

Marcelo Tsuji Tanaka

Exercício 7 – Cria uma Classe abstrata Conta que contenha o campo Saldo de um cliente, com as operações depósito e retirada. Crie uma outra classe Conta corrente que cobra 0,1% do valor de retirada, mas não cobra para depósito. Crie outra classe Conta Poupança que cobra 0,05% para retirada e também não cobra nada para depósito.

```
create or replace type conta as object(
```

```
    saldo number,
```

```
    MEMBER FUNCTION deposito(self in out conta, valor_deposito number) return number,
```

```
    member function retirada(self in out conta, valor_retirado number) return number
```

```
)not final NOT INSTANTIABLE;
```

```
create or replace type conta_corrente under conta(
```

```
    OVERRIDING MEMBER FUNCTION deposito(self in out conta_corrente, valor_deposito  
number) return number,
```

```
    overriding member function retirada(self in out conta_corrente, valor_retirado number)  
return number
```

```
)not final;
```

```
create or replace type conta_poupanca under conta(
```

```
    OVERRIDING MEMBER FUNCTION deposito(self in out conta_poupanca, valor_deposito  
number) return number,
```

```
    overriding member function retirada(self in out conta_poupanca, valor_retirado number)  
return number
```

```
)not final;
```

```
CREATE OR REPLACE TYPE BODY conta_corrente AS
```

```
    OVERRIDING MEMBER FUNCTION deposito(self in out conta_corrente, valor_deposito  
number) RETURN number IS
```

```
    BEGIN
```

```
SELF.saldo:=saldo+valor_deposito;
```

```
RETURN saldo;
```

```
END;
```

```
OVERRIDING MEMBER FUNCTION retirada(self in out conta_corrente, valor_retirado number)  
RETURN number IS
```

```
BEGIN
```

```
SELF.saldo:=saldo-(valor_retirado*1.001);
```

```
RETURN saldo;
```

```
END;
```

```
END;
```

```
CREATE OR REPLACE TYPE BODY conta_poupanca AS
```

```
OVERRIDING MEMBER FUNCTION deposito(self in out conta_poupanca, valor_deposito  
number) RETURN number IS
```

```
BEGIN
```

```
SELF.saldo:=saldo+valor_deposito;
```

```
RETURN saldo;
```

```
END;
```

```
OVERRIDING MEMBER FUNCTION retirada(self in out conta_poupanca, valor_retirado  
number) RETURN number IS
```

```
BEGIN
```

```
SELF.saldo:=saldo-(valor_retirado*1.0005);
```

```
RETURN saldo;
```

```
END;
```

```
END;
```

```
declare
```

```
cc conta_corrente;
```

```
begin
```

```
cc := new conta_corrente(100);
```

```
DBMS_OUTPUT.PUT_LINE(cc.retirada(10));
```

```
end;
```

```
declare
    cp conta_poupanca;
begin
    cp := new conta_poupanca(100);
    DBMS_OUTPUT.PUT_LINE(cp.deposito(10));
end;
```

```
SET SERVEROUTPUT ON;
```

Exercício 8 – Uma agenda possui o nome, endereço e telefones de contato. Crie um conjunto de Objetos no Oracle capaz de armazenar um nome, um endereço e 10 números de contato. Insira a informação no Banco para validação.

```
CREATE OR REPLACE TYPE TELEFONE AS OBJECT
(
    numero number
) NOT FINAL;
```

```
CREATE OR REPLACE TYPE TELEFONES AS VARRAY(10) OF TELEFONE;
```

```
CREATE OR REPLACE TYPE AGENDA AS OBJECT
(
    nome varchar(30),
    endereco varchar(30),
    tels TELEFONES
) NOT FINAL ;
```

```
create table tb_agenda of AGENDA;
```

```

insert into tb_agenda values ('nome 1', 'rua 1',
TELEFONES(TELEFONE(1234567),TELEFONE(98765432), telefone(95959595),
telefone(85604949)));

select * from tb_agenda;

```

Exercício 9 –Um cadastro de empresas permite armazenar o nome da empresa, o CNPJ e todas as suas filiais, armazenando a localidade e o CNPJ, Crie um conjunto de Objetos no Oracle capaz de armazenar esse cadastro para um número qualquer de filiais. Insira os valores .

```

CREATE OR REPLACE TYPE filial AS OBJECT (
    cnpj number,
    localidade varchar(50)
)
CREATE TYPE filiais AS TABLE OF filial;

```

```

CREATE TYPE empresa AS OBJECT (
    nome varchar(30),
    localizacao filiais
)
CREATE TYPE empresas AS TABLE OF empresa;

```

```

CREATE TABLE corporacao (
    idCorporacao number,
    corp empresas
)

```

```

NESTED TABLE corp STORE AS tb_empresas
(NESTED TABLE localizacao STORE AS tb_filiais)

```

```

insert into corporacao values(1,empresas(
    empresa('mikolix',filiais(

```

```
        filial(123,'sorocaba'),  
        filial(546,'jandira'),  
        filial(4859,'itu')  
    )  
,  
    empresa('chcuchu',filiais(  
        filial(4859,'sao roque'),  
        filial(789,'boituva'),  
        filial(7895,'capela'),  
        filial(9685,'ourunhos')  
    )  
)  
)  
);
```

```
select * from corporacao;
```