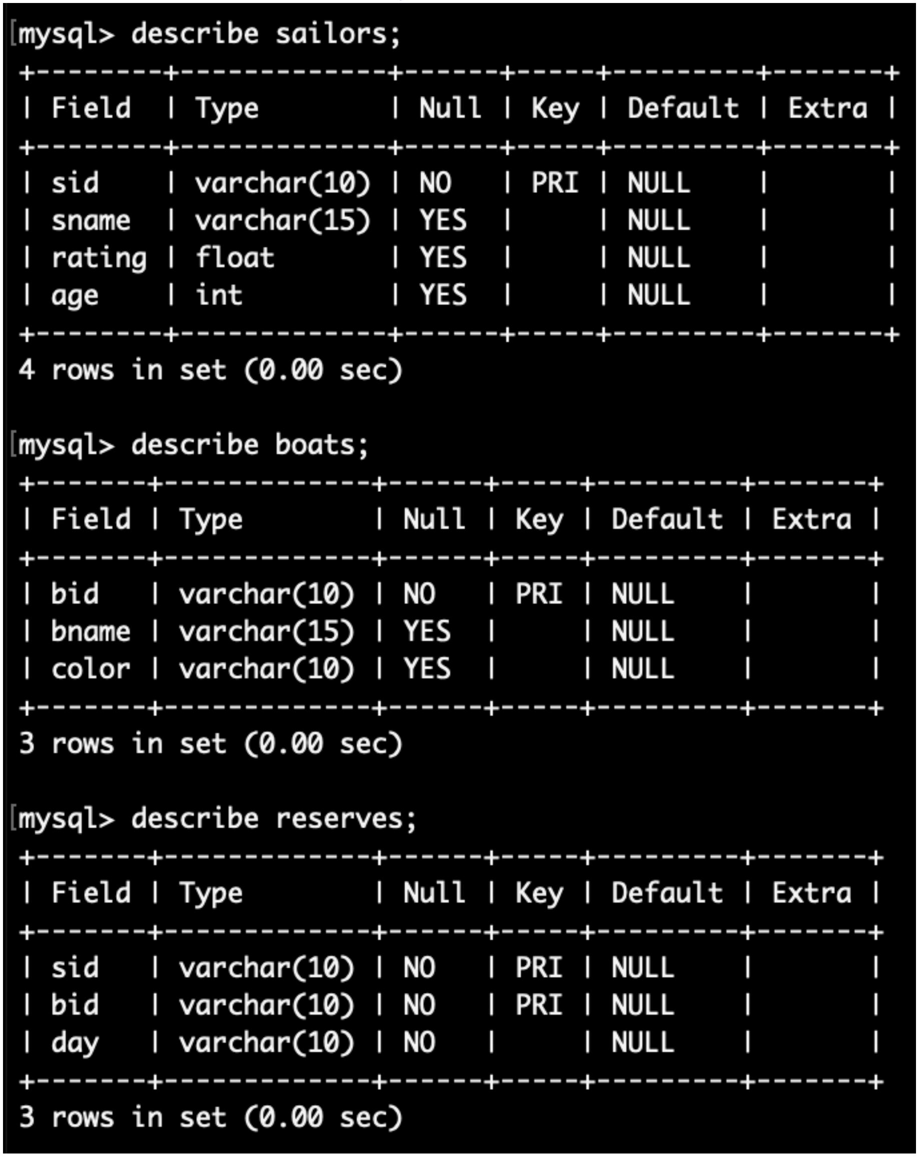
NESTED QUERY AND SUBQUERY

Ishita Sinha

202000506

Consider the following schema:

1. Sailors (sid, sname, rating, age)
2. Boats (bid, bname, color)
3. Reserves (sid, bid, day(date))

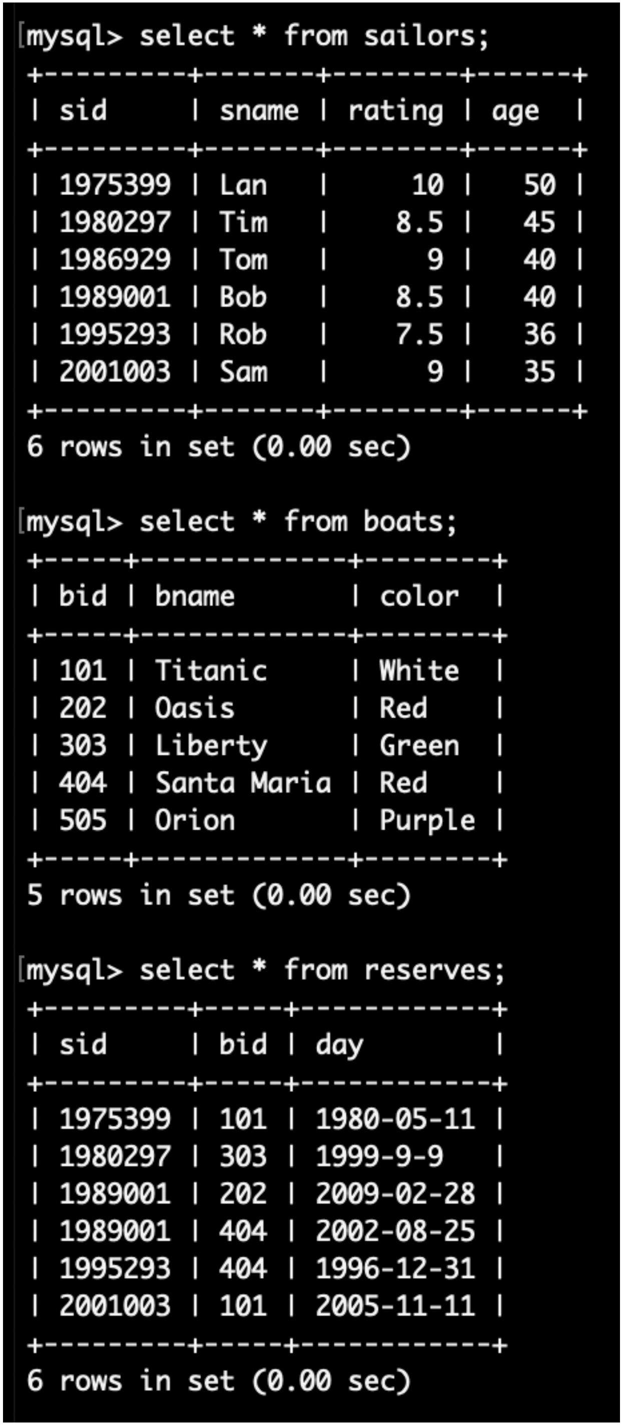


```
[mysql> describe sailors;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| sid     | varchar(10) | NO   | PRI | NULL    |       |
| sname   | varchar(15) | YES  |     | NULL    |       |
| rating  | float       | YES  |     | NULL    |       |
| age     | int         | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

[mysql> describe boats;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| bid   | varchar(10) | NO   | PRI | NULL    |       |
| bname | varchar(15) | YES  |     | NULL    |       |
| color | varchar(10) | YES  |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)

[mysql> describe reserves;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| sid   | varchar(10) | NO   | PRI | NULL    |       |
