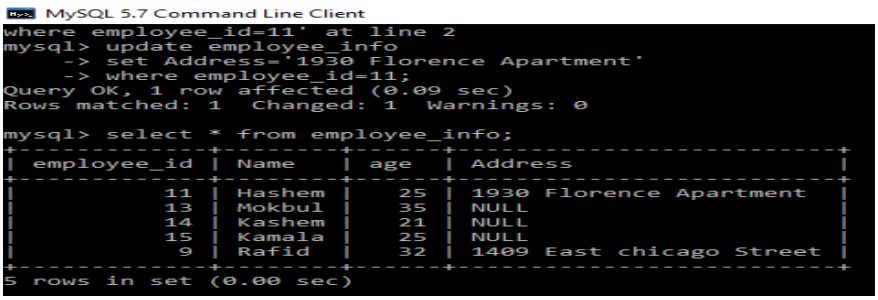
MySQL 5.7 Command Line Client

```
| 35 |
+----+
5 rows in set (0.00 sec)

mysql> select Address from customers;
+-------------+
| Address     |
+-------------+
| Khulna      |
| Dhaka       |
| Chittagong  |
| Sylhet      |
| Dhaka       |
+-------------+
5 rows in set (0.00 sec)

mysql> select Name, Address from customers where Age=25;
+--------+---------+
| Name   | Address |
+--------+---------+
| Hassan | Dhaka   |
| Yemmy  | Sylhet  |
+--------+---------+
2 rows in set (0.00 sec)

mysql>
```

How to update a particular data in a particular table. Here I updated the address of the person whose name was yemmy.



Updatae column



Updating Rows or records

```
mysql> insert into employee values(15,'Marjaan', 52);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`restaurant`.`employ
ERENCES `orders` (`Order_ID`))
mysql> update employee
    -> add Employee_ID=15
    -> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL
ID=15' at line 2
mysql> update Employee_ID
    -> from employee
    -> add Employee_ID=15;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL

add Employee_ID=15' at line 2
mysql> select * from orders;
+----------+-----------+----------------+
| Order_ID | Ordertype | Order_receiver |
+----------+-----------+----------------+
|       48 | Samucha   | Hashem         |
|       56 | Kabab     | Mokhles        |
|       67 | Puri      | Mokbul         |
|       89 | Shingara  | Kashem         |
+----------+-----------+----------------+
4 rows in set (0.07 sec)

mysql> insert into orders values(99, 'Biryani', 'Raju');
Query OK, 1 row affected (0.05 sec)
```

*** notes: you cannot change or update a child row or records, You have to modify the parent row or records at first then modify the child records.

```
3 rows in set (0.03 sec)

mysql> insert into employee values(15, 'kamal', 52);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`restaurant`.`employee`, CONSTRAINT `employee_ibfk_1` FOREIGN KEY (`Order_ID`) REF
ERENCES `orders` (`Order_ID`))
mysql> insert into employee values(15, 'kamal');
ERROR 1136 (21S01): Column count doesn't match value count at row 1
mysql> insert into employee values(15, 'kamal', 99);
Query OK, 1 row affected (0.05 sec)

mysql> select * from employee;
+-------------+--------+----------+
| Employee_ID | Name   | Order_ID |
+-------------+--------+----------+
|           9 | Mokhles |       56 |
|          11 | Hashem  |       89 |
|          13 | Mokbul  |       67 |
|          14 | Kashem  |       48 |
|          15 | kamal   |       99 |
+-------------+--------+----------+
5 rows in set (0.00 sec)
```

Here you can see that we change the data of a child row after updating the parent row. The mistake we made was we tried to change a different or new values which is not right. When insert any values in a child table or row always put the same value of the column which we have changed in the parent row or records.

Where clause with range

```
mysql> select * from customers
    -> where age>20 and age<27;
+----+--------+-----+-----------+-----------+
| Id | Name   | Age | Address   | Ordertype |
+----+--------+-----+-----------+-----------+
|  9 | Hassan |  25 | Dhaka     | Shingara  |
| 12 | Khan   |  26 | Chittagong| Puri      |
| 31 | Yemmy  |  25 | Rajshahi  | Kabab     |
+----+--------+-----+-----------+-----------+
3 rows in set (0.00 sec)

mysql>
```

Delete query

MySQL 5.7 Command Line Client

```
mysql> delete from customers
    -> where Id=35;
Query OK, 1 row affected (0.13 sec)

mysql> select * from customers;
+----+--------+-----+-----------+-----------+
| Id | Name   | Age | Address   | Ordertype |
+----+--------+-----+-----------+-----------+
|  8 | Chris  |  27 | Khulna    | Samucha   |
|  9 | Hassan |  25 | Dhaka     | Shingara  |
| 12 | Khan   |  26 | Chittagong| Puri      |
| 31 | Yemmy  |  25 | Rajshahi  | Kabab     |
+----+--------+-----+-----------+-----------+
4 rows in set (0.00 sec)

mysql>
```

Like Clause

MySQL 5.7 Command Line Client

```
mysql> select * from customers
    -> where Age like '25%';
+----+---------+-----+-----------+-------------+
| Id | Name    | Age | Address   | Ordertype   |
+----+---------+-----+-----------+-------------+
|  9 | Hassan  |  25 | Dhaka     | Shingara    |
| 31 | Yemmy   |  25 | Rajshahi  | Kabab       |
+----+---------+-----+-----------+-------------+
2 rows in set (0.03 sec)

mysql> select * from customers
    -> where Age like '20%';
Empty set (0.00 sec)

mysql> select * from customers
    -> where age>20 and age<27;
+----+---------+-----+-------------+-------------+
| Id | Name    | Age | Address     | Ordertype   |
+----+---------+-----+-------------+-------------+
|  9 | Hassan  |  25 | Dhaka       | Shingara    |
| 12 | Khan    |  26 | Chittagong  | Puri        |
| 31 | Yemmy   |  25 | Rajshahi    | Kabab       |
+----+---------+-----+-------------+-------------+
3 rows in set (0.00 sec)

mysql> _
```

```
mysql> select * from customers
    -> where age>20 and age<27;
+----+---------+-----+-------------+-------------+
| Id | Name    | Age | Address     | Ordertype   |
+----+---------+-----+-------------+-------------+
|  9 | Hassan  |  25 | Dhaka       | Shingara    |
| 12 | Khan    |  26 | Chittagong  | Puri        |
| 31 | Yemmy   |  25 | Rajshahi    | Kabab       |
+----+---------+-----+-------------+-------------+
3 rows in set (0.00 sec)

mysql> select * from customers
    -> where Age Like '%25';
+----+---------+-----+-----------+-------------+
| Id | Name    | Age | Address   | Ordertype   |
+----+---------+-----+-----------+-------------+
|  9 | Hassan  |  25 | Dhaka     | Shingara    |
| 31 | Yemmy   |  25 | Rajshahi  | Kabab       |
+----+---------+-----+-----------+-------------+
2 rows in set (0.00 sec)

mysql> select * from customers
    -> where Name like 'H%';
+----+---------+-----+-----------+-------------+
| Id | Name    | Age | Address   | Ordertype   |
+----+---------+-----+-----------+-------------+
|  9 | Hassan  |  25 | Dhaka     | Shingara    |
+----+---------+-----+-----------+-------------+
1 row in set (0.00 sec)
```

MySQL 5.7 Command Line Client

```
mysql> select * from customers
    -> where ordertype like 'S%';
+----+---------+-----+-----------+-------------+
| Id | Name    | Age | Address   | Ordertype   |
+----+---------+-----+-----------+-------------+
|  8 | Chris   |  27 | Khulna    | Samucha     |
|  9 | Hassan  |  25 | Dhaka     | Shingara    |
+----+---------+-----+-----------+-------------+
2 rows in set (0.00 sec)

mysql> select * from customers
    -> where ordertype like '%S';
Empty set (0.00 sec)

mysql> select * from customers
    -> where ordertype like 'a%';
Empty set (0.00 sec)

mysql> select * from customers
    -> where ordertype like '%a';
+----+---------+-----+-----------+-------------+
| Id | Name    | Age | Address   | Ordertype   |
+----+---------+-----+-----------+-------------+
|  8 | Chris   |  27 | Khulna    | Samucha     |
|  9 | Hassan  |  25 | Dhaka     | Shingara    |
+----+---------+-----+-----------+-------------+
2 rows in set (0.00 sec)
```

```
mysql> select * from customers
    -> where Address like '%h%';
+----+--------+-----+-------------+------------+
| Id | Name   | Age | Address     | Ordertype  |
+----+--------+-----+-------------+------------+
|  8 | Chris  |  27 | Khulna      | Samucha    |
|  9 | Hassan |  25 | Dhaka       | Shingara   |
| 12 | Khan   |  26 | Chittagong  | Puri       |
| 31 | Yemmy  |  25 | Rajshahi    | Kabab      |
+----+--------+-----+-------------+------------+
4 rows in set (0.00 sec)

mysql> select * from customers
    -> where Age like '%2%';
+----+--------+-----+-------------+------------+
| Id | Name   | Age | Address     | Ordertype  |
+----+--------+-----+-------------+------------+
|  8 | Chris  |  27 | Khulna      | Samucha    |
|  9 | Hassan |  25 | Dhaka       | Shingara   |
| 12 | Khan   |  26 | Chittagong  | Puri       |
| 31 | Yemmy  |  25 | Rajshahi    | Kabab      |
+----+--------+-----+-------------+------------+
4 rows in set (0.00 sec)

mysql> select * from customers
    -> where Age like '2%';
+----+--------+-----+-------------+------------+
| Id | Name   | Age | Address     | Ordertype  |
+----+--------+-----+-------------+------------+
|  8 | Chris  |  27 | Khulna      | Samucha    |
|  9 | Hassan |  25 | Dhaka       | Shingara   |
| 12 | Khan   |  26 | Chittagong  | Puri       |
| 31 | Yemmy  |  25 | Rajshahi    | Kabab      |
+----+--------+-----+-------------+------------+
4 rows in set (0.00 sec)

mysql> select * from customers
    -> where Age like '%5';
+----+--------+-----+-----------+------------+
| Id | Name   | Age | Address   | Ordertype  |
+----+--------+-----+-----------+------------+
|  9 | Hassan |  25 | Dhaka     | Shingara   |
| 31 | Yemmy  |  25 | Rajshahi  | Kabab      |
+----+--------+-----+-----------+------------+
```

Here is the example of Like clause, where I wanted to know any character which second alphabet is a. That's why I have used only one underscore and the % means the first alphabet starts with a.

