Monisha.M

80311

# MYSQL Test

1) Option 4. the subquery

2)Option 3. mysqldump

3)Option 3. column level

4)Option 3. --help

5)Option c. PATH

6)Option B.
```
1 CREATE PROCEDURE P ()
2 BEGIN
3 END
```

7)Option 1.ENUM

8)Option D.
```
CREATE TABLE IF NOT EXISTS employee (
        employeeID char(10),
    firstName varchar(50),
    lastName varchar(50),
    phone varchar(20),
    address varchar(50),
```

PRIMARY KEY (employeeID)

);

9)Option 2. IN

10)Option 2.SELECT column FROM tbl

11)Option 2.ANSI

12)Option 4.DESCRIBE table;

13)Option 4.SELECT * FROM inventory;

14)Option 2.configuration files

15)Option 3.UNION

16)Option 3.the subquery must return a single value.

17)Option 1.show grants (displays the privileges and roles that are assigned to a MySQL user account or role)

18)Option 4.insert into cars (make, model, year) values ('Ford', 'Mustang', 2002), ('Mercedes', 'C', 2003)

19)Option 4.DROP TEMPORARY TABLE customers;

20)Option 3.foreign key;

# Query Writing

1)You are working with the table in this diagram. You want to use full-text search to find the customers who live on a street or a drive. What is the command to do that?

Solution : SELECT * FROM customers

WHERE MATCH(address) AGAINST ('street, drive');

2)What is the valid way to create a database view in MySQL?

Solution : CREATE VIEW v1 AS SELECT * FROM t1;

3)You are working with the tables shown below. You need to generate the list of all cars, whether or not they had been sold. Which statement accomplishes that?

Solution : SELECT cars.*, purchases.date FROM cars LEFT JOIN purchases
ON cars.ID = purchases.carID;

4)What steps do you need to take to normalize the table from this diagram?

Solution : Create another table to serve as a lookup for powers with fields for code and description, as well as a junction table with superhero names and power codes.

# Questions

Consider the Sailors-Boats-Reserves DB described in below.

S (sid, sname, rating, age)

B (bid, bname, color)

R (sid, bid, date)

Give a SQL expression for each of the following queries:

1. Find the colors of boats reserved by Dustin.

Solution : SELECT color

FROM S,B,R

WHERE R.sid=S.sid AND R.bid=B.bid AND

sname=`Dustin’

2)Find all IDs of sailors who have a rating of at least 8 or have reserved boat 103.

Solution : (SELECT sid FROM s WHERE rating>=8) UNION (SELECT sid

FROM r WHERE bid=103)

3)Find the names of sailors who have not reserved a red boat.

Solution : SELECT S.sname FROM Sailors S, Boats B, Reserves R

WHERE B.color='red' AND B.bid=R.bid AND S.sid = R.sid

4)Find the IDs of sailors with age over 20 who have not reserved a red boat.

Solution : SELECT S.sname FROM Sailors S, Reserves R, Boats B

WHERE B.color != 'red' and B.bid = R.bid and S.sid = R.sid and
S.sid > 20

5)Find the names of sailors who have reserved at least two boats.

Solution : SELECT S.sname FROM Sailors S, Reserves R, Sailors S2, Reserves R2 WHERE S.sid = R.sid and S2.sid = R2.sid and S.sid = S2.sid and R.bid != R2.bid

6)Find the names of sailors who have reserved all boats.

Solution : SELECT sname FROM s WHERE NOT EXISTS ( SELECT * FROM b WHERE NOT EXISTS ( SELECT * FROM r WHERE r.sid=s.sid AND r.bid=b.bid))

7)Find the names of sailors who have reserved all boats called Interlake.

Solution : SELECT sname FROM s WHERE ( SELECT * FROM b WHERE bname ="Interlake");

8)Find the IDs of sailors whose rating is better than some sailor called Andy.

Solution : SELECT S.sid FROM Sailors S WHERE S.rating > ANY ( SELECT S2.rating FROM Sailors S2 WHERE S2.sname = 'Andy' )

9)Find the IDs of sailors whose rating is better than every sailor called Andy.

Solution : SELECT sid FROM s WHERE rating > all ( SELECT rating FROM s s2 WHERE s2.sname=`Andy')

10) Find the IDs of sailors with the highest rating.

Solution : SELECT S.sid FROM Sailors S WHERE S.rating>=ALL(SELECT S2.rating FROM Sailors S2 )

11)Find the name and age of the oldest sailor

Solution : SELECT s1.sname, s1.age FROM s s1 WHERE NOT EXISTS ( SELECT * FROM s s2 WHERE s2.age>s1.age)