| bid   | varchar(10) | NO   | PRI | NULL    |       |
| day   | varchar(10) | NO   |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

Displaying the values entered into the tables:



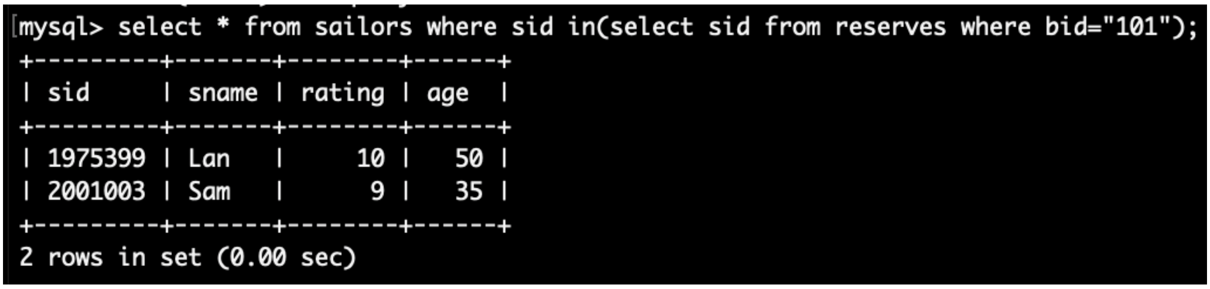
```
[mysql> select * from sailors;
+---------+-------+--------+------+
| sid     | sname | rating | age  |
+---------+-------+--------+------+
| 1975399 | Lan   |     10 |   50 |
| 1980297 | Tim   |    8.5 |   45 |
| 1986929 | Tom   |      9 |   40 |
| 1989001 | Bob   |    8.5 |   40 |
| 1995293 | Rob   |    7.5 |   36 |
| 2001003 | Sam   |      9 |   35 |
+---------+-------+--------+------+
6 rows in set (0.00 sec)

[mysql> select * from boats;
+-----+-------------+--------+
| bid | bname       | color  |
+-----+-------------+--------+
| 101 | Titanic     | White  |
| 202 | Oasis       | Red    |
| 303 | Liberty     | Green  |
| 404 | Santa Maria | Red    |
| 505 | Orion       | Purple |