MySQL 5.7 Command Line Client

```
mysql> select * from customers
    -> where Ordertype like'_K_';
Empty set (0.00 sec)

mysql> select * from customers
    -> where Ordertype like'_a%';
+----+-------+-----+----------+-----------+
| Id | Name  | Age | Address  | Ordertype |
+----+-------+-----+----------+-----------+
|  8 | Chris |  27 | Khulna   | Samucha   |
| 31 | Yemmy |  25 | Rajshahi | Kabab     |
+----+-------+-----+----------+-----------+
2 rows in set (0.00 sec)
```

Same example byt different result when I used address insteda of ordertype.

MySQL 5.7 Command Line Client

```
2 rows in set (0.00 sec)

mysql> select * from customers
    -> where Address like '_a%';
+----+-------+-----+----------+-----------+
| Id | Name  | Age | Address  | Ordertype |
+----+-------+-----+----------+-----------+
| 31 | Yemmy |  25 | Rajshahi | Kabab     |
+----+-------+-----+----------+-----------+
1 row in set (0.00 sec)
```

MySQL 5.7 Command Line Client

```
mysql> select * from customers
    -> where Ordertype like 'P__i';
+----+------+-----+-------------+------------+
| Id | Name | Age | Address     | Ordertype |
+----+------+-----+-------------+------------+
| 12 | Khan |  26 | Chittagong | Puri       |
+----+------+-----+-------------+------------+
1 row in set (0.00 sec)

mysql> select * from customers
    -> where Address like 'D__%_%';
+----+--------+-----+---------+------------+
| Id | Name   | Age | Address | Ordertype |
+----+--------+-----+---------+------------+
|  9 | Hassan |  25 | Dhaka   | Shingara   |
+----+--------+-----+---------+------------+
1 row in set (0.00 sec)

mysql> select * from customers
    -> where Ordertype like 'D___a';
Empty set (0.00 sec)

mysql> select * from customers
    -> where Address like 'D___a';
+----+--------+-----+---------+------------+
| Id | Name   | Age | Address | Ordertype |
+----+--------+-----+---------+------------+
|  9 | Hassan |  25 | Dhaka   | Shingara   |
+----+--------+-----+---------+------------+
1 row in set (0.00 sec)

mysql>
```

This is an example where I wanted to the third character of any address is a. That's why I have used two underscore.

```
mysql> select * from customers
    -> where Address like '__a%';
+----+--------+-----+---------+-----------+
| Id | Name   | Age | Address | Ordertype |
+----+--------+-----+---------+-----------+
|  9 | Hassan |  25 | Dhaka   | Shingara  |
+----+--------+-----+---------+-----------+
1 row in set (0.00 sec)

mysql>
```

MySQL 5.7 Command Line Client

```
mysql> select * from customers
    -> where Address like 'D___a';
+----+--------+-----+---------+-----------+
| Id | Name   | Age | Address | Ordertype |
+----+--------+-----+---------+-----------+
|  9 | Hassan |  25 | Dhaka   | Shingara  |
+----+--------+-----+---------+-----------+
1 row in set (0.00 sec)
```

Top/Limit clause: yo choose the top n numbers of row/record

```
mysql> select * from customers
    -> limit 3
    -> where Age>=25;
ERROR 1064 (42000): You have an error in your SQL
' at line 3
mysql> select * from customers
    -> limit 3;
+----+--------+-----+-------------+-----------+
| Id | Name   | Age | Address     | Ordertype |
+----+--------+-----+-------------+-----------+
|  8 | Chris  |  27 | Khulna      | Samucha   |
|  9 | Hassan |  25 | Dhaka       | Shingara  |
| 12 | Khan   |  26 | Chittagong | Puri      |
+----+--------+-----+-------------+-----------+
3 rows in set (0.00 sec)

mysql> select * from customers
    -> limit 2;
+----+--------+-----+---------+-----------+
| Id | Name   | Age | Address | Ordertype |
+----+--------+-----+---------+-----------+
|  8 | Chris  |  27 | Khulna  | Samucha   |
|  9 | Hassan |  25 | Dhaka   | Shingara  |
+----+--------+-----+---------+-----------+
2 rows in set (0.00 sec)
```

MySQL 5.7 Command Line Client

```
mysql> select Name from customers
    -> limit 3;
+--------+
| Name   |
+--------+
| Chris  |
| Hassan |
| Khan   |
+--------+
3 rows in set (0.00 sec)
```

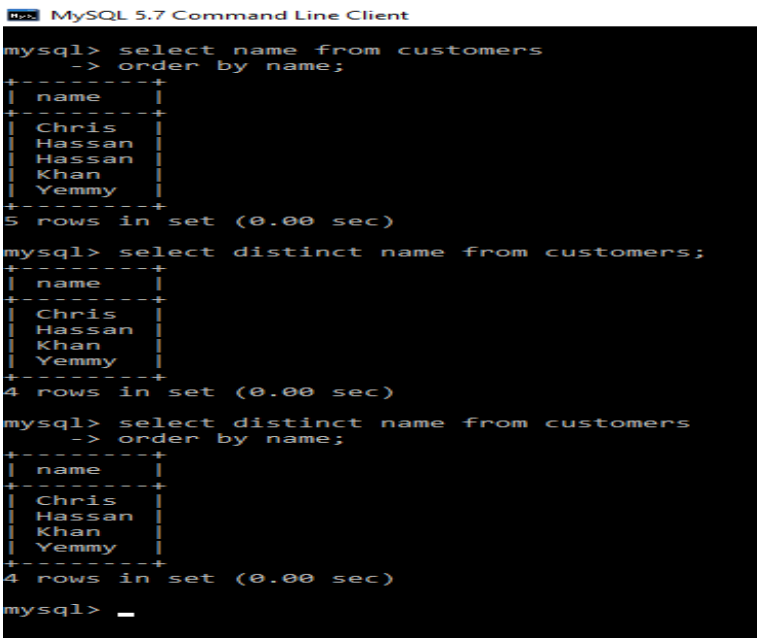
Order by clause: It is used to sort out data in ascending order or descending order

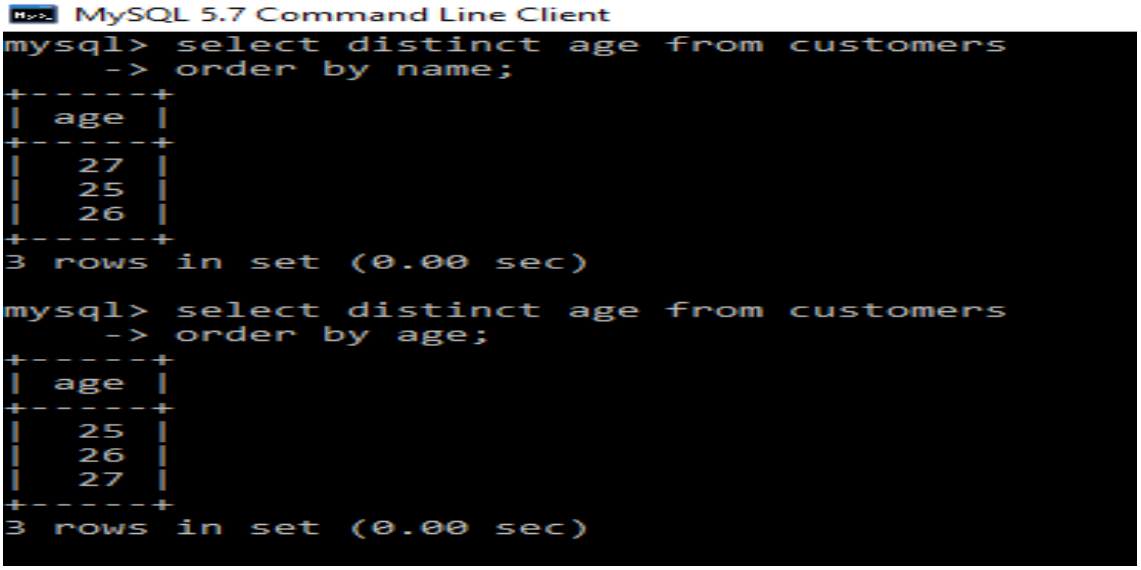
```
mysql> select * from customers
    -> order by Age;
+----+--------+-----+-------------+-----------+
| Id | Name   | Age | Address     | Ordertype |
+----+--------+-----+-------------+-----------+
|  9 | Hassan |  25 | Dhaka       | Shingara  |
| 31 | Yemmy  |  25 | Rajshahi    | Kabab     |
| 12 | Khan   |  26 | Chittagong  | Puri      |
|  8 | Chris  |  27 | Khulna      | Samucha   |
+----+--------+-----+-------------+-----------+
4 rows in set (0.03 sec)

mysql> select * from customers
    -> order by Age desc;
+----+--------+-----+-------------+-----------+
| Id | Name   | Age | Address     | Ordertype |
+----+--------+-----+-------------+-----------+
|  8 | Chris  |  27 | Khulna      | Samucha   |
| 12 | Khan   |  26 | Chittagong  | Puri      |
|  9 | Hassan |  25 | Dhaka       | Shingara  |
| 31 | Yemmy  |  25 | Rajshahi    | Kabab     |
+----+--------+-----+-------------+-----------+
4 rows in set (0.00 sec)
```

Group by clause

```
mysql> select Name, sum(age)
    -> from customers
    -> group by name;
+---------+-----------+
| Name    | sum(age)  |
+---------+-----------+
| Chris   |        27 |
| Hassan  |        52 |
| Khan    |        26 |
| Yemmy   |        25 |
+---------+-----------+
4 rows in set (0.00 sec)

mysql> select Name, sum(age)
    -> from customers
    -> group by Age;
+---------+-----------+
| Name    | sum(age)  |
+---------+-----------+
| Hassan  |        50 |
| Khan    |        26 |
| Chris   |        54 |
+---------+-----------+
3 rows in set (0.00 sec)

mysql> select age, sum(Name)
    -> from customers
    -> group by age;
+-----+------------+
| age | sum(Name)  |
+-----+------------+
|  25 |          0 |
|  26 |          0 |
|  27 |          0 |
+-----+------------+
3 rows in set, 5 warnings (0.00 sec)
```

Distinct clause: it only choose on unique values and remove the duplicates.





Constraints: Constraints are used to limit the fault in the data inside the table. Constraints are two types table level and column level. Column levels are applied only to one column table level constraint are applied to the whole table.

