CPS610 - Assignment 4
Lab 8: Object-Oriented Database

For this assignment, you only need one instance of Oracle (use Oracle 11g on our school oracle.scs.torontomu.ca). Use the following examples and create and populate two new Professors Tables with the following attributes:

1. **Task1: Create new Professors Tables:**

   a. Create Professor1 Table to have an address included as an object. See the example below, which adds address as an object (user-defined type) to a table called addresses.

   b. Create Professor2 Table to contain a circular object type in a way that is similar to MARRIEDPERSON TABLE shown below.

2. **Task 2: Explain to your TA:**

   a. Based on your observations, explain what REF is.

3. **Task 3: Write a PL/SQL:**

   a. Add an attribute to show the number of courses a professor is teaching in the professor object, and then use PL/SQL and write a procedure to increase the number of courses a professor is teaching. You can look at the following link to learn some more fundamentals of PL/SQL: https://w3resource.com/plsql-exercises/

Deliverables:

1. Submit your assignment's source code (.sql) by the deadline for A4.

2. Please submit a readable report (PDF, Word) besides your SQL scripts. This report needs to include the following:
   - List of group members and group number.
   - A clickable index (i.e. clicking the page number takes you directly to the content).
   - You include each requirement point and label it accordingly. You need to include the screenshot(s) that support that requirement.
   - If the lab involves bonus(es) label this clearly.

--------------USE The Following Template---

```
SQL>  CREATE TYPE address_typ AS OBJECT
  2  (StreetNo      NUMBER(10),
  3  StreetName    VARCHAR2(100),
  4  AptNo         NUMBER(5),
  5  City          VARCHAR2(100),
  6  State         VARCHAR2(100),
  7  ZipCode       NUMBER(9),
  8  Country      VARCHAR2(100));


SQL> CREATE TABLE addresses of address_typ;


SQL> SELECT REF(e) FROM addresses e;

no rows selected

SQL> insert into addresses
    values(114, 'third', 2, 'San Mateo','California',43000, 'USA');

1 row created.

SQL> SELECT REF(e) FROM addresses e;

REF(E)
--------------------------------------------------------------------------------
0000280209B27053838222FAF6E040758D0DE70423B27053838221FAF6E040758D0
DE70423018000
AF0000

SQL> CREATE TYPE person_t AS OBJECT
    (name VARCHAR2(20),
    address address_typ);

Type created.

SQL> CREATE TABLE PERSON of person_t;

Table created.




SQL> INSERT INTO PERSON
   VALUES('John', address_typ(112, 'Park Place', 2, 'San Mateo','California',43000,
'USA'));

1 row created.

SQL> SELECT VALUE(e) from PERSON e;

VALUE(E)(NAME, ADDRESS(STREETNO, STREETNAME, APTNO, CITY, STATE,
ZIPCODE, COUNTR
--------------------------------------------------------------------------------
PERSON_T('John', ADDRESS_TYP(112, 'Park Place', 2, 'San Mateo', 'California', 43
000, 'USA'))


SQL> SELECT REF(e) FROM PERSON e;

REF(E)
--------------------------------------------------------------------------------
0000280209B27053838229FAF6E040758D0DE70423B27053838228FAF6E040758D0
DE70423018000
BF0000
```

Creating Circular Object Type

```
SQL> create type married_person_t as object
   (Name VARCHAR2(10),
   Spouse REF married_person_t);

Type created.

SQL> create table  MARRIEDPERSON of married_person_t;

Table created.

SQL> insert into MARRIEDPERSON(Name)
 values('John');

1 row created.



SQL> select * from MARRIEDPERSON
  2  ;

NAME
----------
SPOUSE
--------------------------------------------------------------------------------
John


SQL> insert into MARRIEDPERSON
  2  select 'Sara', REF(M)
  3   from MARRIEDPERSON M
  4   where Name='John';
```

```
1 row created.

SQL> select * from MARRIEDPERSON;

NAME
----------
SPOUSE
--------------------------------------------------------------------------------
John


Sara
0000220208B27E849E8B2C7493E040758D0DE7186AB27E849E8B277493E040758D0
DE7186A



SQL> update MARRIEDPERSON
  2   Set Spouse =
  3  (select REF(M)
  4   from MARRIEDPERSON M
  5   where M.Name='Sara')
  6  where Name='John';

1 row updated.



SQL> select * from MARRIEDPERSON;

NAME
----------
SPOUSE
--------------------------------------------------------------------------------
John
0000220208B27E849E8B2D7493E040758D0DE7186AB27E849E8B277493E040758D0
DE7186A

Sara
0000220208B27E849E8B2C7493E040758D0DE7186AB27E849E8B277493E040758D0
DE7186A
```