+-----+-------------+--------+
5 rows in set (0.00 sec)

[mysql> select * from reserves;
+---------+-----+------------+
| sid     | bid | day        |
+---------+-----+------------+
| 1975399 | 101 | 1980-05-11 |
| 1980297 | 303 | 1999-9-9   |
| 1989001 | 202 | 2009-02-28 |
| 1989001 | 404 | 2002-08-25 |
| 1995293 | 404 | 1996-12-31 |
| 2001003 | 101 | 2005-11-11 |
+---------+-----+------------+
6 rows in set (0.00 sec)
```

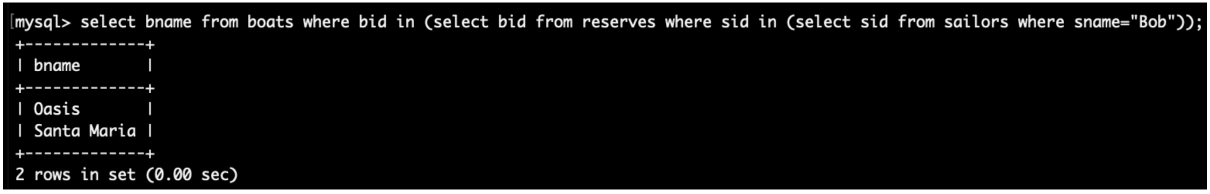
Write subquery statement for the following queries.

1. Find all information of sailors who have reserved boat number 101.



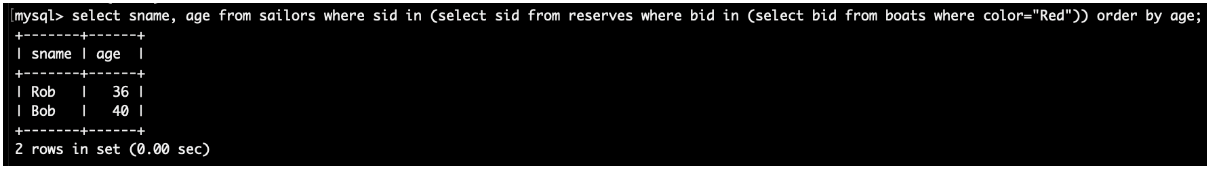
```
[mysql> select * from sailors where sid in(select sid from reserves where bid="101");
+---------+-------+--------+------+
| sid     | sname | rating | age  |
+---------+-------+--------+------+
| 1975399 | Lan   |     10 |   50 |
| 2001003 | Sam   |      9 |   35 |
+---------+-------+--------+------+
2 rows in set (0.00 sec)
```