1. Not Null constraints: a data cannot be empty (Null means unknown or vacant).

2. Default Constraints: it makes the data of any tables default

```
mysql> create table wages(employee_name varchar(20) not null, employee_ID int(5) not null, Salary decimal(10, 2) default 5000.00, foreign key(Employee_ID) references em
ployee(Employee_ID));
Query OK, 0 rows affected (0.44 sec)

mysql> show tables;
+----------------------+
| Tables_in_restaurant |
+----------------------+
| customers            |
| employee             |
| orders               |
| wages                |
+----------------------+
4 rows in set (0.00 sec)

mysql> desc wages;
+---------------+--------------+------+-----+---------+-------+
| Field         | Type         | Null | Key | Default | Extra |
+---------------+--------------+------+-----+---------+-------+
| employee_name | varchar(20)  | NO   |     | NULL    |       |
| employee_ID   | int(5)       | NO   | MUL | NULL    |       |
| Salary        | decimal(10,2)| YES  |     | 5000.00 |       |
+---------------+--------------+------+-----+---------+-------+
3 rows in set (0.00 sec)

mysql> select & from wages;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '& from wages'
 at line 1
mysql> select * from wages;
Empty set (0.00 sec)

mysql>
```

Check constraints with create table

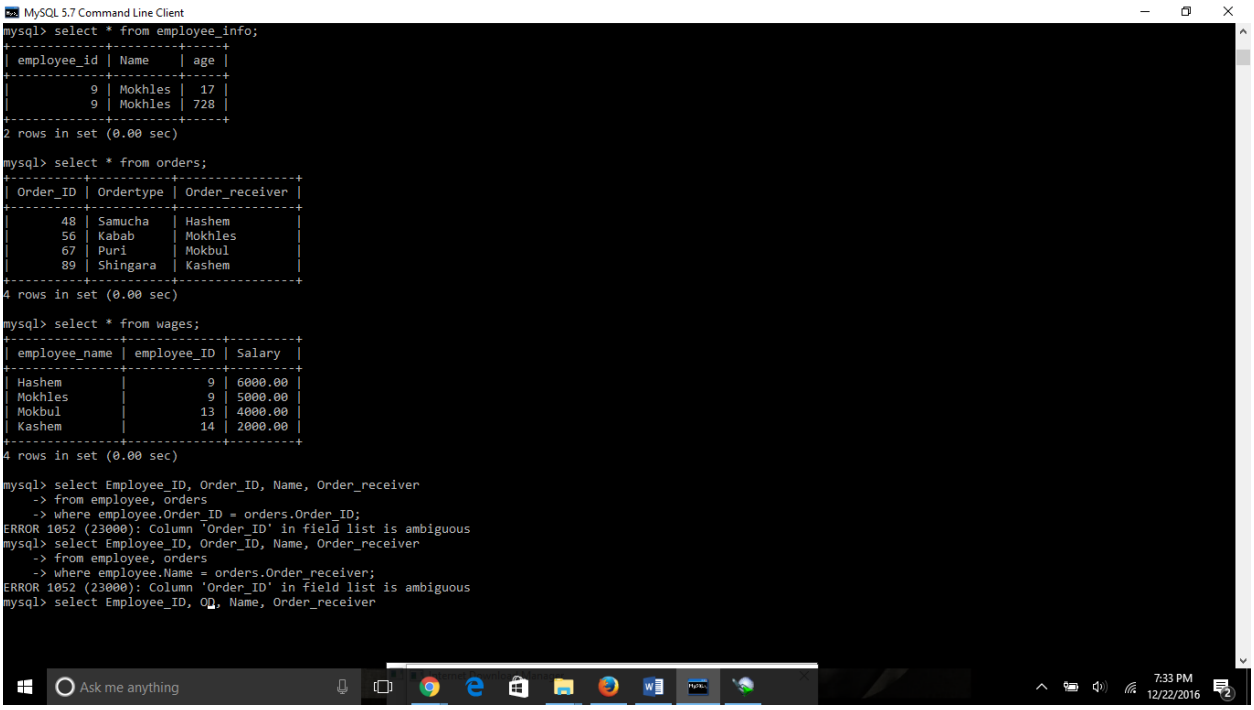
```
mysql> create table employeeinfo(Employee_ID int(5), Name varchar(20), age int(5) not null check(age>=18), foreign key (Employee_ID) references employee(Employee_ID));
Query OK, 0 rows affected (0.47 sec)
```

```
Select MySQL 5.7 Command Line Client
mysql> insert into employee_info values(9,'Mokhles', 728);
Query OK, 1 row affected (0.07 sec)

mysql> select * from employee_info;
+-------------+---------+-----+
| employee_id | Name    | age |
+-------------+---------+-----+
|           9 | Mokhles |  17 |
|           9 | Mokhles | 728 |
+-------------+---------+-----+
2 rows in set (0.00 sec)

mysql> create index idx_age
    -> on customers;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 2
mysql> create index idx_age
    -> on customers(age);
Query OK, 0 rows affected (0.50 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> create index idx_name
    -> on customers(name);
Query OK, 0 rows affected (0.30 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> create index idx_Name
    -> on customers(Name);
ERROR 1061 (42000): Duplicate key name 'idx_Name'
mysql> create index idx_Address
    -> on Customers(Address);
Query OK, 0 rows affected (0.36 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql>
```

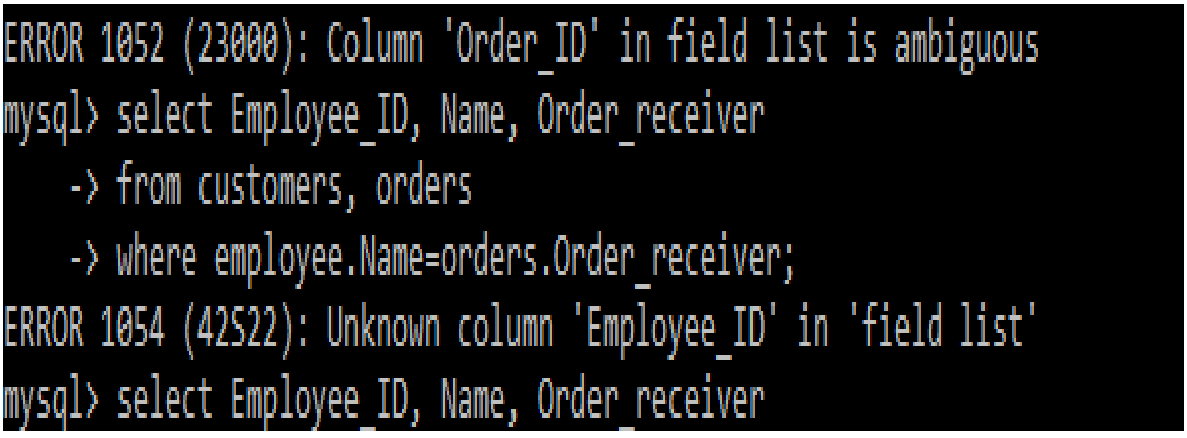
SQL join

Sql join is used to combine two or more tables with the same values.

Following is example of the error which shows the ambiguous of the order_id



Another mistake



Because I choose the wrong name of the table in the join syntax. The ambigious mistake occurs when we put wrong values in the same field but in different tables.

Here I choose thw right syntax

```
    -> where employee.Name=orders.Order_receiver;
ERROR 1054 (42S22): Unknown column 'Employee_ID' in 'field list'
mysql> select Employee_ID, Name, Order_receiver
    -> from employee, orders
    -> where employee.Order_ID = orders.Order_ID;
+-------------+---------+----------------+
| Employee_ID | Name    | Order_receiver |
+-------------+---------+----------------+
|          14 | Kashem  | Hashem         |
|           9 | Mokhles | Mokhles        |
|          13 | Mokbul  | Mokbul         |
|          11 | Hashem  | Kashem         |
+-------------+---------+----------------+
4 rows in set (0.06 sec)

mysql> select * from employee;
+-------------+---------+----------+
| Employee_ID | Name    | Order_ID |
+-------------+---------+----------+
|           9 | Mokhles |       56 |
|          11 | Hashem  |       89 |
|          13 | Mokbul  |       67 |
|          14 | Kashem  |       48 |
+-------------+---------+----------+
4 rows in set (0.00 sec)

mysql> select * from orders;
+----------+-----------+----------------+
| Order_ID | Ordertype | Order_receiver |
+----------+-----------+----------------+
|       48 | Samucha   | Hashem         |
|       56 | Kabab     | Mokhles        |
|       67 | Puri      | Mokbul         |
|       89 | Shingara  | Kashem         |
+----------+-----------+----------------+
4 rows in set (0.00 sec)
```

Inner join : there is less difference between inner join and simple equal join

```
mysql> select Employee_ID, Name, Order_receiver
    -> from employee
    -> inner join orders
    -> on employee.order_id=orders.order_id;
+-------------+---------+----------------+
| Employee_ID | Name    | Order_receiver |
+-------------+---------+----------------+
|          14 | Kashem  | Hashem         |
|           9 | Mokhles | Mokhles        |
|          13 | Mokbul  | Mokbul         |
|          11 | Hashem  | Kashem         |
+-------------+---------+----------------+
4 rows in set (0.00 sec)

mysql> select Employee_ID, Name, Order_receiver
    -> from orders
    -> inner join employee
    -> on employee.order_id=orders.order_id;
+-------------+---------+----------------+
| Employee_ID | Name    | Order_receiver |
+-------------+---------+----------------+
|           9 | Mokhles | Mokhles        |
|          11 | Hashem  | Kashem         |
|          13 | Mokbul  | Mokbul         |
|          14 | Kashem  | Hashem         |
+-------------+---------+----------------+
4 rows in set (0.00 sec)

mysql>
```

Left join: when there is no match it also show values

MySQL 5.7 Command Line Client

```
mysql> select ID, Age, Address, Order_Id
    -> from customers
    -> left join employee
    -> on customers.Name=employee.Name;
+----+-----+------------+----------+
| ID | Age | Address    | Order_Id |
+----+-----+------------+----------+
|  8 |  27 | Khulna     |     NULL |
|  9 |  25 | Dhaka      |     NULL |
| 12 |  26 | Chittagong |     NULL |
| 31 |  25 | Rajshahi   |     NULL |
| 35 |  27 | Dhaka      |     NULL |
+----+-----+------------+----------+
5 rows in set (0.03 sec)

mysql>
```

Right join:

MySQL 5.7 Command Line Client

```
4 rows in set (0.00 sec)

mysql> select id, address, ordertype
    -> from customers
    -> right join employee
    -> on customers.Name=employee.Name;
+------+---------+-----------+
| id   | address | ordertype |
+------+---------+-----------+
| NULL | NULL    | NULL      |
| NULL | NULL    | NULL      |
| NULL | NULL    | NULL      |
| NULL | NULL    | NULL      |
+------+---------+-----------+
4 rows in set (0.00 sec)

mysql>
```

