

# Practica-Triggers-RESUELTA.pdf



**Juandf03**



**Administración de Bases de Datos**



**3º Grado en Ingeniería Informática**



**Escuela Técnica Superior de Ingeniería Informática  
Universidad de Málaga**



Estamos de  
**Aniversario**

De la universidad al  
mercado laboral:  
especialízate con los posgrados  
de EOI y marca la diferencia.



**EOI** Escuela de  
organización  
industrial



**saber más**

# Menú King Ahorro

Molar sin gastar mucho,  
mola mucho más.



Menú Cheeseburger



Menú Crispy Chicken



Menú Cheddar Lover Double Cheeseburger

PÍDE YA

```
alter session set nls_date_format = 'dd/mm/yy hh24:mi:ss';
```

```
--1
```

```
create table MENSAJES(codigo NUMBER(20) primary key, texto VARCHAR2(200));  
create table AUDITA_MENSAJES(quien VARCHAR2(20), como VARCHAR2(20), cuando  
DATE);
```

```
insert into mensajes values(1, 'Hola');  
insert into mensajes values(2, 'Hola2');  
insert into mensajes values(3, 'Hola3');  
insert into mensajes values(4, 'Hola4');
```

```
update mensajes set texto = 'Adios' where codigo = 1;  
delete from mensajes where codigo = 2;
```

```
commit;
```

```
--2
```

```
CREATE TABLE MENSAJES_Info (  
    Tipo VARCHAR2(30) PRIMARY KEY,  
    Cuantos_Mensajes NUMBER(2),  
    Ultimo VARCHAR2(200)  
);
```

```
insert into mensajes_info(tipo, cuantos_mensajes) select tipo, count(*) from mensajes group  
by tipo;  
commit;
```

```
--3
```

```
CREATE TABLE MENSAJES_TEXTO as select codigo, texto from mensajes;  
CREATE TABLE MENSAJES_TIPO as select codigo, tipo from mensajes;
```

```
DROP TABLE MENSAJES;
```

```
CREATE VIEW MENSAJES AS  
SELECT MT.codigo, MT.texto, MTI.tipo  
FROM MENSAJES_TEXTO MT  
JOIN MENSAJES_TIPO MTI ON MT.codigo = MTI.codigo;
```

```
select * from mensajes;  
insert into mensajes values(17, 'Mensajes sobre vista', 'AYUDA');
```

WUOLAH

--4

```
CREATE TABLE MENSAJES_BORRADOS AS SELECT * FROM MENSAJES_TEXTO;
```

```
DELETE FROM MENSAJES_TEXTO WHERE codigo = 10;
```

```
commit;
```

```
ALTER TABLE MENSAJES_BORRADOS ADD fecha_borrado DATE;
```

--5

```
BEGIN
```

```
  DBMS_SCHEDULER.CREATE_JOB (
```

```
    job_name      => 'borrar_mensajes_borrados_job',
```

```
    job_type      => 'PLSQL_BLOCK',
```

```
    job_action     => 'BEGIN
```

```
      DELETE FROM MENSAJES_BORRADOS WHERE fecha_borrado <
```

```
SYSTIMESTAMP - INTERVAL "2" MINUTE;
```

```
    END;',
```

```
    start_date     => SYSTIMESTAMP,
```

```
    repeat_interval => 'FREQ=MINUTELY; INTERVAL=2',
```

```
    enabled        => TRUE
```

```
  );
```

```
  COMMIT;
```

```
END;
```

```
/
```

```
CREATE OR REPLACE TRIGGER TR_MENSAJES
```

```
AFTER INSERT OR UPDATE OR DELETE ON MENSAJES
```

```
BEGIN
```

```
  IF INSERTING THEN
```

```
    INSERT INTO AUDITA_MENSAJES VALUES (USER, 'INSERT' , SYSDATE);
```

```
  ELSIF UPDATING THEN
```

```
    INSERT INTO AUDITA_MENSAJES VALUES (USER, 'UPDATE' , SYSDATE);
```

```
  ELSIF DELETING THEN
```

```
    INSERT INTO AUDITA_MENSAJES VALUES (USER, 'DELETE' , SYSDATE);
```

```
  END IF;
```

```
END;  
/
```

```
CREATE OR REPLACE TRIGGER TR_INFO_MENSAJES  
BEFORE INSERT OR DELETE OR UPDATE ON MENSAJES  
FOR EACH ROW  
BEGIN  
    IF inserting THEN  
        UPDATE mensajes_info SET cuantos_mensajes = cuantos_mensajes + 1,  
            ultimo = :new.texto WHERE tipo = :new.tipo;  
  
    ELSIF deleting THEN  
        UPDATE mensajes_info SET cuantos_mensajes = cuantos_mensajes - 1,  
            ultimo = NULL WHERE tipo = :old.tipo;  
  
    ELSE  
        IF :old.tipo != :new.tipo THEN  
            UPDATE mensajes_info SET cuantos_mensajes = cuantos_mensajes - 1,  
                ultimo = NULL WHERE tipo = :old.tipo;  
  
            UPDATE mensajes_info SET cuantos_mensajes = cuantos_mensajes + 1,  
                ultimo = :new.texto WHERE tipo = :new.tipo;  
        END IF;  
    END IF;  
END;  
/
```

```
CREATE OR REPLACE TRIGGER TR_MENSAJES2  
INSTEAD OF DELETE OR INSERT OR UPDATE ON MENSAJES  
BEGIN  
    IF inserting then  
        insert into mensajes_texto values(:new.codigo, :new.texto);  
        insert into mensajes_tipo values(:new.codigo, :new.tipo);  
    END IF;  
END;  
/
```

```
CREATE OR REPLACE TRIGGER TR_MENSAJES3  
BEFORE DELETE ON MENSAJES_TEXTO  
FOR EACH ROW
```

# ESTUDIA MEJOR con las Flashcards 2.0

Recuadro de color + poderes digitales

Fichas de estudio que te permitirán mejorar tu productividad



Organiza por color



Escanea tus  
Flashcards con



scribzee

y Repasa  
desde tu móvil

Sesiones de  
estudio eficientes  
con Scribzee.



Ideales para:

Pregunta/  
Respuesta  
Conceptos clave  
Vocabulario  
Resúmenes  
Esquemas  
Mapas mentales



BEGIN

INSERT INTO MENSAJES\_BORRADOS (codigo, texto, fecha\_borrado)

VALUES (:old.codigo, :old.texto, SYSTIMESTAMP);

END;

/

WUOLAH