2. Find the name of boat reserved by Bob.



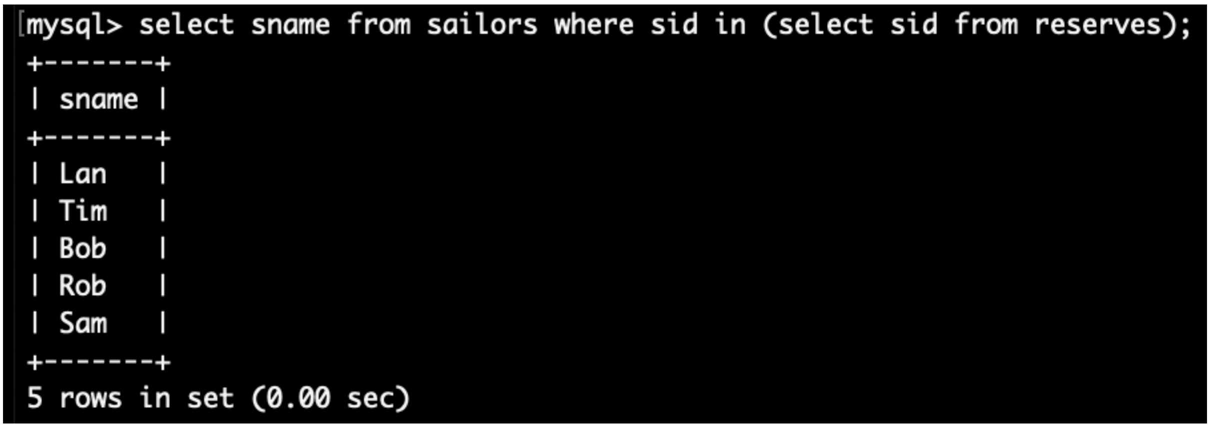
```
[mysql> select bname from boats where bid in (select bid from reserves where sid in (select sid from sailors where sname="Bob"));
+-------------+
| bname       |
+-------------+
| Oasis       |
| Santa Maria |
+-------------+
2 rows in set (0.00 sec)
```

3. Find the names of sailors who have reserved a red boat, and list in the order of age.



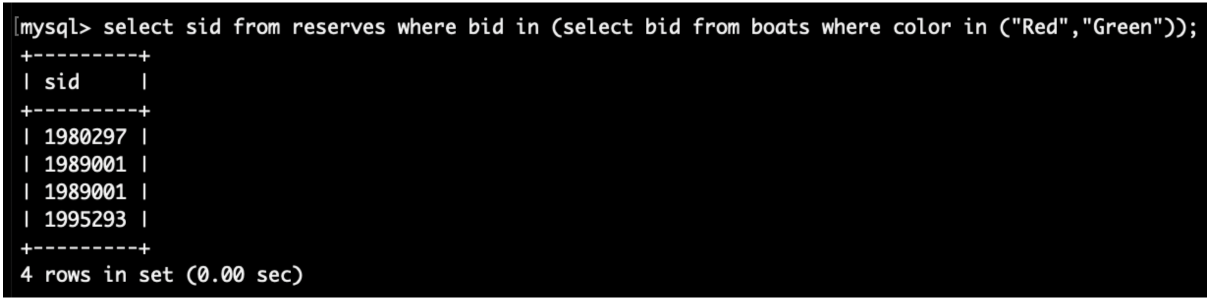
```
[mysql> select sname, age from sailors where sid in (select sid from reserves where bid in (select bid from boats where color="Red")) order by age;
+-------+------+
| sname | age  |
+-------+------+
| Rob   |   36 |
| Bob   |   40 |
+-------+------+
2 rows in set (0.00 sec)
```

4. Find the names of sailors who have reserved at least one boat.



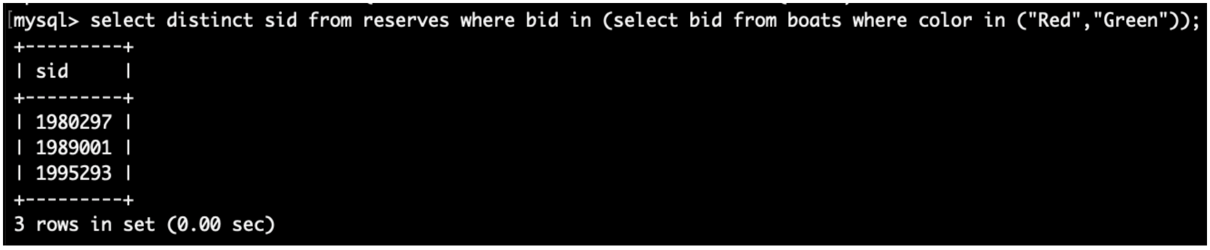
```
[mysql> select sname from sailors where sid in (select sid from reserves);
+-------+
| sname |
+-------+
| Lan   |
| Tim   |
| Bob   |
| Rob   |
| Sam   |
+-------+
5 rows in set (0.00 sec)
```