Full Join: as we are working on My SQL instead of SQL we have to use “UNION ALL” like below

```
4 rows in set (0.00 sec)

mysql> select id, address, ordertype
    -> from customers
    -> full join employee
    -> on customers.Name=employee.Name;
ERROR 1054 (42S22): Unknown column 'customers.Name' in 'on clause'
mysql> select id, address, ordertype
    -> from customers
    -> left join employee
    -> on customers.Name=employee.Name
    -> union all
    -> select id, address, ordertype
    -> from customers
    -> right join employee
    -> on customers.Name=employee.Name;
+------+------------+-----------+
| id   | address    | ordertype |
+------+------------+-----------+
|    8 | Khulna     | Samucha   |
|    9 | Dhaka      | Shingara  |
|   12 | Chittagong | Puri      |
|   31 | Rajshahi   | Kabab     |
|   35 | Dhaka      | Puri      |
| NULL | NULL       | NULL      |
| NULL | NULL       | NULL      |
| NULL | NULL       | NULL      |
| NULL | NULL       | NULL      |
+------+------------+-----------+
9 rows in set (0.06 sec)

mysql>
```

Union Clause: difference between union and union clause

```
mysql> select employee_id, Name, Order_receiver, ordertype
    -> from employee
    -> left join orders
    -> on employee.order_id=orders.order_id
    -> union
    -> select employee_id, Name, Order_receiver, ordertype
    -> from employee
    -> right join orders
    -> on employee.order_id=orders.order_id
    -> ;
+-------------+---------+----------------+------------+
| employee_id | Name    | Order_receiver | ordertype  |
+-------------+---------+----------------+------------+
|          14 | Kashem  | Hashem         | Samucha    |
|           9 | Mokhles | Mokhles        | Kabab      |
|          13 | Mokbul  | Mokbul         | Puri       |
|          11 | Hashem  | Kashem         | Shingara   |
+-------------+---------+----------------+------------+
4 rows in set (0.00 sec)

mysql> select employee_id, Name, Order_receiver, ordertype
    -> from employee
    -> left join orders
    -> on employee.order_id=orders.order_id
    -> union all
    -> select employee_id, Name, Order_receiver, ordertype
    -> from employee
    -> right join orders
    -> on employee.order_id=orders.order_id;
+-------------+---------+----------------+------------+
| employee_id | Name    | Order_receiver | ordertype  |
+-------------+---------+----------------+------------+
|          14 | Kashem  | Hashem         | Samucha    |
|           9 | Mokhles | Mokhles        | Kabab      |
|          13 | Mokbul  | Mokbul         | Puri       |
|          11 | Hashem  | Kashem         | Shingara   |
|           9 | Mokhles | Mokhles        | Kabab      |
|          11 | Hashem  | Kashem         | Shingara   |
|          13 | Mokbul  | Mokbul         | Puri       |
|          14 | Kashem  | Hashem         | Samucha    |
+-------------+---------+----------------+------------+
8 rows in set (0.00 sec)
```

Intersect and except clause: in MY sql intersect and except does not work

```
8 rows in set (0.00 sec)

mysql> select employee_id, Name, Order_receiver, ordertype
    -> from employee
    -> left join orders
    -> on employee.order_id=orders.order_id
    -> intersect
    -> select employee_id, Name, Order_receiver, ordertype
    -> from employee
    -> right join orders
    -> on employee.order_id=orders.order_id;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'intersect
select employee_id, Name, Order_receiver, ordertype
from employee
righ' at line 5
mysql> select employee_id, Name, Order_receiver, ordertype
    -> from employee
    -> left join orders
    -> on employee.order_id=orders.order_id
    -> except
    -> select employee_id, Name, Order_receiver, ordertype
    -> from employee
    -> right join orders
    -> on employee.order_id=orders.order_id;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'except
select employee_id, Name, Order_receiver, ordertype
from employee
right j' at line 5
mysql>
```

Null values:

```
+----+----------+-----+-----------+----------+
| ID | NAME     | AGE | ADDRESS   | SALARY   |
+----+----------+-----+-----------+----------+
|  1 | Ramesh   |  32 | Ahmedabad |  2000.00 |
|  2 | Khilan   |  25 | Delhi     |  1500.00 |
|  3 | kaushik  |  23 | Kota      |  2000.00 |
|  4 | Chaitali |  25 | Mumbai    |  6500.00 |
|  5 | Hardik   |  27 | Bhopal    |  8500.00 |
|  6 | Komal    |  22 | MP        |          |
|  7 | Muffy    |  24 | Indore    |          |
+----+----------+-----+-----------+----------+
```

Now, following is the usage of **IS NOT NULL** operator:

```
SQL> SELECT  ID, NAME, AGE, ADDRESS, SALARY
     FROM CUSTOMERS
     WHERE SALARY IS NOT NULL;
```

This would produce the following result:

```
+----+----------+-----+-----------+----------+
| ID | NAME     | AGE | ADDRESS   | SALARY   |
+----+----------+-----+-----------+----------+
|  1 | Ramesh   |  32 | Ahmedabad |  2000.00 |
|  2 | Khilan   |  25 | Delhi     |  1500.00 |
|  3 | kaushik  |  23 | Kota      |  2000.00 |
|  4 | Chaitali |  25 | Mumbai    |  6500.00 |
|  5 | Hardik   |  27 | Bhopal    |  8500.00 |
+----+----------+-----+-----------+----------+
```

Now, following is the usage of **IS NULL** operator:

```
SQL> SELECT  ID, NAME, AGE, ADDRESS, SALARY
     FROM CUSTOMERS
     WHERE SALARY IS NULL;
```

This would produce the following result:

```
+----+----------+-----+-----------+----------+
| ID | NAME     | AGE | ADDRESS   | SALARY   |
+----+----------+-----+-----------+----------+
|  6 | Komal    |  22 | MP        |          |
|  7 | Muffy    |  24 | Indore    |          |
+----+----------+-----+-----------+----------+
```

Changing table name using alias. Alias is using for changing the table or column name temporary.

Table alias

MySQL 5.7 Command Line Client

```
mysql> select O.ordertype, O.Order_receiver, E.Name
    -> from orders as O, employee as E
    -> where O.order_id=E.order_id;
+-----------+----------------+---------+
| ordertype | Order_receiver | Name    |
+-----------+----------------+---------+
| Kabab     | Mokhles        | Mokhles |
| Shingara  | Kashem         | Hashem  |
| Puri      | Mokbul         | Mokbul  |
| Samucha   | Hashem         | Kashem  |
+-----------+----------------+---------+
4 rows in set (0.00 sec)
```

Column alias

```
mysql> select Id as C_Id, Name as C_Name, Age as C_Age
    -> from customers
    -> ;
+------+--------+-------+
| C_Id | C_Name | C_Age |
+------+--------+-------+
|    8 | Chris  |    27 |
|    9 | Hassan |    25 |
|   12 | Khan   |    26 |
|   31 | Yemmy  |    25 |
|   35 | Hassan |    27 |
+------+--------+-------+
5 rows in set (0.01 sec)
```

Index:

- Indexing is used to search and find out the data quickly.
- Always use index in a big table, and huge columns
- Use index on to those columns and tables which are more likely to be searched on frequently.
- The advantage of using index is it make faster the SELECT and WHERE clause but it slow down the UPDATE and INSERT statement.
- Usually the user cannot see the result of the index
- Updating the column with index takes more time.

Single column index

```
mysql> create index idx_first_name
    -> from actor (first_name);
ERROR 1064 (42000): You have an error in your SQL syntax; check
irst_name)' at line 2
mysql> create index first_name
    -> on actor(first_name);
Query OK, 0 rows affected (0.39 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Unique index

```
mysql> create unique index (actor_id)
    -> on actor (actor_id);
ERROR 1064 (42000): You have an error in your SQL sy
on actor (actor_id)' at line 1
mysql> create unique index actor_id
    -> on actor (actor_id);
Query OK, 0 rows affected (0.29 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Multiple index

```
mysql> create index nameindex
    -> on actor(first_name, last_name);
Query OK, 0 rows affected (0.38 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Drop index

MySQL 5.7 Command Line Client

```
at line 1
mysql> alter table actor
    -> drop index nameindex;
Query OK, 0 rows affected (0.24 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql>
```

Alter Command:

Alter command is used to add, drop, modify the column in a table

Adding column

```
mysql> alter table employee_info
    -> add column Address char(25);
Query OK, 0 rows affected (1.22 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from employee_info;
+-------------+---------+------+---------+
| employee_id | Name    | age  | Address |
+-------------+---------+------+---------+
|           9 | Mokhles |   17 | NULL    |
|          11 | Hashem  |   25 | NULL    |
|          13 | Mokbul  |   35 | NULL    |
|          14 | Kashem  |   21 | NULL    |
|          15 | Kamala  |   25 | NULL    |
+-------------+---------+------+---------+
5 rows in set (0.00 sec)
```

Dropping column

```
mysql> alter table employee_info
    -> drop column Address;
Query OK, 0 rows affected (1.32 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from employee_info;
+-------------+---------+------+
| employee_id | Name    | age  |
+-------------+---------+------+
|           9 | Mokhles |   17 |
|          11 | Hashem  |   25 |
|          13 | Mokbul  |   35 |
|          14 | Kashem  |   21 |
|          15 | Kamala  |   25 |
+-------------+---------+------+
5 rows in set (0.00 sec)
```

Modify datatypes of a column

```
mysql> alter table employee_info
    -> modify column Address char(25);
Query OK, 6 rows affected (0.99 sec)
Records: 6  Duplicates: 0  Warnings: 0
```

