

EJERCICIO 1

A)

La tabla auditoría no tiene clave primaria porque no hay manera de identificar a una tupla de manera unívoca. Por ejemplo, supongamos que dos personas deciden realizar un pedido simultáneamente. Entonces se realizará la operación de inserción casi simultáneamente en la tabla pedidos y en la tabla auditoría será la misma tupla que está repetida dos veces, pues el campo hora no tiene la suficiente precisión.

Código:

```
CREATE TABLE pedidos (  
    código CHAR(6) PRIMARY KEY,  
    fecha CHAR(10),  
    importe NUMBER(6,2),  
    cliente CHAR(20),  
    notas CHAR(1024),  
    especial CHAR(1),  
    CHECK (especial IN ('S','N')),  
    CHECK(fecha LIKE '___/___/___')  
);
```

```
CREATE TABLE contiene(  
    pedido CHAR(6) REFERENCES pedidos(código),  
    plato CHAR(20),  
    precio NUMBER(6,2) DEFAULT 0,  
    unidades NUMBER(2,0) DEFAULT 0,  
    PRIMARY KEY(pedido, plato),  
    CHECK(precio >= 0 AND unidades >= 0)  
);
```

```
CREATE TABLE auditoría (  
    operación CHAR(6),  
    tabla CHAR(50),
```

```
fecha CHAR(10),  
hora CHAR(8),  
CHECK(fecha LIKE '__/__/____'),  
CHECK(hora LIKE '__:__:__')  
);
```

Output:

Table PEDIDOS creado.

Table AUDITORÍA creado.

Table CONTIENE creado.

Código de creación de tuplas:

```
INSERT INTO PEDIDOS VALUES ('0', '18/11/2022', 10, 'Laura Macías', '', 'S');  
INSERT INTO AUDITORÍA VALUES ('INSERT', 'PEDIDOS', '18/11/2022', '13:25:00');  
INSERT INTO CONTIENE VALUES ('0', 'Tortilla de patata', 5, 2);  
INSERT INTO AUDITORÍA VALUES ('INSERT', 'CONTIENE', '18/11/2022', '13:25:00');
```

```
INSERT INTO PEDIDOS VALUES ('1', '18/11/2022', 52.8, 'Leonardo Rodrigo', 'Esto es una nota',  
'N');  
INSERT INTO AUDITORÍA VALUES ('INSERT', 'PEDIDOS', '18/11/2022', '13:30:00');  
INSERT INTO CONTIENE VALUES ('1', 'Jamón', 50, 1);  
INSERT INTO AUDITORÍA VALUES ('INSERT', 'CONTIENE', '18/11/2022', '13:30:00');  
INSERT INTO CONTIENE VALUES ('1', 'Pan', 2.8, 1);  
INSERT INTO AUDITORÍA VALUES ('INSERT', 'CONTIENE', '18/11/2022', '13:33:00');
```

Output:

	❖ CÓDIGO	❖ FECHA	❖ IMPORTE	❖ CLIENTE	❖ NOTAS
1	0	18/11/2022	10	Laura Macías	(null)
2	1	18/11/2022	52,8	Leonardo Rodrigo	Esto es una nota

	❖ PEDIDO	❖ PLATO	❖ PRECIO	❖ UNIDADES
1	0	Tortilla de patata	5	2
2	1	Jamón	50	1
3	1	Pan	2,8	1

	❖ OPERACIÓN	❖ TABLA	❖ FECHA	❖ HORA
1	INSERT	PEDIDOS	18/11/2022	13:25:00
2	INSERT	CONTIENE	18/11/2022	13:25:00
3	INSERT	PEDIDOS	18/11/2022	13:30:00
4	INSERT	CONTIENE	18/11/2022	13:30:00
5	INSERT	CONTIENE	18/11/2022	13:33:00

B)

Código:

```
CREATE OR REPLACE TRIGGER tr_pedidos
AFTER INSERT OR UPDATE OR DELETE ON pedidos
--STATEMENT POR DEFECTO
DECLARE
v_operación CHAR(6) := 'INSERT';
BEGIN
IF UPDATING THEN
v_operación := 'UPDATE';
ELSIF DELETING THEN
v_operación := 'DELETE';
END IF;

INSERT INTO AUDITORÍA VALUES (v_operación, 'PEDIDOS', to_char(sysdate,'dd/mm/yyyy'),
to_char(sysdate,'hh:mi:ss'));

END;
```

Output:

```
Trigger TR_PEDIDOS compilado
```

Probamos a insertar un nuevo pedido:

Código:

```
INSERT INTO PEDIDOS VALUES ('2', '18/11/2022', 1, 'Fernando Sáenz', '', 'S');
```

