Please include the following declaration together with your signature on the cover page of your submissions/reports:

I Mo Danthuluru declare that I have completed this assignment completely and entirely on my own, without any consultation with others. I understand that any breach of the UAB Academic Honor Code may result in severe penalties.

(Please include your webuser name and password in the submission)

Webuser name :mvdanthuweb
Password : mvdanthuuab

In this assignment, you are required to create tables in your PostgreSQL database using the sample data from Figure 2.22 (pp. 56). The schema information can be found in Exercise 2.3.2 (pp. 37).

Notice that there are some "dangling tuples" in this dataset, e.g., ships mentioned in Outcomes that are not mentioned in Ships. You cannot assume that every ship has a class.

(NOTE: Your SQL queries must pass the compiler check or 0 point will be awarded. *Submit the queries in text form and the results in screenshot form.* **Avoid** using 'distinct' unless it is absolutely necessary!)

1. Textbook: pp. 279, Exercise 6.3.2 (b) and (e) (24 pts).
   (You need to provide two significantly different SQL queries for each subquestion and each query should include a subquery.)

2. Textbook: pp. 290, Exercise 6.4.7. (60pts)

Q1. (6.3.2)
b) Find the classes of ships, at least one of which was sunk in a battle.
SELECT class FROM Classes WHERE class IN (SELECT Ships.class FROM Outcomes INNER JOIN Ships ON Outcomes.ship = Ships.name WHERE Outcomes.result = 'sunk');

```
mvdanthu=> SELECT class
FROM Classes
WHERE class IN (
SELECT Ships.class
FROM Outcomes
INNER JOIN Ships ON Outcomes.ship = Ships.name
WHERE Outcomes.result = 'sunk'
);
 class
-------
 Kongo
(1 row)

mvdanthu=>
```

```
[mvdanthu=> SELECT class
[mvdanthu-> FROM Ships
[mvdanthu-> WHERE name IN (
[mvdanthu(> SELECT Ship
[mvdanthu(> FROM Outcomes
[mvdanthu(> WHERE result = 'sunk'
[mvdanthu(> );
  class
-------
  Kongo
(1 row)

mvdanthu=>
```

SELECT class FROM Ships WHERE name IN (SELECT Ship FROM Outcomes WHERE result = 'sunk');

e) Find the names of the ships whose number of guns was the largest for those ships of the same bore.

SELECT name FROM Ships, Classes as c1 WHERE Ships.class = c1.class AND numGuns=(select MAX(numGuns) FROM Classes as c2 WHERE c1.bore=c2.bore);

```
[mvdanthu=> SELECT name
[mvdanthu-> FROM Ships,
[mvdanthu-> Classes as c1
[mvdanthu-> WHERE Ships.class = c1.class
[mvdanthu-> AND numGuns=(select MAX(numGuns)
[mvdanthu(> FROM Classes as c2
[mvdanthu(> WHERE c1.bore=c2.bore);
       name
-----------------
  Iowa
  Missouri
  New Jersey
  Wisconsin
  North Carolina
  Washington
  Ramillies
  Resolution
  Revenge
  Royal Oak
  Royal Sovereign
  California
  Tennessee
  Musashi
  Yamato
(15 rows)

mvdanthu=>
```

SELECT s.name FROM Ships s, Classes c where s.class=c.class AND numGuns>=all(select numGuns from Ships s1, Classes c1 where s1.class=c1.class AND c1.bore=c.bore);

```
mvdanthu=> SELECT s.name
mvdanthu-> FROM Ships s,
mvdanthu-> Classes c where s.class=c.class
mvdanthu-> AND numGuns>=all(select numGuns from Ships s1,
mvdanthu(> Classes c1 where s1.class=c1.class AND c1.bore=c.bore);
       name
-----------------
 Wisconsin
 New Jersey
 Missouri
 Iowa
 Washington
 North Carolina
 Royal Sovereign
 Royal Oak
 Revenge
 Resolution
 Ramillies
 Tennessee
 California
 Yamato
 Musashi
(15 rows)

mvdanthu=>
```