```
mysql> desc employee;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| Employee_ID | int(5)      | NO   | PRI | NULL    |       |
| Name        | varchar(20) | YES  |     | NULL    |       |
| Order_ID    | int(10)     | YES  | MUL | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)

mysql> alter table employee
    -> modify Name varchar(20) not null;
Query OK, 0 rows affected (0.84 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc employee;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| Employee_ID | int(5)      | NO   | PRI | NULL    |       |
| Name        | varchar(20) | NO   |     | NULL    |       |
| Order_ID    | int(10)     | YES  | MUL | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set (0.02 sec)
```

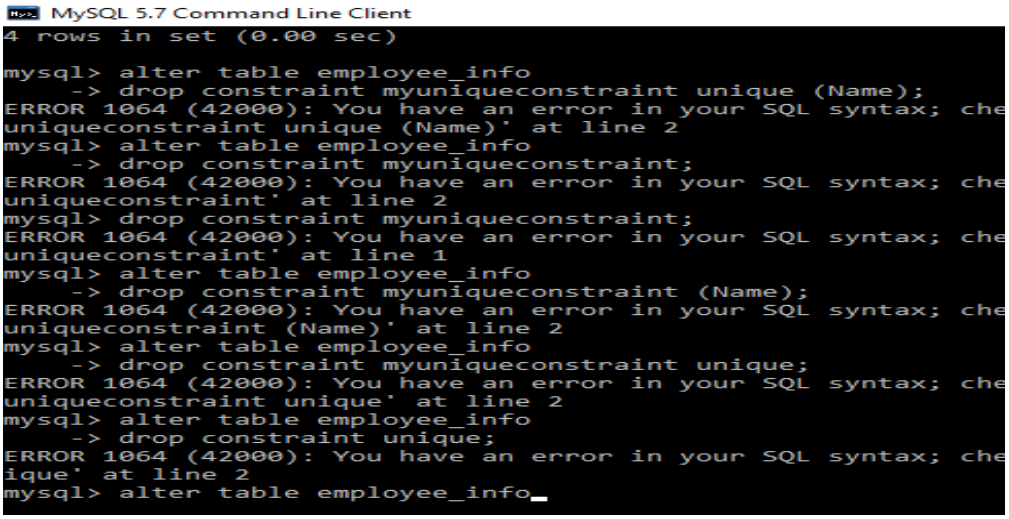
Add unique constraint

```
mysql> desc employee_info;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| employee_id | int(2)      | NO   |     | NULL    |       |
| Name        | varchar(20) | YES  |     | NULL    |       |
| age         | int(2)      | YES  |     | NULL    |       |
| Address     | char(25)    | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> alter table employee_info
    -> add constraint myuniqueconstraint unique (Name);
Query OK, 0 rows affected (0.32 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc employee_info;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| employee_id | int(2)      | NO   |     | NULL    |       |
| Name        | varchar(20) | YES  | UNI | NULL    |       |
| age         | int(2)      | YES  |     | NULL    |       |
| Address     | char(25)    | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

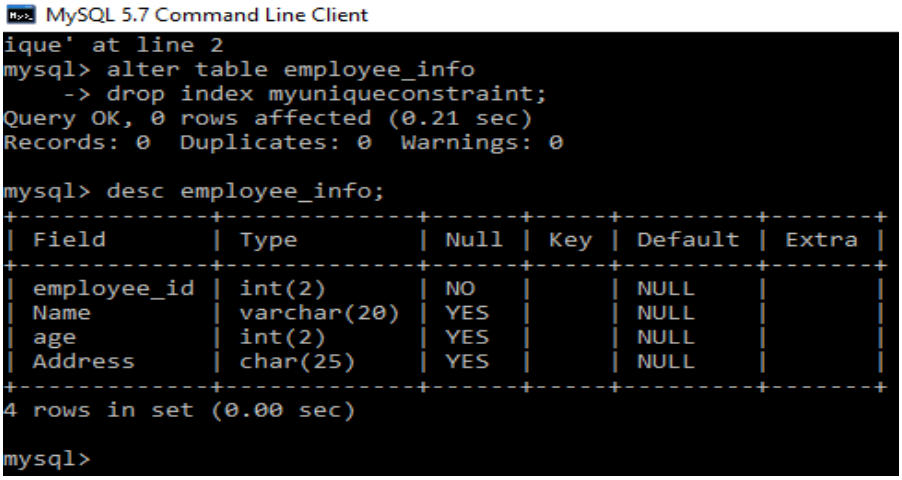
Drop constraint

There is a slight differences in coding between SQL and MY SQL in case of drop constraint. In sql the code is as following :

```
ALTER TABLE table_name
DROP CONSTRAINT MyUniqueConstraint;
```



In case of My SQL it is:



Check constraint

```
mysql> alter table employee_info
    -> add constraint myuniqueconstraint check(age>17);
Query OK, 0 rows affected (0.07 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from employee_info;
+-------------+--------+------+----------------------+
| employee_id | Name   | age  | Address              |
+-------------+--------+------+----------------------+
|           9 | Rafid  |   32 | 1407 Chicago Street  |
|          11 | Hashem |   25 | 1407 Chicago Street  |
|          13 | Mokbul |   35 | 1407 Chicago Street  |
|          14 | Kashem |   21 | 1407 Chicago Street  |
|          15 | Kamala |   25 | 1407 Chicago Street  |
+-------------+--------+------+----------------------+
5 rows in set (0.00 sec)
```

Add primary key

MySQL 5.7 Command Line Client

```
mysql> alter table employee_info
    -> add constraint myprimarykey PRIMARY KEY (employee_id);
Query OK, 0 rows affected (1.16 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc employee_info;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| employee_id | int(2)      | NO   | PRI | NULL    |       |
| Name        | varchar(20) | YES  |     | NULL    |       |
| age         | int(2)      | YES  |     | NULL    |       |
| Address     | char(20)    | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

Truncate table: if you want to delete the data of a table by keeping the table then we can use truncate table syntax. The difference between truncate table and drop table is that drop table will delete the whole table from the database. So in future we need to create the whole table by create table syntax. But in truncate command it will only delete the data inside the table by keeping the table.

```
mysql> select * from employee_info;
+-------------+--------+------+---------+
| employee_id | Name   | age  | Address |
+-------------+--------+------+---------+
|           9 | Rafid  |   32 | NULL    |
|          11 | Hashem |   25 | NULL    |
|          13 | Mokbul |   35 | NULL    |
|          14 | Kashem |   21 | NULL    |
|          15 | Kamala |   25 | NULL    |
+-------------+--------+------+---------+
5 rows in set (0.00 sec)

mysql> truncate table employee_info;
Query OK, 0 rows affected (0.29 sec)

mysql> select * from employee_info;
Empty set (0.00 sec)

mysql> desc employee_info;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| employee_id | int(2)      | NO   | PRI | NULL    |       |
| Name        | varchar(20) | YES  |     | NULL    |       |
| age         | int(2)      | YES  |     | NULL    |       |
| Address     | char(20)    | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

View: the main purpose of view is when we need to work on or multiple columns of a table instead of the whole table of the database. It works like we are creating a new temporary table with our selected columns.

```
+----+--------+-----+------------+------------+
|  8 | Chris  |  27 | Khulna     | Samucha    |
|  9 | Hassan |  25 | Dhaka      | Shingara   |
| 12 | Khan   |  26 | Chittagong | Puri       |
| 31 | Yemmy  |  25 | Rajshahi   | Kabab      |
| 35 | Hassan |  27 | Dhaka      | Puri       |
+----+--------+-----+------------+------------+
5 rows in set (0.50 sec)

mysql> create view customers_view as
    -> select Name, Address
    -> from customers;
Query OK, 0 rows affected (0.29 sec)

mysql> select * from customers_view;
+--------+-------------+
| Name   | Address     |
+--------+-------------+
| Chris  | Khulna      |
| Hassan | Dhaka       |
| Khan   | Chittagong  |
| Yemmy  | Rajshahi    |
| Hassan | Dhaka       |
+--------+-------------+
5 rows in set (0.10 sec)

mysql> create view nameorder as
    -> select Name, ordertype
    -> from customers;
Query OK, 0 rows affected (0.04 sec)

mysql> select * from nameorder;
+--------+-----------+
| Name   | ordertype |
+--------+-----------+
| Chris  | Samucha   |
| Hassan | Shingara  |
| Khan   | Puri      |
| Yemmy  | Kabab     |
| Hassan | Puri      |
+--------+-----------+
5 rows in set (0.08 sec)
```

Create view with where clause and check option. The purpose of the WITH CHECK OPTION is to ensure that all UPDATE and INSERTs satisfy the condition(s) in the view definition.

```
mysql> create view agebarrier as
    -> select name, age
    -> from customers
    -> where age is not null
    -> with check option;
Query OK, 0 rows affected (0.13 sec)

mysql> select * from agebarrier;
+---------+-----+
| name    | age |
+---------+-----+
| Chris   |  27 |
| Hassan  |  25 |
| Khan    |  26 |
| Yemmy   |  25 |
| Hassan  |  27 |
+---------+-----+
5 rows in set (0.08 sec)
```

```
mysql> create view age_barrier as
    -> select name, age
    -> from customers
    -> where age is null
    -> with check option;
Query OK, 0 rows affected (0.08 sec)

mysql> select * from age_barrier;
Empty set (0.10 sec)
```

MySQL 5.7 Command Line Client

```
mysql> create view age_close as
    -> select name, age
    -> from customers
    -> where age is null;
Query OK, 0 rows affected (0.05 sec)

mysql> select * from customers;
+----+--------+-----+-------------+------------+
| Id | Name   | Age | Address     | Ordertype  |
+----+--------+-----+-------------+------------+
|  8 | Chris  |  27 | Khulna      | Samucha    |
|  9 | Hassan |  25 | Dhaka       | Shingara   |
| 12 | Khan   |  26 | Chittagong  | Puri       |
| 31 | Yemmy  |  25 | Rajshahi    | Kabab      |
| 35 | Hassan |  27 | Dhaka       | Puri       |
+----+--------+-----+-------------+------------+
5 rows in set (0.00 sec)

mysql> select * age_close;
ERROR 1064 (42000): You have an error in your SQL
 line 1
mysql> select * from age_close;
Empty set (0.00 sec)
```

Update view: when we update a view the data of the parent column will also change with the update of the view table.

