

```
-----1a-----
DROP TABLE contiene;
DROP TABLE auditoría;
DROP TABLE pedidos;

CREATE TABLE pedidos (
  código   VARCHAR2(6) PRIMARY KEY,
  fecha    VARCHAR2(10),
  importe  NUMBER(6,2),
  cliente  VARCHAR2(20),
  notas    VARCHAR(1024),
  especial VARCHAR(1),
  CHECK (especial = 'S' OR especial = 'N'),
  CHECK (fecha like '%/%/%'));

CREATE TABLE contiene
(pedido   VARCHAR2(6),
 plato   VARCHAR2(20),
 precio  NUMBER(6, 2),
 unidades NUMBER(2, 0),
 PRIMARY KEY (pedido, plato),
 constraint del_pedido foreign key (pedido) references pedidos (código) on
delete cascade);

CREATE TABLE auditoría
  (operación VARCHAR2(6),
   tabla     VARCHAR2(50),
   fecha     VARCHAR2(10),
   hora      VARCHAR2(8),
   CHECK (fecha like '%/%/%'),
   CHECK (hora  like '%:~:~'),
   CHECK (operación = 'INSERT' or operación = 'DELETE' or operación = 'UPDATE')
  );
--Sin clave primaria porque se pueden repetir filas.

COMMIT;

-----
Table CONTIENE borrado.

Table AUDITORÍA borrado.

Table PEDIDOS borrado.

Table PEDIDOS creado.

Table CONTIENE creado.
```

Table AUDITORÍA creado.

Confirmación terminada.

EN ESTE MOMENTO SE ENCUENTRAN TODAS LAS TABLAS VACÍAS

```
-----1b-----
CREATE OR REPLACE TRIGGER tr_pedidos
after insert or delete or update on pedidos
  DECLARE
    v_op auditoría.operación%type;
    v_tabla auditoría.tabla%type;
    v_fecha auditoría.fecha%type;
    v_hora auditoría.hora%type;
  BEGIN
    v_tabla := 'pedidos';
    v_fecha := to_char(sysdate, 'dd/mm/yyyy');
    v_hora := to_char(sysdate, 'hh:mi:ss');
    if deleting then
      v_op := 'DELETE';
    elsif inserting then
      v_op := 'INSERT';
    elsif updating then
      v_op := 'UPDATE';
    end if;
    insert into auditoría values (v_op, v_tabla, v_fecha, v_hora);
END;
```

Trigger TR_PEDIDOS compilado

```
-----
INSERT INTO pedidos VALUES('000000', '06/01/2021', 0, 'Pepe', '', 'N');
INSERT INTO pedidos VALUES('000001', '06/01/2021', 0, 'Juan', 'Sin gluten',
'S');
INSERT INTO pedidos VALUES('000002', '06/01/2021', 0, 'María', '', 'N');
UPDATE pedidos set cliente = 'Lucía' where código = '000002';
DELETE FROM pedidos where código = '000002';
```

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

1 fila actualizadas.

1 fila eliminado

Se registran correctamente las acciones en la tabla auditoría que ahora queda

```
INSERT pedidos 05/01/2021      07:28:14
INSERT pedidos 05/01/2021      07:28:14
INSERT pedidos 05/01/2021      07:28:14
UPDATE pedidos 05/01/2021      07:28:14
DELETE pedidos 05/01/2021      07:28:14
```

La tabla pedidos queda como sigue

000000	06/01/2021	0	Pepe		N
000001	06/01/2021	0	Juan	Sin gluten	S

-----2-----

```
CREATE OR REPLACE TRIGGER tr_contiene
after insert or delete or update on contiene
  for each row
  DECLARE
    v_imp pedidos.importe%type;
  BEGIN

    if deleting then
      select importe into v_imp from pedidos where código = :old.pedido;
      v_imp := v_imp - :old.precio * :old.unidades;
      update pedidos set importe = v_imp where código = :old.pedido;
    elsif inserting then
      select importe into v_imp from pedidos where código = :new.pedido;
      v_imp := v_imp + :new.precio * :new.unidades;
      update pedidos set importe = v_imp where código = :new.pedido;
    elsif updating then
      select importe into v_imp from pedidos where código = :old.pedido;
      v_imp := v_imp - :old.precio * :old.unidades + :new.precio *
:new.unidades;
      update pedidos set importe = v_imp where código = :old.pedido;
    end if;
  END;
```

