Lab Assignment

1. Requirements Summary

For this part of the assignment, you will use the supplied insurance database and complete a set of queries.
You are required to submit a single file `proj3.sql` containing:

- Oracle SQL view definitions for the queries below

The file should be executable as a script from SQL*Plus and should simply load all of the definitions into your Oracle account (assume that they will be executed later). A `proj3.sql` template file is available to save you some typing. You are free to insert additional views column formatting statements into this file, but you must preserve the view definitions.

When defining `select` queries you are required to adhere to the following output formatting conventions:

- names of people should be printed as
  GivenName FamilyName (e.g. `John Smith`)
  in a column labelled `NAME`
- addresses should be printed as
  Street, City, State, Zipcode (e.g. `123 Anzac Pde, Austin, Texas, 78232`)
  in a column labelled `ADDRESS`
- monetary values should be printed with a dollar symbol (`$`), two digits after the decimal point, and with space for 7 digits before the decimal point

Matching on names and addresses should take into account possible inconsistent use of upper and lower case in strings stored in the database (e.g. the suburb `Randwick` may have also been entered as `randwick` or `RANDWICK` or even `RaNdWiCk`).

You should use consistent and legibile formatting in laying out your SQL queries.When you making the select all query please make sure to add the column statement mentioned below. Include (brief) comments for any query that uses an "unusual" approach.

2. SQL Queries

Write Oracle SQL queries (packaged as views ... see the `proj3.sql` template file) to answer the following.

1. List the names and salaries of all the underwriters in the company.

```
SQL> column name format a20;
SQL> column salary format $9999990.00
```

2. List the names and address of all clients who have had coverage declined.

```
SQL> column name format a20
SQL> column address format a40
SQL> select * from q2;
```

3. List the names of all employees who hold an insurance policy with the company. The policy must have an "ok" status.

   ```
   SQL> column name format a20
   ```

4. List the total amount of money collected from policy premiums.

   ```
   SQL> column MoneyCollected format $9999990.00
   SQL> column MoneyCollected heading "Total|Money|Collected"
   SQL> select * from q4;
   ```

5. List the total amount of money paid *out* in claims.

   ```
   SQL> column MoneyPaid format $9999990.00
   SQL> column MoneyPaid heading "Total|Money|Paid Out"
   SQL> select * from q5;
   ```

6. List the names of clients who have made claim against their own insurance policies.

   ```
   SQL> column name format a20
   SQL> select * from q6;
   ```

7. List the policy numbers for any policies that have a coverage benefit which is larger than the market value of the car. Sort the result in ascending order of policy number.

8. How many of each make/model of car are insured? List in ascending order by the make of car.

   ```
   SQL> column make format a12
   SQL> column model format a12
   SQL> column NumberInsured format 999 heading "#"
   SQL> select * from q8;
   ```

9. For each staff member who is a policy holder, list the other staff member in the company who has processed the policies held by this staff member. That is, for each policy held by a staff member, list a staff member who has either rated or underwritten the policy.

   ```
   SQL> column PolicyHolder format a20
   SQL> column PolicyHolder heading "Employee|Holding|Policy"
   SQL> column PolicyProcessor format a20
   SQL> column PolicyProcessor heading "Employee|Processing|Policy"
   SQL> select * from q10;
   ```

10. List the policy id and the car make/model where the vehicle is covered by all of the possible coverages.

    ```
    SQL> column id heading "Policy#"
    SQL> column make heading "Make"
    SQL> column model heading "Model"
    SQL> select * from q10;
    ```

3. Doing the Work

I suggest that you create a protected subdirectory under your home directory (let's call it `/home/`*`you`*`/cs5332/proj3`) and put all of the materials for this assignment there.

The first thing you need to do is to set up a database instance of your own, so that you can start querying and updating it. The simplest way to do this is to start up an SQL*Plus session and run the following two SQL*Plus commands:

```
SQL> @/path/to/directory/makeSchema
Clearing database and building fresh schema.
SQL> @/path/to/directory/makeDb
Building tables.  Please wait ...
SQL>
```

Note: the first step removes any existing database and then recreates the schema afresh.

Once you've finished, please make a hard copy by writing each of your queries along with your output.

4. Sample Answer

```
create or replace view q1 as
    -- your SQL query;
```

```
NAME                       SALARY
-------------------- ------------
Eric Edwards            $78400.00
Wai Yao                 $91500.00
Ronald Lai              $97400.00
Kit Goh                 $78900.00
Mark Zhang              $81000.00
Yin Korth               $81700.00
Mark Saad               $77000.00
Chung Isaacson          $89900.00
Benjamin Au             $86500.00
Henry Lai               $90400.00
Jun Tsang               $89800.00
William Tanaka          $92300.00
Catherine Brown         $79000.00
Julie Tang              $88000.00
Lap Le                  $95900.00
James Baker             $94200.00
Lachlan Ma              $86400.00
Hai Lu                  $96600.00

18 rows selected.
```

Note: the first step removes any existing database and then recreates the schema afresh.