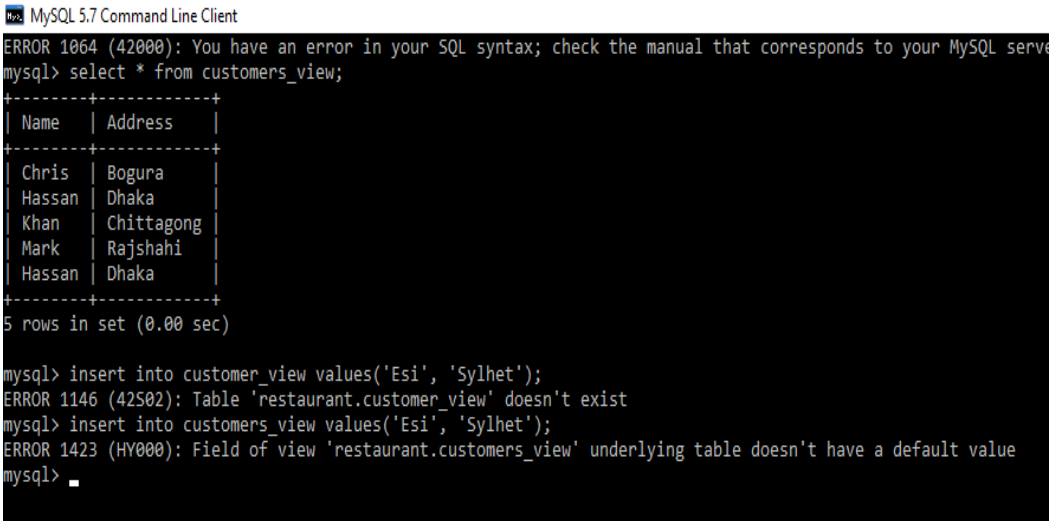
Create view with where clause condition.

```
mysql> create view agelimit as
    -> select name, age
    -> from customers
    -> where age>25;
Query OK, 0 rows affected (0.12 sec)

mysql> select * from agelimit;
+--------+-----+
| name   | age |
+--------+-----+
| Chris  |  27 |
| Khan   |  26 |
| Hassan |  27 |
+--------+-----+
3 rows in set (0.06 sec)
```

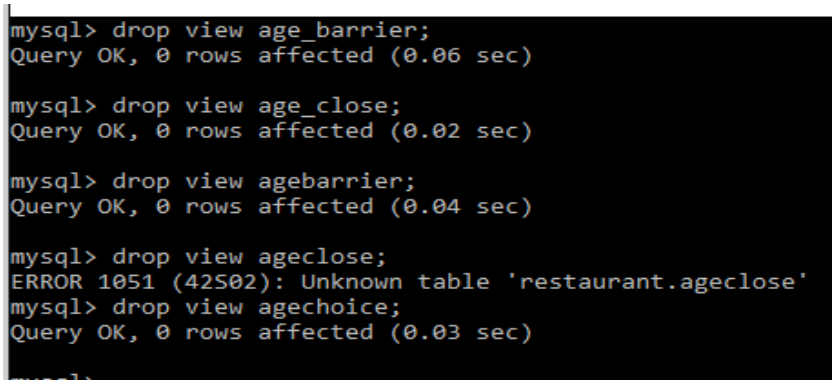
```
mysql> update customers_view
    -> set name='Mark'
    -> where address='Rajshahi';
Query OK, 1 row affected (0.30 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from customers_view;
+--------+-------------+
| Name   | Address     |
+--------+-------------+
| Chris  | Khulna      |
| Hassan | Dhaka       |
| Khan   | Chittagong  |
| Mark   | Rajshahi    |
| Hassan | Dhaka       |
+--------+-------------+
5 rows in set (0.00 sec)

mysql> select * from customers;
+----+--------+-----+-------------+------------+
| Id | Name   | Age | Address     | Ordertype  |
+----+--------+-----+-------------+------------+
|  8 | Chris  |  27 | Khulna      | Samucha    |
|  9 | Hassan |  25 | Dhaka       | Shingara   |
| 12 | Khan   |  26 | Chittagong  | Puri       |
| 31 | Mark   |  25 | Rajshahi    | Kabab      |
| 35 | Hassan |  27 | Dhaka       | Puri       |
+----+--------+-----+-------------+------------+
5 rows in set (0.00 sec)
```

Inserting Rows into a View:

Rows of data can be inserted into a view. The same rules that apply to the UPDATE command also apply to the INSERT command. Here we can not insert rows in CUSTOMERS_VIEW because we have not included all the NOT NULL columns in this view, otherwise you can insert rows in a view in similar way as you insert them in a table.



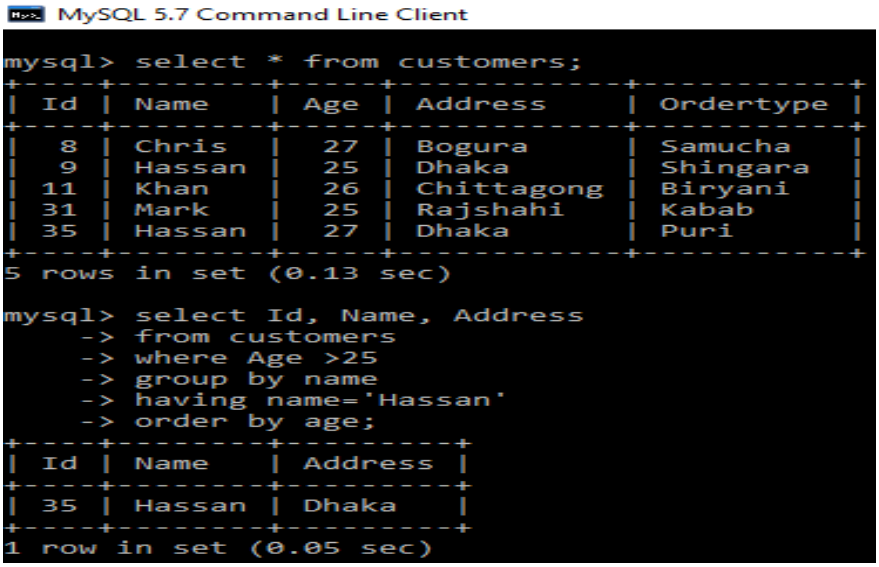
Drop view



Having Clause: the having clause is used as like as where clause but it is used only when there is any group by statement is used

```
Syntax: SELECT column1, column2
FROM table1, table2
WHERE [ conditions ]
GROUP BY column1, column2
HAVING [ conditions ]
ORDER BY column1, column2
```

If some one asks you that find out the id, name and address of the guy who's age is more than 25 years and name is Hassan then you can use the having clause like following.



Transaction:

- **COMMIT:** to save the changes.
- **ROLLBACK:** to rollback the changes.
- **SAVEPOINT:** creates points within groups of transactions in which to ROLLBACK
- **SET TRANSACTION:** Places a name on a transaction.

Transactional control commands are only used with the DML commands INSERT, UPDATE and DELETE only. They can not be used while creating tables or dropping them because these operations are automatically commited in the database.

Commit

```
mysql> update employee_info
    -> set salary=7500.00
    -> where name='Kamal';
Query OK, 1 row affected (0.05 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from employee_info;
+-------------+---------+------+---------+---------+
| employee_id | Name    | age  | Address | Salary  |
+-------------+---------+------+---------+---------+
|           9 | Mokhles |   32 | Barisal | 5000.00 |
|          11 | Hashem  |   35 | Dhaka   | 7000.00 |
|          13 | Mokbul  |   26 | Sylhet  | 8000.00 |
|          14 | Kashem  |   29 | Rongpur | 3500.00 |
|          15 | Kamal   |   22 | Joshore | 7500.00 |
+-------------+---------+------+---------+---------+
5 rows in set (0.00 sec)

mysql> delete from employee-info
    -> where age=22;
ERROR 1064 (42000): You have an error in your SQL synta
where age=22' at line 1
mysql> delete from employee_info
    -> where age=22;
Query OK, 1 row affected (0.05 sec)

mysql> commit;
Query OK, 0 rows affected (0.03 sec)

mysql> select * from employee_info;
+-------------+---------+------+---------+---------+
| employee_id | Name    | age  | Address | Salary  |
+-------------+---------+------+---------+---------+
|           9 | Mokhles |   32 | Barisal | 5000.00 |
|          11 | Hashem  |   35 | Dhaka   | 7000.00 |
|          13 | Mokbul  |   26 | Sylhet  | 8000.00 |
|          14 | Kashem  |   29 | Rongpur | 3500.00 |
+-------------+---------+------+---------+---------+
4 rows in set (0.00 sec)
```

Rollback:

```
mysql> rollback employee_info;
ERROR 1064 (42000): You have an error in your SQL syntax
' at line 1
mysql> rollback
    -> where age=22;
ERROR 1064 (42000): You have an error in your SQL syntax
 at line 2
mysql> rollback;
Query OK, 0 rows affected (0.00 sec)

mysql> select * from employee_info;
+-------------+---------+------+---------+---------+
| employee_id | Name    | age  | Address | Salary  |
+-------------+---------+------+---------+---------+
|           9 | Mokhles |   32 | Barisal | 5000.00 |
|          11 | Hashem  |   35 | Dhaka   | 7000.00 |
|          13 | Mokbul  |   26 | Sylhet  | 8000.00 |
|          14 | Kashem  |   29 | Rongpur | 3500.00 |
+-------------+---------+------+---------+---------+
4 rows in set (0.00 sec)

mysql>
```

In the upper screenshot the rollback command did not work well. It might be the reason of the MY SQL command mismatch.

A SAVEPOINT is a point in a transaction when you can roll the transaction back to a certain point without rolling back the entire transaction.

Temporary table:

```
mysql> create temporary table Restaurant_sales;
ERROR 1113 (42000): A table must have at least 1 column
mysql> create temporary table Restaurant_sales(Item_name varchar(15), item_price decimal (5, 2), unit_sold int (5), total_sales int(5));
Query OK, 0 rows affected (0.07 sec)
```

Changing the not null into the column

MySQL 5.7 Command Line Client

```
mysql> alter table Restaurant_sales
    -> modify column Item_name varchar(15) not null;
Query OK, 0 rows affected (0.00 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc restaurant_sales;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| Item_name   | varchar(15) | NO   |     | NULL    |       |
| item_price  | decimal(5,2)| YES  |     | NULL    |       |
| unit_sold   | int(5)      | YES  |     | NULL    |       |
| total_sales | int(5)      | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

```
mysql> insert into restaurant_sales values('Shingara', 5.00, 1, 5);
Query OK, 1 row affected (0.00 sec)

mysql> insert into restaurant_sales values('Puri', 2.00, 2, 4);
Query OK, 1 row affected (0.00 sec)

mysql> insert into restaurant_sales values('Kabab', 3.00, 2, 6);
Query OK, 1 row affected (0.00 sec)

mysql> insert into restaurant_sales values('Samucha', 1.00, 3, 3);
Query OK, 1 row affected (0.00 sec)

mysql> insert into restaurant_sales values('Roshogolla', 2.00, 6, 12);
Query OK, 1 row affected (0.00 sec)

mysql> select * from restaurant_sales;
+-------------+-------------+-------------+--------------+
| Item_name   | item_price  | unit_sold   | total_sales  |
+-------------+-------------+-------------+--------------+
| Shingara    |        5.00 |           1 |            5 |
| Puri        |        2.00 |           2 |            4 |
| Kabab       |        3.00 |           2 |            6 |
| Samucha     |        1.00 |           3 |            3 |
| Roshogolla  |        2.00 |           6 |           12 |
+-------------+-------------+-------------+--------------+
5 rows in set (0.00 sec)
```

Create sequence in a table:

Frequently databases use the sequential order to call any row of the table. To do that we need to make sequence. In sequence if we do not put any order or serial number it will automatically show the result with a serial.

Syntax:

