ECE464 Database
Prof.Sokolov
Di (David) Mei

Problem Set#1
Part#1

1.

```
SELECT DISTINCT b.bname, s.sname, COUNT(*) AS num_reservations
FROM sailors s, boats b, reserves r
WHERE s.sid = r.sid AND b.bid = r.bid
GROUP BY b.bid, b.bname, s.sname
HAVING num_reservations >= ALL (SELECT COUNT(*) FROM reserves r
WHERE r.bid = b.bid GROUP BY r.sid)
ORDER BY b.bname;
```

```
+-----------+----------+------------------+
| bname     | sname    | num_reservations |
+-----------+----------+------------------+
| Clipper   | dusting  |                1 |
| Clipper   | emilio   |                1 |
| Clipper   | figaro   |                1 |
| Clipper   | horatio  |                1 |
| Clipper   | lubber   |                1 |
| Clipper   | scruntus |                1 |
| Driftwood | dye      |                1 |
| Driftwood | jit      |                1 |
| Driftwood | stum     |                1 |
| Driftwood | vin      |                1 |
| Interlake | dusting  |                1 |
| Interlake | horatio  |                1 |
| Interlake | lubber   |                1 |
| Klapser   | dan      |                2 |
| Marine    | dan      |                1 |
| Marine    | emilio   |                1 |
| Marine    | figaro   |                1 |
| Marine    | jit      |                2 |
| Marine    | stum     |                1 |
| Sooney    | dan      |                1 |
| Sooney    | ossola   |                1 |
+-----------+----------+------------------+
21 rows in set (0.00 sec)
```

2.

```
SELECT b.bid, bname, COUNT(*) AS num_reservations
FROM reserves r, boats b
WHERE r.bid = b.bid
GROUP BY b.bid, bname
ORDER BY b.bid;
```

```
+-----+-----------+------------------+
| bid | bname     | num_reservations |
+-----+-----------+------------------+
| 101 | Interlake |                2 |
| 102 | Interlake |                3 |
| 103 | Clipper   |                3 |
| 104 | Clipper   |                5 |
| 105 | Marine    |                3 |
| 106 | Marine    |                3 |
| 107 | Marine    |                1 |
| 108 | Driftwood |                1 |
| 109 | Driftwood |                4 |
| 110 | Klapser   |                3 |
| 111 | Sooney    |                1 |
| 112 | Sooney    |                1 |
+-----+-----------+------------------+
12 rows in set (0.00 sec)
```

3.

```
SELECT sname, s.sid FROM sailors s
WHERE NOT EXISTS (SELECT * FROM boats b WHERE color = 'red'
AND NOT EXISTS (SELECT * FROM reserves r
WHERE r.sid = s.sid AND r.bid = b.bid));
```

```
mysql> SELECT sname, s.sid FROM sailors s
    -> WHERE NOT EXISTS (SELECT * FROM boats b WHERE color = 'red'
    -> AND NOT EXISTS (SELECT * FROM reserves r
    -> WHERE r.sid = s.sid AND r.bid = b.bid));
Empty set (0.00 sec)
```

4.

```
SELECT s.sid, sname
FROM sailors s
WHERE 'red' = ALL(SELECT color FROM reserves r, boats b
WHERE r.bid = b.bid AND r.sid = s.sid);
```

```
+-----+----------+
| sid | sname    |
+-----+----------+
|  23 | emilio   |
|  24 | scruntus |
|  29 | brutus   |
|  32 | andy     |
|  35 | figaro   |
|  58 | rusty    |
|  61 | ossola   |
|  62 | shaun    |
|  71 | zorba    |
|  85 | art      |
|  95 | bob      |
+-----+----------+
11 rows in set (0.00 sec)
```

5.

```
SELECT b.bid, bname, COUNT(*) AS num_r
FROM reserves r INNER JOIN boats b ON r.bid = b.bid
GROUP BY b.bid, bname
HAVING num_r >= ALL(SELECT COUNT(*) FROM reserves r GROUP BY r.bid);

a different method:
SELECT bid, count(bid) as num_reserves FROM reserves r
GROUP BY bid ORDER BY num_reserves DESC LIMIT 1;
```

```
+-----+---------+-------+
| bid | bname   | num_r |
+-----+---------+-------+
| 104 | Clipper |     5 |
+-----+---------+-------+
1 row in set (0.00 sec)
```

6.

```
SELECT sid, sname
FROM sailors
WHERE sid
NOT IN (SELECT r.sid
FROM reserves r INNER JOIN boats b ON r.bid = b.bid
WHERE color = 'red');
```

```
+-----+---------+
| sid | sname   |
+-----+---------+
|  29 | brutus  |
|  32 | andy    |
|  58 | rusty   |
|  60 | jit     |
|  71 | zorba   |
|  74 | horatio |
|  85 | art     |
|  90 | vin     |
|  95 | bob     |
+-----+---------+
9 rows in set (0.00 sec)
```

7.

```
SELECT AVG(age) FROM sailors WHERE rating = ’10';
```

```
+----------+
| AVG(age) |
+----------+
|  35.0000 |
+----------+
1 row in set (0.00 sec)
```

Part#2

Check source codes "tables.py", "insert.py", "query.py".

Part#3

Check source codes "new_tables.py", "new_insert.py", "new_query.py". Improvements focus on the track of every boat's maintenance:

1. Create a new object table "employees" which represent employees in the boat rental company. Employees are responsible for boats' maintenance and they have different job titles.
2. Create a new relationship table "maintenance" which include some maintenance information for each boat. For example, the cost and date of a boat's maintenance are showed in this table.
3. To ensure the security of each boat, fields including "daylast" (date of boat's last maintenance) and "numrent" (number of rental times for a boat) are added to the table "boats".
4. To keep track of the condition of each boat, the field "condition" (the general condition of a boat after it is given back by a sailor) is added to the table "reserves".
5. To keep track of the maintenance cost, the field "cost" is added to table "maintenance".
6. More fields about personal information are added, e.g. "phone" (phone number), "ssn" (social security number).