Trigger TR_CONTIENE compilado

```
-----
INSERT INTO contiene VALUES('000000', '4 quesos', 10, 2);
INSERT INTO contiene VALUES('000001', '4 quesos', 10, 1);
INSERT INTO contiene VALUES('000001', 'bbq', 15, 1);
```

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

000000	06/01/2021	20	Pepe		N
000001	06/01/2021	25	Juan	Sin gluten	S
000000	4 quesos	10	2		
000001	4 quesos	10	1		
000001	bbq	15	1		

UPDATE contiene set unidades = 2 where pedido = '000001' and plato = 'bbq';

1 fila actualizadas.

000000	06/01/2021	20	Pepe		N
000001	06/01/2021	40	Juan	Sin gluten	S
000000	4 quesos	10	2		
000001	4 quesos	10	1		
000001	bbq	15	2		

DELETE FROM contiene where pedido = '000001' and plato = '4 quesos';

1 fila eliminado

000000	06/01/2021	20	Pepe		N
000001	06/01/2021	30	Juan	Sin gluten	S
000000	4 quesos	10	2		
000001	bbq	15	2		

Se actualizan correctamente los importes de los pedidos

-----3-----

```
CREATE OR REPLACE TRIGGER tr_especial
after insert or update on pedidos
for each row
DECLARE
    v_media pedidos.importe%type;
BEGIN
    select avg(importe) into v_media from pedidos;
    if :new.importe > v_media then
        update pedidos set especial = 'S' where código = :new.código;
    else
        update pedidos set especial = 'N' where código = :new.código;
    end if;
END;
```

Es mutante! El disparador modifica la tabla a la que es sensible.

```
-----
create view v_pedidos as select * from pedidos;

create or replace procedure especiales(med pedidos.importe%type)
```

```

is
    cursor c_ped is select código, importe from pedidos;
    v_imp pedidos.importe%type;
    v_cod pedidos.código%type;
begin
    open c_ped;
    fetch c_ped into v_cod, v_imp;
    while c_ped%found loop
        if v_imp > med then
            UPDATE pedidos set especial = 'S' where código = v_cod;
        else
            UPDATE pedidos set especial = 'N' where código = v_cod;
        end if;
        fetch c_ped into v_cod, v_imp;
    end loop;
    close c_ped;
end;

```

Procedure ESPECIALES compilado

```

CREATE OR REPLACE TRIGGER tr_especial
instead of insert or update on v_pedidos
for each row
DECLARE
    v_media pedidos.importe%type;
BEGIN
    if inserting then
        insert into pedidos values(:new.código, :new.fecha, :new.importe,
:new.cliente, :new.notas, 'S');
    else
        update pedidos set fecha = :new.fecha, importe = :new.importe,
cliente = :new.cliente, notas = :new.notas, especial = 'S' where código =
:new.código;
    end if;
    select avg(importe) into v_media from pedidos;
    especiales(v_media);
END;

```

Trigger TR_ESPECIAL compilado

```

-----
INSERT INTO v_pedidos VALUES('000005', '06/01/2021', 0, 'Laura', '', 'N');
INSERT INTO v_pedidos VALUES('000006', '06/01/2021', 0, 'Alicia', 'Sin gluten',
'S');

```

1 fila insertadas.

1 fila insertadas.

000000	06/01/2021	20	Pepe		S
000001	06/01/2021	30	Juan	Sin gluten	S

000005	06/01/2021	0	Laura		N
000006	06/01/2021	0	Alicia	Sin gluten	N

UPDATE v_pedidos set importe = 45 where código = '000005';

1 fila actualizadas.

000000	06/01/2021	20	Pepe		N
000001	06/01/2021	30	Juan	Sin gluten	S
000005	06/01/2021	45	Laura		S
000006	06/01/2021	0	Alicia	Sin gluten	N

El campo especial se actualiza correctamente en todas las filas. Sin embargo para mantenerlo siempre actualizado sería necesario tratar también la operación DELETE

Asimismo, también habría que modificar el disparador tr_contiene para que las modificaciones las haga sobre la vista.

-----5-----
Realizada

-----6-----
Realizada