```
mysql> create table customer_info(customer_id int(10) not null auto_increment, customer_name varchar(20), income decimal(15,2), primary key (customer_id));
Query OK, 0 rows affected (0.40 sec)
```

```
ERROR 1075 (42000): Incorrect table definition; there can be only one auto column and it must be defined as a key
mysql> create table customer_info(customer_id int(10) unsigned not null auto_increment, customer_name varchar(20), income decimal(10, 2), primary key(customer_id));
Query OK, 0 rows affected (0.35 sec)

mysql> desc customer_info;
+--------------+-----------------+------+-----+---------+----------------+
| Field        | Type            | Null | Key | Default | Extra          |
+--------------+-----------------+------+-----+---------+----------------+
| customer_id  | int(10) unsigned | NO   | PRI | NULL    | auto_increment |
| customer_name | varchar(20)     | YES  |     | NULL    |                |
| income       | decimal(10,2)   | YES  |     | NULL    |                |
+--------------+-----------------+------+-----+---------+----------------+
3 rows in set (0.00 sec)
```

MySQL 5.7 Command Line Client

```
ERROR 1054 (42S22): Unknown column 'Chris' in 'field list'
mysql> insert into customer_info values(null, 'Chris', 15000.20);
Query OK, 1 row affected (0.07 sec)

mysql> insert into customer_info values(null, 'Hassan', 20000.50);
Query OK, 1 row affected (0.03 sec)

mysql> insert into customer_info values(null, 'Khan', 12000.00);
Query OK, 1 row affected (0.07 sec)

mysql> insert into customer_info values(null, 'Mark', 15000.00);
Query OK, 1 row affected (0.07 sec)

mysql> insert into customer_info values(null, 'Hassan', 15000.00);
Query OK, 1 row affected (0.05 sec)

mysql> select * from customer_info;
+-------------+---------------+----------+
| customer_id | customer_name | income   |
+-------------+---------------+----------+
|           1 | Chris         | 15000.20 |
|           2 | Hassan        | 20000.50 |
|           3 | Khan          | 12000.00 |
|           4 | Mark          | 15000.00 |
|           5 | Hassan        | 15000.00 |
+-------------+---------------+----------+
5 rows in set (0.00 sec)
```

Renumbering or resequencing a table: there might be situation when we need to renumbering our tables. In that case at first, we need to drop that exact column from the table by using alter command and then add that column again. Before doing so we need to be very careful that our particular table should not have any join.

The syntax is as followed

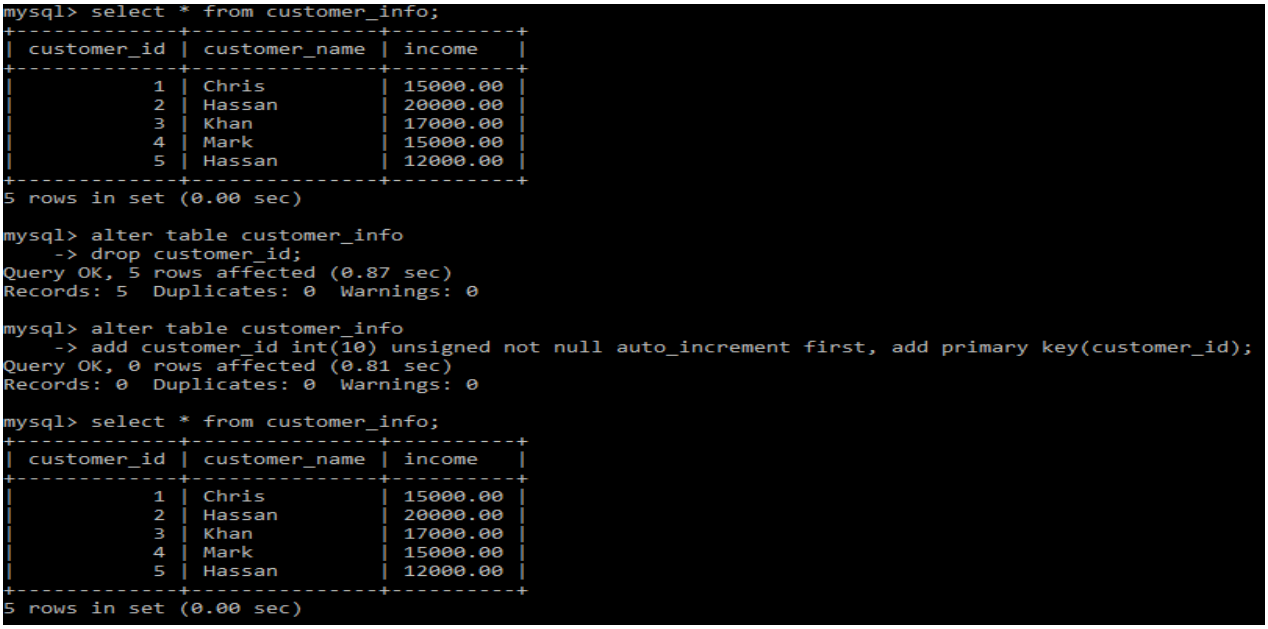
>Alter table table_name

>drop column_name;

Then again add that column

>alter table table_name

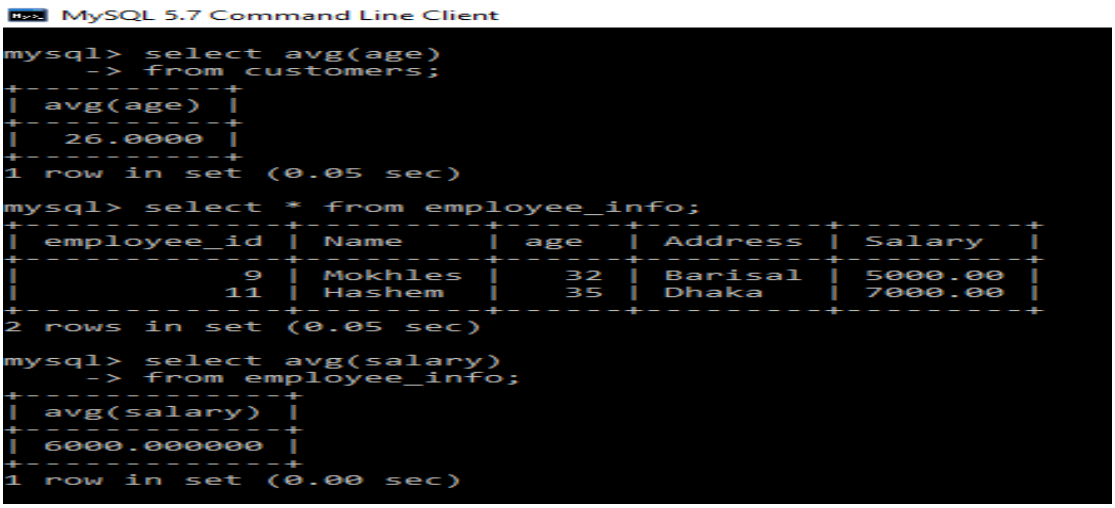
>add column_name datatype unsigned not null auto_increment first, add primary key(column_name);



Average function: when you need to find out any average of any particular column then it would be like the following.

Basic syntax: >select avg(column_name)

>from table_name;

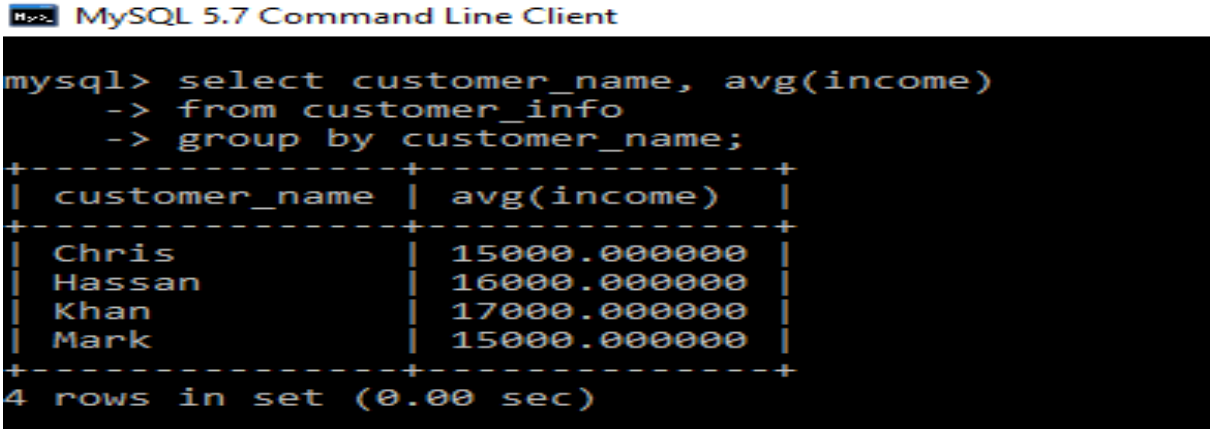


When you need to find out the averages of several records then the syntax would be as following.

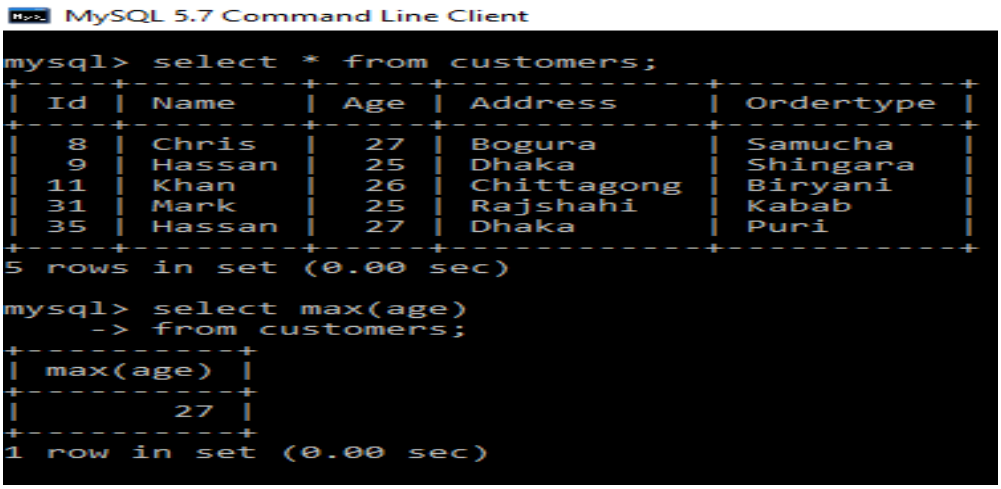
>select column_name(**this is the column which you need to do the grouping) avg(column_name)

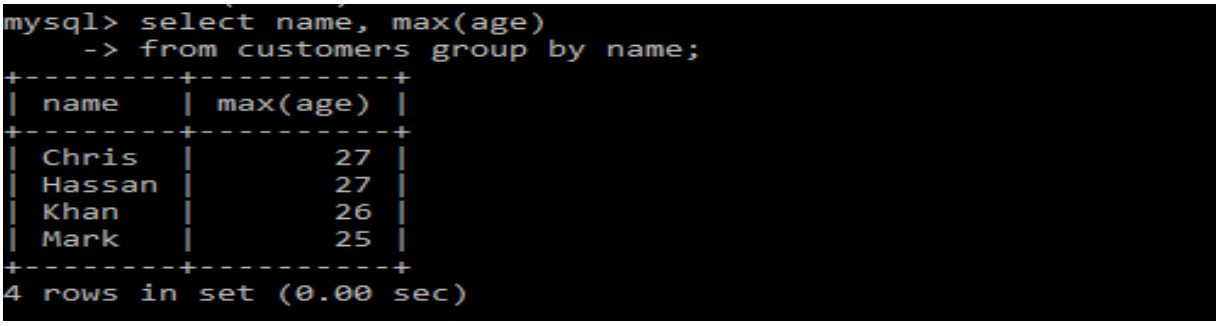
>from table_name

>group by column_name



Max function:

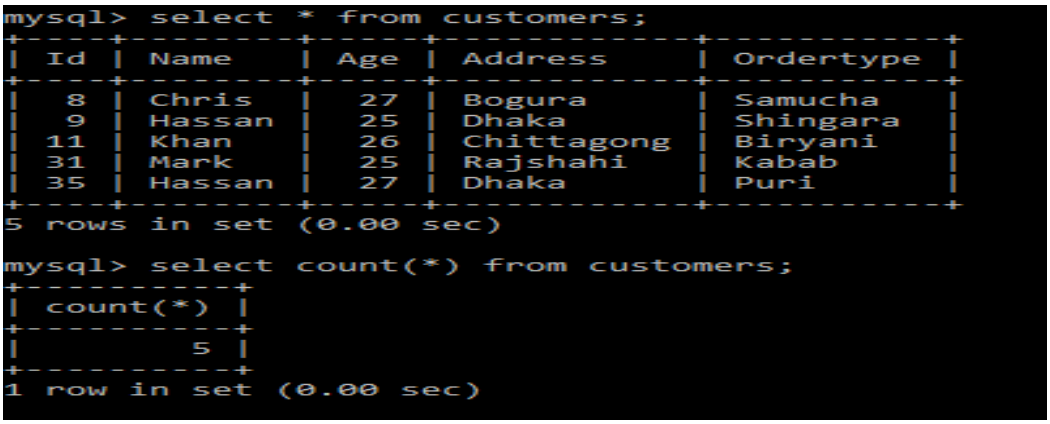




Count function: it will show us the number of all rows in a table or rows in a particular column.

No.of total rows in a table

>select count(*) from table_name;



No. of rows in a particular table

MySQL 5.7 Command Line Client

