



1- Assume the following table:

CREATE TABLE MyTable (uid int, pid int, PRIMARY KEY (uid,pid));

What output will the following statements yield?

INSERT IGNORE INTO MyTable (uid,pid) VALUES (1,1),(1,2),(1,3),(1,4);

select * from MyTable;

uid, pid

1, 1

1, 2

1, 3

1, 4

INSERT INTO MyTable (uid,pid) VALUES (1,1),(1,2),(1,3),(1,4) ON DUPLICATE KEY UPDATE uid=pid-1;

select * from MyTable;

uid, pid

0, 1

1, 2

2, 3

3, 4

REPLACE INTO MyTable (uid,pid) VALUES (1,1),(1,2),(1,3),(1,4);

select * from MyTable;

uid, pid

0, 1

1, 1

1, 2

1, 3

1, 4

2, 3

3, 4

2- Suppose that the table Asd has the following structure and contents:

Field	Type	Null	Key	Default	Extra
name	binary(7)	YES		NULL	

name
Lennart
lennart
LENNART
lEnNaRt

What output will the following statements yield?

SELECT DISTINCT name FROM Asd;

name
Lennart
lennart
LENNART
lEnNaRt

SELECT name, COUNT(*) FROM Asd GROUP BY name;

name	COUNT(*)
LENNART	1
Lennart	1
lEnNaRt	1
lennart	1

SELECT name, COUNT(*) FROM Asd;

ERROR 1140 (42000): Mixing of GROUP columns (MIN(),MAX(),COUNT()...) with no GROUP columns is illegal if there is no GROUP BY clause

3. The table **access_log** contains information on the number of times employees of a secured office open a door protected by personal ID number (PIN) codes. The structure of the table is:

Field	Type	Null	Key	Default	Extra
PIN	char(6)	NO	PRI		
entries	int(10) unsigned	NO		0	

The system was been put into use recently, and the table contains the following entries:

PIN	entries
156734	6
578924	2
479645	10
356845	5

Now, two employees enter through the secured door using their PIN codes.

- The first employee uses the PIN code 578924.
- The second employee, who has not used the system before, uses the PIN code 687456 (which is a valid PIN for the door).

How can you log both entries in the access_log table using a statement that is the same for each entry (except for the PIN codes)?

Answer:

For the first employee, an UPDATE statement is needed to increase the count of the times PIN code 578924 has been used to open the door. For the second employee, a new record must be entered into the table, as PIN code 687456 is being used for the first time.

You can provide for both occurrences by utilizing the ON DUPLICATE KEY UPDATE clause of the INSERT statement:

INSERT INTO access_log (PIN, entries) VALUES ('578924', 1) ON DUPLICATE KEY UPDATE entries = entries+1;

INSERT INTO access_log (PIN, entries) VALUES ('687456', 1) ON DUPLICATE KEY UPDATE entries = entries+1;

After these two statements are executed, the access_log table has the following contents:

PIN	entries
156734	6
578924	3
479645	10
356845	5
687456	1

4. The table petnames contains the following data:

name
Lucy
Macie
Myra
Cheep
Lucy
Myra
Cheep
Macie
Pablo
Stefan

Assume that you issue the following statement:

```
mysql> UPDATE petnames SET name = CONCAT(name, '_!!!') ORDER BY name LIMIT 1;
```

What will the table's contents be after the UPDATE?

Answer:

name
Lucy
Macie
Myra
Cheep_!!!
Lucy
Myra
Cheep
Macie
Pablo
Stefan

5. The table MyTable contains the following data:

```
CREATE TABLE MyTable (i int, j int);
```

```
insert into mytable values(42,55),(42,66),(42,77),(23,88),(23,99);
```

Write sql statement to swap between values of two columns i, j.

Answer:

```
update mytable set i=i-j,j=j+i,i=j-i;
```