

**Fiche de TD N° 1: BD Relationnelle Objet (SQL3 : Les méthodes)**

**Exercice 2 : Les comptes bancaires**

1. Définir un Type TClient avec les attributs suivants : NCB, Nom, Prénom, Tél.

*create type Tclient as object (*

*NCB number,*

*Nom varchar(20),*

*Prénom varchar(20),*

*Tél varchar(10) );*

2. Définir la méthode Afficher ( ) permettant d'afficher les informations du Client en cours.

Alter type TClient add member PROCEDURE Afficher CASCADE;

```
CREATE or replace TYPE BODY TClient AS
member PROCEDURE Afficher is
BEGIN
DBMS_OUTPUT.PUT_LINE('NCB:'||self.ncb);
DBMS_OUTPUT.PUT_LINE('NCB:'||self.nom);
DBMS_OUTPUT.PUT_LINE('NCB:'||self.PRÉNOM);
DBMS_OUTPUT.PUT_LINE('NCB:'||self.TÉL);
END;
END;
```

**Test :**

```
SET SERVEROUTPUT ON
DECLARE
p TClient ;
BEGIN
p:=Tclient(123,'Merabti','ADEM','0777889890');
p.Afficher() ;
END;
```

3. Créer la table Client

Create table Client of Tclient ;

4. Créer un Type TCompte caractérisée par son solde et un code ainsi que son propriétaire qui représente un client

*create or replace type TcompteB as object (*

*code number, solde float, client **ref** Tclient ) **not final**;*

5. Ajouter au Type TCompte les méthodes suivantes :

- ✓ Une méthode permettant de Crediter() le compte, prenant une somme en paramètre.

*Alter type TcompteB add member PROCEDURE Crediter(somme in float) CASCADE;*

- ✓ Une méthode permettant de Crediter() le compte, prenant une somme et un compte bancaire en paramètres, créditant le compte et débitant le compte passé en paramètres.

*Alter type TcompteB add member PROCEDURE Crediter(somme in float, compte in out TcompteB ) CASCADE*

- ✓ Une méthode permettant de Debiter() le compte, prenant une somme en paramètre

*Alter type TcompteB add member PROCEDURE Debiter (somme in float) CASCADE*

- ✓ Une méthode permettant de Débiter() le compte, prenant une somme et un compte bancaire en paramètres, débitant le compte et créditant le compte passé en paramètres.

*Alter type TcompteB add member PROCEDURE Debiter (somme in float, compte in out Tcompte) CASCADE*

- ✓ Une méthode qui permet d'afficher le résumé d'un compte.

*Alter type TcompteB add Member PROCEDURE Afficher CASCADE*

### **Implémentation des méthodes :**

*CREATE or replace TYPE BODY TcompteB AS*

```
MEMBER PROCEDURE Debiter (somme in float) is  
solde float;  
BEGIN  
if self.solde >= somme THEN  
solde:=self.solde;  
self.solde:=solde-somme;  
DBMS_OUTPUT.PUT_LINE('Opération bien effectuée');  
Else DBMS_OUTPUT.PUT_LINE('Solde insuffisant');  
end if;  
END;
```

```

MEMBER PROCEDURE Debiter (somme in float, compte in out TcompteB) is
lsolde float;
lsolde1 float;
BEGIN
  if self.solde >= somme THEN
lsolde:=self.solde;
  lsolde1:=compte.solde;
  self.solde:=lsolde-somme;
  compte.solde:=lsolde1+somme;
DBMS_OUTPUT.PUT_LINE('Opération bien effectuée');
Else DBMS_OUTPUT.PUT_LINE('Solde insuffisant');
end if;
END;

```

```

MEMBER PROCEDURE Crediter(somme in float, compte in out TcompteB) is
lsolde float;
lsolde1 float;
BEGIN
if compte.solde >= somme THEN
lsolde:=self.solde;
lsolde1:=compte.solde;
self.solde:=lsolde+somme;
compte.solde:=lsolde1-somme;
DBMS_OUTPUT.PUT_LINE('Opération bien effectuée');
Else DBMS_OUTPUT.PUT_LINE('Solde de compte insuffisant');
end if;
END;