Exercise 6.4.7: Write the following queries, based on the database schema
Classes(class, type, country, numGuns, bore, displacement)
Ships(name, class, launched)
Battles(name, date)
Outcomes(ship, battle, result)
of Exercise 2.4.3, and evaluate your queries using the data of that exercise.

a) Find the number of battleship classes.
SELECT count (*) FROM Classes where type='bb';

```
mvdanthu=> SELECT count (*) FROM Classes where type='bb';
 count
-------
     6
(1 row)
```

b) Find the average number of guns of battleship classes.

```
mvdanthu=> SELECT avg(numGuns)
mvdanthu-> FROM Classes
mvdanthu-> WHERE type ='bb';
         avg
--------------------
 9.1666666666666667
(1 row)
```

SELECT avg(numGuns) FROM Classes WHERE type ='bb';

c) Find the average number of guns of battleships. Note the difference between (b) and (c); do we weight a class by the number of ships of that class or not?

SELECT avg(numGuns) FROM Ships, Classes Where Ships.class = Classes.class AND Classes.type='bb';

```
mvdanthu=> SELECT avg(numGuns)
mvdanthu-> FROM Ships,
mvdanthu-> Classes Where Ships.class = Classes.class
mvdanthu-> AND Classes.type='bb';
         avg
--------------------
 9.0666666666666667
(1 row)

mvdanthu=>
```

d) Find for each class the year in which the first ship of that class was launched.

SELECT c.class, min(s.launched) AS year FROM Classes C inner join Ships s on c.class=s.class group by c.class order by c.class;

```
mvdanthu=> SELECT c.class, min(s.launched)
mvdanthu-> AS year
mvdanthu-> FROM Classes C
mvdanthu-> inner join Ships s
mvdanthu-> on c.class=s.class
mvdanthu-> group by c.class order by c.class;
      class       | year
------------------+------
 Iowa             | 1943
 Kongo            | 1913
 North Carolina   | 1941
 Renown           | 1916
 Revenge          | 1916
 Tennessee        | 1920
 Yamato           | 1941
(7 rows)

mvdanthu=>
```

e) Find for each class the number of ships of that class sunk in battle.

SELECT Classes.class, count(*) FROM Classes, Ships, Outcomes WHERE Classes.class=Ships.class AND Outcomes.ship=Ships.name AND result='sunk' group by classes.class;

```
mvdanthu=> SELECT Classes.class, count(*)
mvdanthu-> FROM Classes, Ships, Outcomes
mvdanthu-> WHERE Classes.class=Ships.class
mvdanthu-> AND Outcomes.ship=Ships.name
mvdanthu-> AND result='sunk'
mvdanthu-> group by classes.class;
 class | count
-------+-------
 Kongo |     1
(1 row)

mvdanthu=>
```

f) Find for each class with at least three ships the number of ships of that class sunk in battle.

SELECT s.class FROM Outcomes o, Ships s where o.ship=s.name and result='sunk' AND s.class in (
SELECT s1.class FROM Ships s1 where (SELECT count (s2.name)>=3 FROM ships s2 where class=s1.class)) group by class;

```
mvdanthu=> SELECT s.class
mvdanthu-> FROM Outcomes o,
mvdanthu-> Ships s where o.ship=s.name
mvdanthu-> and result='sunk'
mvdanthu-> AND s.class in (
mvdanthu(> SELECT s1.class FROM Ships s1 where (
mvdanthu(> SELECT count (s2.name)>=3 FROM ships s2 where class=s1.class)) group by class;
  class
-------
  Kongo
(1 row)

mvdanthu=>
```

g) The weight (in pounds) of the shell fired from a naval gun is approximately one half the cube of the bore (in inches). Find the average weight of the shell for each country's ships.

SELECT c.country, avg(bore*bore*bore/2) AS avg_weight FROM Classes c inner join Ships s on c.class = s.class group by country;

```
mvdanthu=> SELECT c.country, avg(bore*bore*bore/2)
mvdanthu-> AS avg_weight
mvdanthu-> FROM Classes c
mvdanthu-> inner join Ships s on c.class = s.class group by country;
   country   |      avg_weight
-------------+----------------------
 Gt. Britain | 1687.0000000000000000
 USA         | 1879.0000000000000000
 Japan       | 1886.6666666666666667
(3 rows)

mvdanthu=>
```