# Practica-3-Resuelta.pdf

Wuolah: PedroChota

# Práctica 3 - Resuelta

OPERATION_ID.sql:

```
CREATE SEQUENCE secuencia_id_op START WITH 23 MINVALUE 1 MAXVALUE
999 INCREMENT BY 1;

CREATE OR REPLACE TRIGGER actualizaIdOperation
BEFORE INSERT
  ON Operation
  FOR EACH ROW
BEGIN
  :new.cod_op := secuencia_id_op.NEXTVAL;
END;
/
```

SALDO_FI.sql:

```
DROP TABLE saldo_fi;

CREATE TABLE saldo_fi (
cod_fi     SMALLINT NOT NULL   CONSTRAINT pk_saldo_fi   PRIMARY KEY,
saldo      NUMBER (10,2) CONSTRAINT nn_saldo_saldofi    NOT NULL);

CREATE OR REPLACE PROCEDURE carga_saldo_fi IS
  CURSOR fondos IS
    SELECT * FROM Fund;
  CURSOR inversiones IS
    SELECT * FROM OP_OSRFI;
BEGIN
    /*Primero inicializamos cada fondo a 0*/
    FOR fondo IN fondos LOOP
      INSERT INTO saldo_fi VALUES(fondo.id,0);
    END LOOP;
    /*Actualizamos los fondos de cada fundación*/
    FOR inversion IN inversiones LOOP
      IF(inversion.operation='subscription') THEN
        UPDATE saldo_fi s SET s.saldo=s.saldo+inversion.amount
        WHERE  inversion.Fund_id=s.cod_fi;
      END IF;
      IF(inversion.operation='refund') THEN
        UPDATE saldo_fi s SET s.saldo=s.saldo-inversion.amount
        WHERE  inversion.Fund_id=s.cod_fi;
      END IF;
    END LOOP;
END;
/

EXECUTE carga_saldo_fi;/*Ejecución del procedimiento*/

SELECT * FROM saldo_fi;
```

```sql
SELECT * FROM OP_OSRFI;
SELECT * FROM Fund;
```

## TGR_DESCUENTO.sql:

```sql
CREATE OR REPLACE TRIGGER tgr_descubierto
BEFORE UPDATE OF balance ON account
FOR EACH ROW
DECLARE
  fondos_retirados NUMBER;
BEGIN
        IF(:new.balance<:old.balance)THEN
          fondos_retirados := :old.balance - :new.balance;
          IF (fondos_retirados>:old.balance*1.10) THEN
            RAISE_APPLICATION_ERROR(-20000,'No se puede retirar un
saldo superior al 110% del saldo actual');
          END IF;
        END IF;
END;

SELECT * FROM account;
```

## TGR_OP_I.sql:

```sql
CREATE OR REPLACE TRIGGER  TRG_OP_I
BEFORE INSERT ON op_i
FOR EACH ROW
DECLARE
  dinero op_i.amount%type;
  codigo op_i.cod_op%type;
BEGIN
  dinero := :new.amount;
  codigo := :new.cod_op;
  UPDATE account ac SET ac.balance=ac.balance+dinero
  WHERE ac.id = (SELECT o."Cli-Acc_id"
                 FROM Operation o
                 WHERE o.cod_op = codigo);

END;
/

SELECT * FROM account;
SELECT * FROM operation;
/*Prueba de trigger:*/
INSERT INTO Operation values (21,TO_DATE('2015-01-30 14:07:01',
'yyyy-mm-dd hh24:mi:ss'),4);
INSERT INTO OP_I VALUES (21,0049,1653,000676,450);
```

## TGR_SALDO_FI.sql:

```sql
CREATE OR REPLACE TRIGGER TRG_SALDO_FI
```

```
AFTER INSERT
  ON OP_OSRFI
FOR EACH ROW
DECLARE
  dinero OP_OSRFI.amount%type;
BEGIN
  IF(:new.operation='subscription') THEN
    dinero := :new.amount;
  END IF;
  IF(:new.operation='refund') THEN
    dinero := :new.amount;
  END IF;

  UPDATE saldo_fi s SET s.saldo=s.saldo+dinero
  WHERE  s.cod_fi = :new.Fund_id;
END;
/

SELECT * FROM Operation;
```