Group 8

Hasna sherin tp – 26

Jincy – 28

Jithin t Mathews – 30

Kripa p – 31

Rohit N – 48

Sandra jeyes c – 50

Topic:Set operations

1. UNION
2. UNION ALL
3. INTERSECT
4. MINUS

**UNION** is used to combine the results of two or more SELECT statements. However it will eliminate duplicate rows from its resultset. In case of union, number of columns and datatype must be same in both the tables, on which UNION operation is being applied.

**UNION** **ALL**-This operation is similar to Union. But it also shows the duplicate rows

**Intersect** operation is used to combine two SELECT statements, but it only retuns the records which are common from both SELECT statements

The **Minus** operation combines results of two SELECT statements and return only those in the final result, which belongs to the first set of the result.

Questions:

1.consider the following schema.emptest(empid int,empname string,empaddr string,empssn string),empdesign(empid int,empname string,empaddr string,empssn string).perform the following operations on the table.

1.union

2.Union all

3.Intersect

4.Minus

Create statement

create table emptest(empid int,empname varchar(20),empaddr

varchar(20),empssn varchar(20));

create table empdesign(empid int,empname varchar(20),empaddr

varchar(20),empssn varchar(20));

insert into emptest values((1,’akshay’,’bnglr’,11)(2,’anandhu’,’bnglr’,22)(3,’hadi’,bnglr’,33)(4,’jithin’,’bnglr’,44));

insert into empdesign values((1,’akshay’,’bnglr’,11)(4,’jithin’,’bnglr’,44));

query:

1.select from emptiest union empdesign;

2.select from emptiest union all empdesign;

3.mysql does not support intersect,intersect done using join

Select emptest.empid,emptest.empname from emptest on emptest.empid=empdesign.empid and emptest.empname=empdesign.empname;

4.mysql does not support minus,minus done using join.

Select emptest.empid,emptest.empname from emptest join empdesign using (empid,empname)

emptest

|  |  |  |  |
| --- | --- | --- | --- |
| Empid | empname | empaddr | Empssn |
| 1 | Akshay | bnglr | 11 |
| 2 | Anandhu | Bnglr | 22 |
| 3 | Hadi | Bnglr | 33 |
| 4 | jithin | bnglr | 44 |

Empdesign

|  |  |  |  |
| --- | --- | --- | --- |
| empid | empname | empaddr | Empssn |
| 1 | Akshay | Bnglr | 11 |
| 4 | jithin | Bnglr | 44 |

Output

1.

|  |  |  |  |
| --- | --- | --- | --- |
| Empid | empname | empaddr | Empssn |
| 1 | Akshay | bnglr | 11 |
| 2 | Anandhu | Bnglr | 22 |
| 3 | Hadi | Bnglr | 33 |
| 4 | jithin | bnglr | 44 |

2.

|  |  |  |  |
| --- | --- | --- | --- |
| Empid | empname | empaddr | Empssn |
| 1 | Akshay | bnglr | 11 |
| 2 | Anandhu | Bnglr | 22 |
| 3 | Hadi | Bnglr | 33 |
| 4 | jithin | bnglr | 44 |
| 1 | akshay | bnglr | 11 |
| 4 | jithin | bnglr | 44 |

3.

|  |  |
| --- | --- |
| Empid | Empname |
| 1 | Akshay |
| 4 | Jtihin |

4.

|  |  |
| --- | --- |
| empid | Empname |
| 1 | Akshay |
| 4 | Jtihin |

2.consider the following schema supplier(supply id int,supply\_name varchar(20)) company(companyid int,company name varchar(20))

Retrieve the id value and name of the supplier whose id greater than 1000 and companies whose id value is >1000,rename resultant table to as id value and name value

**Create**

Create table supplier(sid int,sname varchar(20));

Create table company(cid int,cname varchar(20));

Insert into company values((500,’intel’)(2000,’dell’)(3000,’hp’));

Insert into supplier values((1000,’kt’)(3000,’hadi’)(5000,’jithin’));

Create table in(idvalue cid,namevalue varchar(20),id\_value int,name\_value varchar(20));

**Query**

Insert into in select sid,snamecid,cid,cname from supplier,company where supply.sid in(select sid from supply where sid>2000 and company.cid in (select cid from company whose cid>1000);

OUTPUT

Supply company

|  |  |
| --- | --- |
| sname | Sid |
| Kt | 500 |
| Hadi | 3000 |
| jithin | 5000 |

|  |  |
| --- | --- |
| cname | cid |
| Intel | 500 |
| Hp | 3000 |
| dell | 2000 |

|  |  |  |  |
| --- | --- | --- | --- |
| idvalue | namevalue | Id\_value | Name\_value |
| 3000 | Hadi | 3000 | Hp |
| 5000 | jithin | 2000 | Dell |