**PRÁCTICA 5**

**APARTADO 1. DISPARADOR POR TABLA**

* **PROCEDIMIENTOS**

CREATE TABLE pedidos (

código CHAR(6) PRIMARY KEY,

fecha CHAR(10),

importe NUMBER(6,2),

cliente CHAR(20),

notas CHAR(1024)

);

CREATE TABLE contiene (

pedido CHAR(6),

plato CHAR(20),

precio NUMBER(6,2),

unidades NUMBER(2,0),

PRIMARY KEY(pedido,plato),

FOREIGN KEY (pedido) REFERENCES pedidos (código)

);

CREATE TABLE auditoría (

operación CHAR(6),

tabla CHAR(50),

fecha CHAR(10),

hora CHAR(8)

);

create or replace TRIGGER trigger\_pedidos

AFTER INSERT OR DELETE OR UPDATE ON pedidos

BEGIN

IF INSERTING THEN

INSERT INTO auditoría VALUES('INSERT','pedidos',to\_char(sysdate,'dd/mm/yyyy'),to\_char(sysdate,'hh:mi:ss'));

ELSIF DELETING THEN

INSERT INTO auditoría VALUES('DELETE','pedidos',to\_char(sysdate,'dd/mm/yyyy'),to\_char(sysdate,'hh:mi:ss'));

ELSIF UPDATING THEN

INSERT INTO auditoría VALUES('UPDATE','pedidos',to\_char(sysdate,'dd/mm/yyyy'),to\_char(sysdate,'hh:mi:ss'));

END IF;

END;

* **INSTRUCCIONES**

INSERT INTO pedidos VALUES('123456','10/10/2010',10.00,'Belén','---');

INSERT INTO pedidos VALUES('987654','12/12/2012',5.50,'Martín','---');

UPDATE pedidos SET notas = 'Con mucha salsa' WHERE cliente = 'Belén';

UPDATE pedidos SET importe = 2 WHERE importe > 5;

DELETE FROM pedidos WHERE código = '987654';

* **RESULTADO**

1 fila insertadas.

1 fila insertadas.

1 fila actualizadas.

2 filas actualizadas.

1 fila eliminado

**APARTADO 2. DISPARADOR POR FILA**

* **PROCEDIMIENTOS**

create or replace TRIGGER trigger\_contiene

AFTER INSERT OR DELETE OR UPDATE ON contiene

FOR EACH ROW

BEGIN

IF DELETING THEN

UPDATE pedidos SET importe = importe - :OLD.precio WHERE código = :OLD.pedido;

ELSIF UPDATING THEN

UPDATE pedidos SET importe = importe - :OLD.precio + :NEW.precio WHERE código = :OLD.pedido;

ELSIF INSERTING THEN

UPDATE pedidos SET importe = importe + (:NEW.precio \* :NEW.unidades) WHERE código = :NEW.pedido;

END IF;

END trigger\_contiene;

* **INSTRUCCIONES**

INSERT INTO contiene VALUES('123456','Pizza napolitana',10.00,2);

INSERT INTO contiene VALUES('123456','Helado',3.00,1);

UPDATE contiene SET precio = 5 WHERE plato = 'Helado';

DELETE FROM contiene WHERE plato = 'Pizza napolitana';

* **RESULTADO** (Ejecutando el apartado 1 anteriormente)

1 fila insertadas.

1 fila insertadas.

1 fila actualizadas.

1 fila eliminado

**APARTADO 3. DISPARADOR POR FILA**

* **PROCEDIMIENTOS**

SET TIMING ON

CREATE INDEX index\_pedidos ON pedidos(cliente);

***Transcurrido: 00:00:00.009***

BEGIN

FOR I IN REVERSE 1..300000

LOOP

INSERT INTO pedidos VALUES(I,'06/01/2015',10.0,'C'||I,' ');

END LOOP;

END;

***Transcurrido: 00:00:41.051***

SELECT \*

FROM pedidos

WHERE cliente = 'C300000';

***Transcurrido: 00:00:00.032***

DROP INDEX index\_pedidos;

***Transcurrido: 00:00:00.072***

SELECT \*

FROM pedidos

WHERE cliente = 'C300000';

***Transcurrido: 00:00:03.138***

* *Véase que es mucho más rápida la ejecución con índice. -*

SELECT \*

FROM pedidos

WHERE código = 300000;

***Transcurrido: 00:00:02.219***

ALTER TABLE pedidos DROP PRIMARY KEY;

***Transcurrido: 00:00:00.047***

SELECT \*

FROM pedidos

WHERE código = 300000;

***Transcurrido: 00:00:01.613***

* *Véase que es más lenta la ejecución con PRIMARY KEY. -*