```

```

MEMBER PROCEDURE Crediter(somme in float) is
lsolde float;
BEGIN
lsolde:=self.solde;
self.solde:=lsolde+somme;
END;

```

```

Member PROCEDURE Afficher is
c tclient;
BEGIN
DBMS_OUTPUT.PUT_LINE('CODE: '||self.code );
DBMS_OUTPUT.PUT_LINE('solde : '||self.solde);
DBMS_OUTPUT.PUT_LINE('-----Le client -----: ' );
select Deref(self.client) into c from dual;
c.Afficher() ;
END;
END;
DREF pour récupérer l'Object

```

- Créer la table client et insérer les clients ci-dessous ; ensuite créer un programme de PL /SQL pour tester les différentes méthodes.

### 1<sup>er</sup> compte(méthode 1) :

Insert into Comptes (code, solde) values(13,100);

UPDATE Comptes SET client = ( SELECT **REF(c)** FROM clients c WHERE c.NCB=133 )  
WHERE code = 13 ;

## 2<sup>ème</sup> compte(méthode 2 PL-SQL) :

DECLARE

rp ref Tclient;

**BEGIN**

SELECT **REF(c)** INTO **rp** FROM clients c WHERE c.NCB=123;

Insert into Comptes values (TcompteB(12,1234568,**rp**));

**END;**

## 3<sup>ème</sup> compte(méthode 3) :

INSERT INTO Comptes SELECT **14, 100000, REF(C)** FROM clients c WHERE c.NCB=143

- Un compte Epargne est un compte bancaire qui possède en plus un champ « Taux Intérêt » et une méthode **calculIntérêt()** qui permet de mettre à jour le solde en tenant compte des intérêts.
- ✓ Définir un Type TcompteEpargne avec la méthode **calculIntérêt()** et la méthode qui permet d'afficher le résumé d'un compte.

Create or replace type TCompteEpargne under TcompteB (  
tauxinteret float);

Alter type TCompteEpargne add member PROCEDURE **calculIntérêt** CASCADE ;

CREATE or replace TYPE BODY TCOMPTEEPARGNE AS

Member PROCEDURE calculIntérêt is

**BEGIN**

**(self AS TcompteB).Crediter((self.tauxinteret\*( self AS TcompteB).solde)/100 );**

**END;**

**END;**

- Un ComptePayant est un compte bancaire pour lequel chaque **opération de retrait et de versement** est payante et vaut 50 DA.

- ✓ Définir un Type TComptePayant avec la méthode **Debiter()** le compte, la méthode de **Crediter()** le compte et la méthode qui permet **d'afficher** le résumé d'un compte.

Create or replace type TComptePayant under TcompteB ();

- Alter type TComptePayant add **OVERRIDING** member PROCEDURE Debiter (somme in float) CASCADE ;
- Alter type TComptePayant add **OVERRIDING** member PROCEDURE Crediter (somme in float) CASCADE ;

- Alter type TComptePayant add **OVERRIDING** member PROCEDURE afficher CASCADE ;

### Implémentation des méthodes :

```
CREATE or replace TYPE BODY TComptePayant AS
OVERRIDING member PROCEDURE Debiter (somme in float) is
BEGIN
(self AS TcompteB).Debiter(somme+5 );
END;
OVERRIDING member PROCEDURE Crediter (somme in float) is
BEGIN
(self AS TcompteB).Crediter(somme-5 );
END;
OVERRIDING member PROCEDURE afficher is
BEGIN
DBMS_OUTPUT.PUT_LINE('Compte Payant:');
(self AS TcompteB).afficher();
END;
END;
```

- Créer un programme de PL /SQL pour tester les différentes méthodes.

```
SET SERVEROUTPUT ON
```

```
DECLARE
```

```
c2 TcompteB;
```

```
c1 TcompteB;
```

```
BEGIN
```

```
SELECT value(c) INTO c1 FROM comptes WHERE c.code=12;
```

```
SELECT value(c) INTO c2 FROM comptes WHERE c.code=13;
```

```
c1.afficher();
```

```
-----
```

```
c2.Crediter(20,c1);
```

```
c2.afficher();
```

```
c2.Debiter (20,c1);
```

```
c1.afficher();
```

```
END;
```