5. Find the ids of sailors who have reserved a red boat or a green boat.



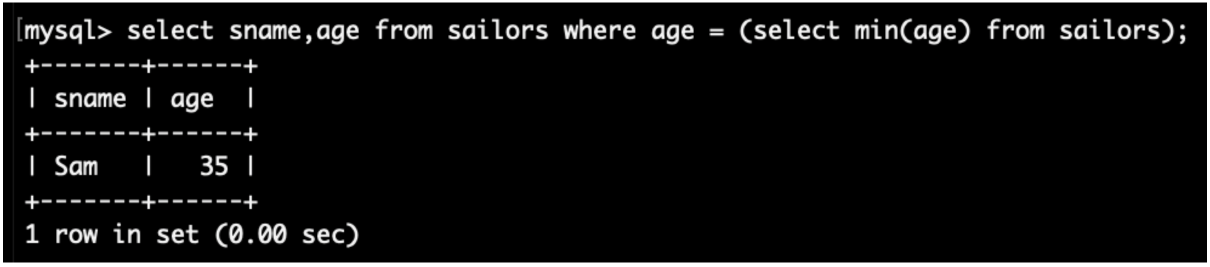
```
[mysql> select sid from reserves where bid in (select bid from boats where color in ("Red","Green"));
+---------+
| sid     |
+---------+
| 1980297 |
| 1989001 |
| 1989001 |
| 1995293 |
+---------+
4 rows in set (0.00 sec)
```

Using distinct keyword to prevent duplicates



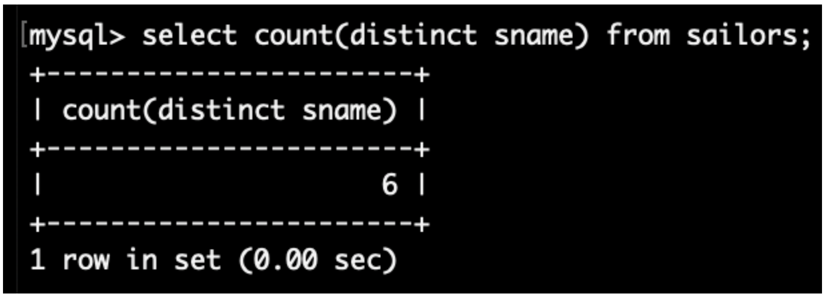
```
[mysql> select distinct sid from reserves where bid in (select bid from boats where color in ("Red","Green"));
+---------+
| sid     |
+---------+
| 1980297 |
| 1989001 |
| 1995293 |
+---------+
3 rows in set (0.00 sec)
```

6. Find the name and the age of the youngest sailor.



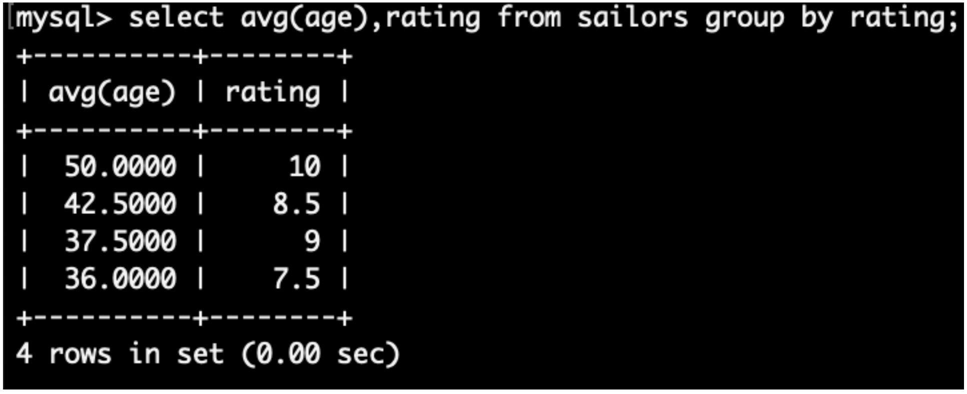
```
[mysql> select sname,age from sailors where age = (select min(age) from sailors);
+-------+------+
| sname | age  |
+-------+------+
| Sam   |   35 |
+-------+------+
1 row in set (0.00 sec)
```

