Evaluation (TD)

Full name :.....

Mark:.....

Evaluation (TD)

Exercise 1

R and F are a relation and a set of FDs respectively defined as R(X, Y, Z, W, V), F = {XY→ZW, XWV→YZ}. Witch of the following is an $FD \epsilon F^+$ **(True or False)**

1- XY→XZ .....True.....
2- X→ZW .....False.....
3- XYV→ZW .....True.....
4- WV→W .....True.....
5- XWV→Z .....True.....
6- XW→Y .....False.....

Exercise 2

1- Which the following FDs holds on the opposite relation "R" ? (**True** or **False**)

a) SID→Grade.....False.....
b) Faculty→F_phone.....True.....
c) C_name→Grade.....False.....
d) CID→SID .....False.....

SID	CID	S_name	C_name	Grade	Faculty	F_phone
1	[S2] A	Adams	Database	A	Howard	60192
1	[S3] B	Adams	Program	B	Langley	45869
2	[S2] A	Jones	Database	A	Howard	60192
3	[S2] B	Smith	Database	B	Howard	60192
4	[S3] A	Baker	Program	A	Langley	45869
4	[S2] B	Baker	Database	B	Howard	60192

2- Knowing that the key of the opposite relation is {SID, CID}, In what Normal Form is R?

.....2NF.....

3- Normalize "R"

.....R_1(SID, CID, S_name, C_name, Grade, Faculty).....

.....R_2(#Faculty, F_phone).....

.....

.....

Exercise 3

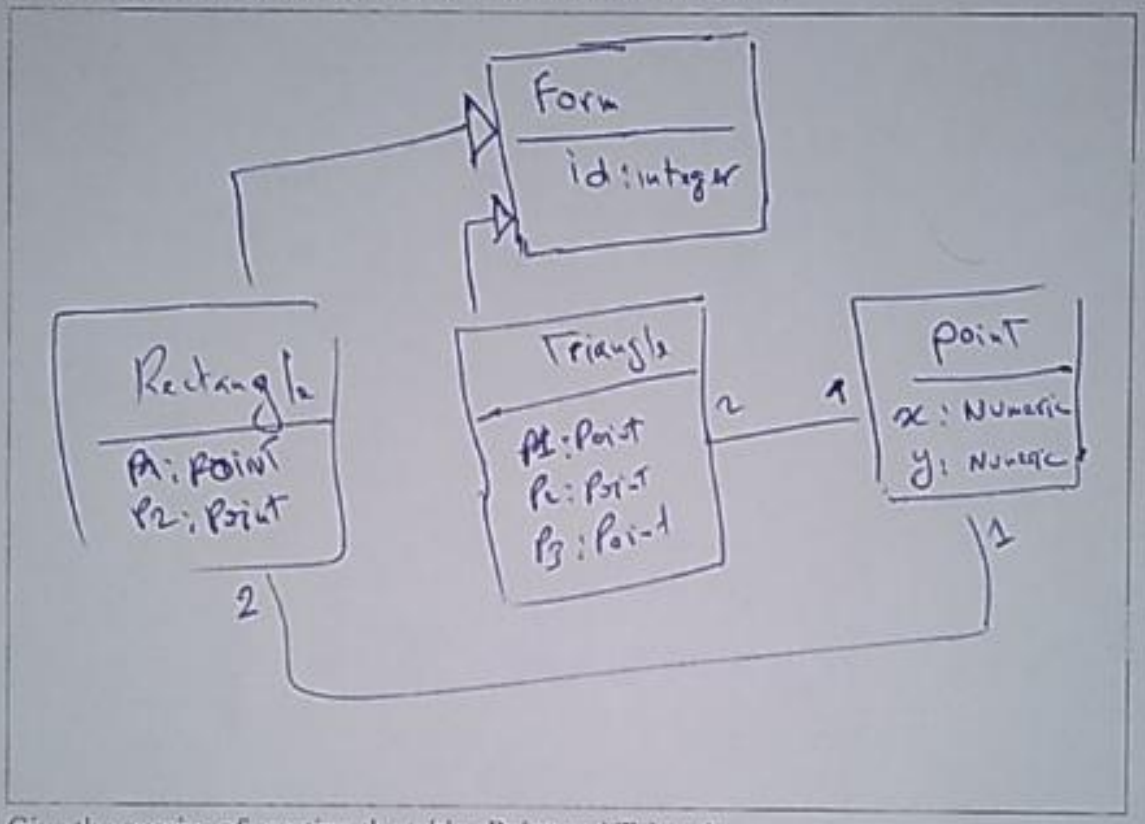
```
CREATE TYPE TForm as OBJECT(ID integer) NOT FINAL;
CREATE TYPE TPoint AS OBJECT(x NUMERIC(4,2),y NUMERIC(4,2));
CREATE TYPE TTriangle UNDER Form (p1 TPoint, p2 TPoint, p3 TPoint);
CREATE TYPE TRectangle UNDER Form (p1 TPoint, p2 TPoint);
```

Evaluation (TD)

Full name :.....