```
mysql> select count(*) from customers
    -> where age=27;
+------------+
| count(*)   |
+------------+
|          2 |
+------------+
1 row in set (0.00 sec)

mysql> select count(*) from customers
    -> where age=25;
+------------+
| count(*)   |
+------------+
|          2 |
+------------+
1 row in set (0.00 sec)

mysql> select count(*) from customers
    -> where name='Hassan';
+------------+
| count(*)   |
+------------+
|          2 |
+------------+
1 row in set (0.00 sec)
```

Concat Function: it is used to concatenate two different columns. We can also use order by or group by clause as needed.

MySQL 5.7 Command Line Client

```
1 row in set (0.00 sec)

mysql> select * from customers;
+----+---------+------+-------------+------------+
| Id | Name    | Age  | Address     | Ordertype  |
+----+---------+------+-------------+------------+
|  8 | Chris   |   27 | Bogura      | Samucha    |
|  9 | Hassan  |   25 | Dhaka       | Shingara   |
| 11 | Khan    |   26 | Chittagong  | Biryani    |
| 31 | Mark    |   25 | Rajshahi    | Kabab      |
| 35 | Hassan  |   27 | Dhaka       | Puri       |
+----+---------+------+-------------+------------+
5 rows in set (0.00 sec)

mysql> select concat(id, name)
    -> from customers;
+--------------------+
| concat(id, name)   |
+--------------------+
| 8Chris             |
| 9Hassan            |
| 35Hassan           |
| 11Khan             |
| 31Mark             |
+--------------------+
5 rows in set (0.00 sec)

mysql> select concat(id, name)
    -> from customers
    -> order by name;
+--------------------+
| concat(id, name)   |
+--------------------+
| 8Chris             |
| 9Hassan            |
| 35Hassan           |
| 11Khan             |
| 31Mark             |
+--------------------+
5 rows in set (0.00 sec)
```

```
MySQL 5.7 Command Line Client
5 rows in set (0.00 sec)

mysql> select concat(id, name)
    -> from customers
    -> order by id;
+----------------------+
| concat(id, name)     |
+----------------------+
| 8Chris               |
| 9Hassan              |
| 11Khan               |
| 31Mark               |
| 35Hassan             |
+----------------------+
5 rows in set (0.00 sec)
```

Sql numeric function:

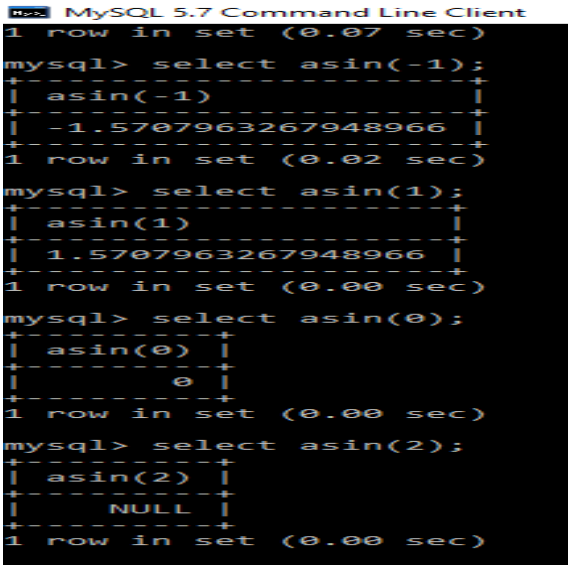
Absolute(abs) function it will give us the absolute of any numeric value.

```
mysql> select abs(200);
+-------------+
| abs(200)    |
+-------------+
|         200 |
+-------------+
1 row in set (0.04 sec)

mysql> select abs(-200);
+--------------+
| abs(-200)    |
+--------------+
|          200 |
+--------------+
1 row in set (0.02 sec)

mysql> select abs(-12);
+-------------+
| abs(-12)    |
+-------------+
|          12 |
+-------------+
1 row in set (0.00 sec)
```

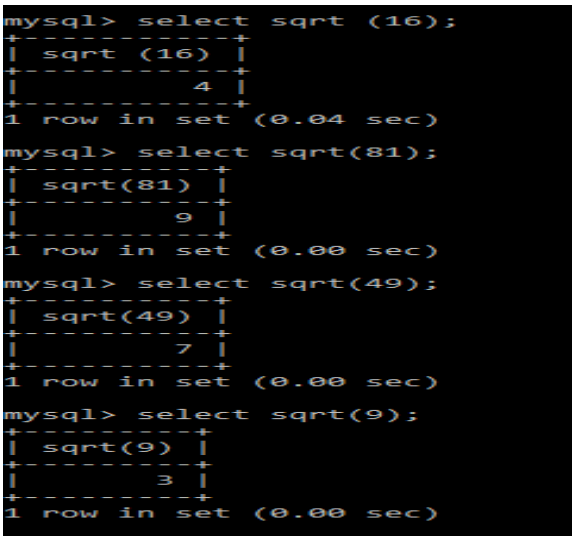
Acos function: it will give us the accosine of any numeric value. The range should between -1 to +1 otherwise it will provide us the null value.

```
mysql> select acos(-1);
+---------------------+
| acos(-1)            |
+---------------------+
| 3.1415926535897793 |
+---------------------+
1 row in set (0.05 sec)

mysql> select acos(1);
+----------+
| acos(1)  |
+----------+
|        0 |
+----------+
1 row in set (0.00 sec)

mysql> select acos(-2);
+-----------+
| acos(-2)  |
+-----------+
|      NULL |
+-----------+
1 row in set (0.00 sec)

mysql> select acos(0);
+---------------------+
| acos(0)             |
+---------------------+
| 1.5707963267948966 |
+---------------------+
1 row in set (0.07 sec)
```

Asin function:



Sqrt function: to find out the square root function.



```
mysql> select * from customer_info;
+-------------+---------------+----------+
| customer_id | customer_name | income   |
+-------------+---------------+----------+
|           1 | Chris         | 15000.00 |
|           2 | Hassan        | 20000.00 |
|           3 | Khan          | 17000.00 |
|           4 | Mark          | 15000.00 |
|           5 | Hassan        | 12000.00 |
+-------------+---------------+----------+
5 rows in set (0.00 sec)

mysql> select customer_name, sqrt(income)
    -> from customer_info;
+---------------+--------------------+
| customer_name | sqrt(income)       |
+---------------+--------------------+
| Chris         | 122.47448713915891 |
| Hassan        |  141.4213562373095 |
| Khan          | 130.38404810405297 |
| Mark          | 122.47448713915891 |
| Hassan        | 109.54451150103323 |
+---------------+--------------------+
5 rows in set (0.00 sec)

mysql> select * from employee_info;
+-------------+---------+------+---------+---------+
| employee_id | Name    | age  | Address | Salary  |
+-------------+---------+------+---------+---------+
|           9 | Mokhles |   32 | Barisal | 5000.00 |
|          11 | Hashem  |   35 | Dhaka   | 7000.00 |
+-------------+---------+------+---------+---------+
2 rows in set (0.00 sec)

mysql> select name, sqrt(salary)
    -> from employee_info;
+---------+-------------------+
| name    | sqrt(salary)      |
+---------+-------------------+
| Mokhles | 70.71067811865476 |
| Hashem  | 83.66600265340756 |
+---------+-------------------+
```