7. Count the number of different sailor names.



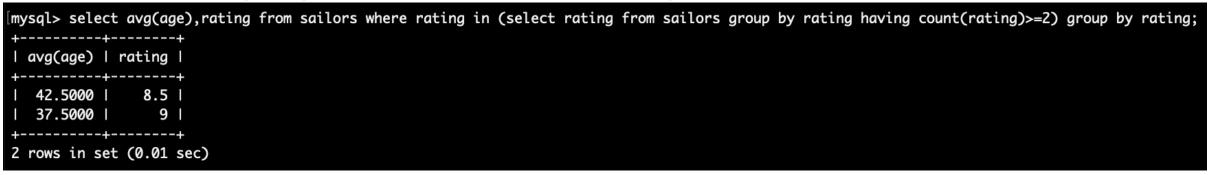
```
[mysql> select count(distinct sname) from sailors;
+-----------------------+
| count(distinct sname) |
+-----------------------+
|                     6 |
+-----------------------+
1 row in set (0.00 sec)
```

8. Find the average age of sailors for each rating level.



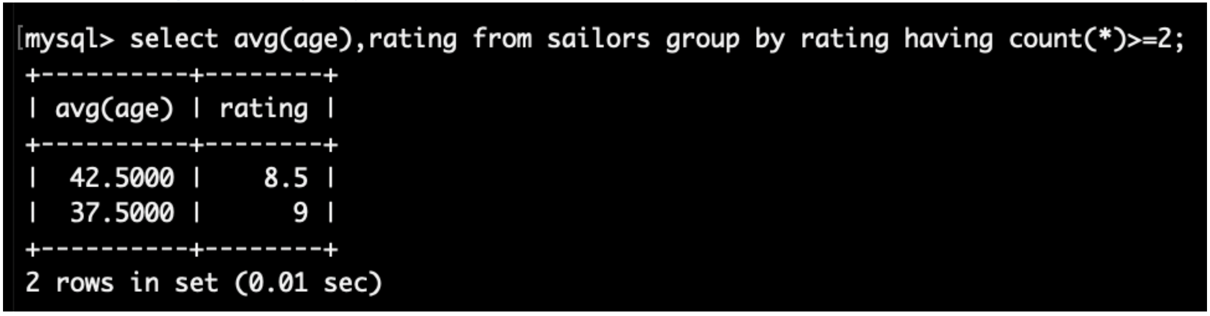
```
[mysql> select avg(age),rating from sailors group by rating;
+----------+--------+
| avg(age) | rating |
+----------+--------+
|  50.0000 |     10 |
|  42.5000 |    8.5 |
|  37.5000 |      9 |
|  36.0000 |    7.5 |
+----------+--------+
4 rows in set (0.00 sec)
```

9. Find the average age of sailors for each rating level that has at least two sailors.



```
[mysql> select avg(age),rating from sailors where rating in (select rating from sailors group by rating having count(rating)>=2) group by rating;
+----------+--------+
| avg(age) | rating |
+----------+--------+
|  42.5000 |    8.5 |
|  37.5000 |      9 |
+----------+--------+
2 rows in set (0.01 sec)
```

Without using nested query:-



```
[mysql> select avg(age),rating from sailors group by rating having count(*)>=2;
+----------+--------+
| avg(age) | rating |
+----------+--------+
|  42.5000 |    8.5 |
|  37.5000 |      9 |
+----------+--------+
2 rows in set (0.01 sec)
```