Mark:.....

1- From the previous types, deduce the entity/association model



2- Give the queries of creating the tables **Point** and **Triangle**

.....

CREATE TABLE POINT OF TPOINT ;

.....

.....

.....

.....

.....

.....

.....

.....

.....

Evaluation (TP)

Full name : .....

Mark: .....

Exercise 1

A and B are two tables defined as follows:
CREATE TABLE A(id int, x varchar2(10), y date);
CREATE TABLE B(id int, id_a int, z varchar2(10));
Where each **id** in both tables is a primary key and **id_a** in B table is a foreign key to the **A(id)**

1- Using SQL, add the keys of the table B.

```
ALTER TABLE B ADD CONSTRAINTS p_k_b
Primary Key (B);
ALTER TABLE B ADD CONSTRAINTS f_k_b_a
FOREIGN KEY (id_a) REFERENCES A(id);
```

2- For each table, insert one new record (line)

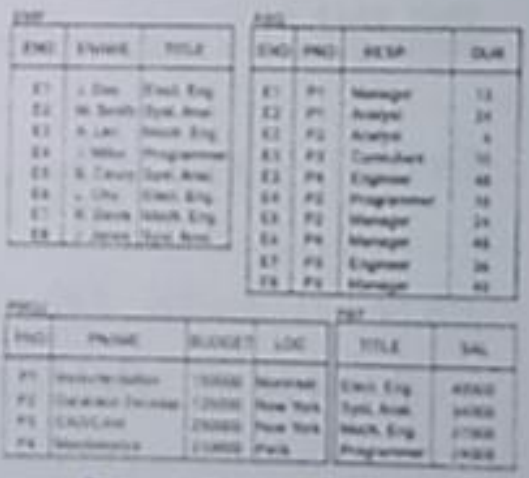
A) `INSERT INTO A VALUES (1, 'xx', 'dd/mm/yyyy');`

B) `INSERT INTO B VALUES (2, 1, 'yy');`

Exercise 2

EMP, ASG, PROJ, and PAY represent a database of four tables for Employees, Assign, Project, and Payment respectively
Give the SQL queries that extract:

1- The names of the electricity engineer **"Elect. Eng"**
2- The names of employees who participated in projects as **"Manager"**
3- The Employee Number (**ENO**) of the employee who has been assigned the maximum duration (**DUR**)



1)

```
SELECT name
FROM EMP
WHERE title='Elect. Eng'
```

2)

```
SELECT e.name
FROM EMP e
JOIN ASG a
ON e.ENO = a.ENO
AND a.RESP='manager';
```

Evaluation (TP)

Full name :.....

Mark:.....

3).....

```
SELECT ENO
FROM ASG
WHERE DUR = ( SELECT max(DUR)
              FROM ASG);
```

Exercise 3

CREATE TYPE TPart as OBJECT(id int, name varchar2(10), part REF TPart);
CREATE TABLE Part OF TPart;

We want fill the table "Part" as follows, where the OBJ.TPART are references to the first part which has the **id=1**

id	name	part
1	Chair	Null
2	Wood leg 1	OBJ.TPART
3	Wood leg 2	OBJ.TPART
4	Wood leg 3	OBJ.TPART
5	Wood leg 4	OBJ.TPART

1- Write the SQL queries that insert the two first "**parts**" of id **1** and **2**.

2- What is the result returned by the following query:

```
SELECT p.name
FROM part p
WHERE p.part = (SELECT REF(p) FROM part p WHERE p.id=1);
```

1).....

```
INSERT INTO Part VALUES (TPart(1, 'chair', NULL));

INSERT INTO Part VALUES (TPart(2, 'wood leg 1',
    (SELECT REF(P) FROM Part P
```

2).....

```
WHERE p.id = 1)));
```

name
wood leg 1
wood leg 2
wood leg 3
wood leg 4