Vemos que se ha creado una nueva tupla en la tabla auditoría

	OPERACIÓN	TABLA	FECHA	HORA
1	INSERT	PEDIDOS	18/11/2022	13:25:00
2	INSERT	CONTIENE	18/11/2022	13:25:00
3	INSERT	PEDIDOS	18/11/2022	13:30:00
4	INSERT	CONTIENE	18/11/2022	13:30:00
5	INSERT	CONTIENE	18/11/2022	13:33:00
6	INSERT	PEDIDOS	18/11/2022	01:46:44

```
UPDATE PEDIDOS SET notas = 'Esto se ha añadido más tarde' WHERE código = '2';
```

	CÓDIGO	FECHA	IMPORTE	CLIENTE	NOTAS
1	0	18/11/2022	10	Laura Macías	(null)
2	1	18/11/2022	52,8	Leonardo Rodrigo	Esto es una nota
3	2	18/11/2022	1	Fernando Sáenz	Esto se ha añadido más tarde

	OPERACIÓN	TABLA	FECHA	HORA
1	INSERT	PEDIDOS	18/11/2022	13:25:00
2	INSERT	CONTIENE	18/11/2022	13:25:00
3	INSERT	PEDIDOS	18/11/2022	13:30:00
4	INSERT	CONTIENE	18/11/2022	13:30:00
5	INSERT	CONTIENE	18/11/2022	13:33:00
6	INSERT	PEDIDOS	18/11/2022	01:46:44
7	UPDATE	PEDIDOS	18/11/2022	01:50:32

```
DELETE FROM PEDIDOS WHERE código = '2';
```

	CÓDIGO	FECHA	IMPORTE	CLIENTE	NOTAS
1	0	18/11/2022	10	Laura Macías	(null)
2	1	18/11/2022	52,8	Leonardo Rodrigo	Esto es una nota

	OPERACIÓN	TABLA	FECHA	HORA
1	INSERT	PEDIDOS	18/11/2022	13:25:00
2	INSERT	CONTIENE	18/11/2022	13:25:00
3	INSERT	PEDIDOS	18/11/2022	13:30:00
4	INSERT	CONTIENE	18/11/2022	13:30:00
5	INSERT	CONTIENE	18/11/2022	13:33:00
6	INSERT	PEDIDOS	18/11/2022	01:46:44
7	UPDATE	PEDIDOS	18/11/2022	01:50:32
8	DELETE	PEDIDOS	18/11/2022	01:52:06

EJERCICIO 2

Código:

```
CREATE OR REPLACE TRIGGER tr_contiene
AFTER INSERT OR UPDATE OR DELETE ON contiene
FOR EACH ROW
DECLARE
v_importe CHAR(6);
v_pedido CHAR(6);
BEGIN
IF UPDATING OR INSERTING THEN
SELECT importe
INTO v_importe
FROM PEDIDOS
WHERE código = :NEW.pedido;

ELSE
SELECT importe
INTO v_importe
FROM PEDIDOS
WHERE código = :OLD.pedido;
END IF;

IF UPDATING THEN
IF :OLD.pedido <> :NEW.pedido THEN
```

```
UPDATE PEDIDOS SET importe = importe - (:OLD.precio*OLD.unidades) where código = :OLD.pedido;
```

```
    v_importe := v_importe + :NEW.precio*NEW.unidades;
```

```
ELSE
```

```
    v_importe := v_importe - :OLD.precio*OLD.unidades + :NEW.precio*NEW.unidades;
```

```
END IF;
```

```
v_pedido := :NEW.pedido;
```

```
ELSIF INSERTING THEN
```

```
    v_importe := v_importe + :NEW.precio*NEW.unidades;
```

```
    v_pedido := :NEW.pedido;
```

```
ELSIF DELETING THEN
```

```
    v_importe := v_importe - :OLD.precio*OLD.unidades;
```

```
    v_pedido := :OLD.pedido;
```

```
END IF;
```

```
UPDATE PEDIDOS SET importe = v_importe where código = v_pedido;
```

```
END;
```

Vamos a hacer algunas pruebas.

Código:

```
INSERT INTO CONTIENE VALUES ('0', 'Gazpacho', 4.5, 2);
```

```
DELETE FROM CONTIENE WHERE pedido = '1' AND plato = 'Pan';
```

Output:

	PEDIDO	PLATO	PRECIO	UNIDADES
1	0	Tortilla de patata	5	2
2	1	Jamón	50	1
3	0	Gazpacho	4,5	2

	❖ CÓDIGO	❖ FECHA	❖ IMPORTE	❖ CLIENTE	❖ NOTAS
1	0	18/11/2022	19	Laura Macías	(null)
2	1	18/11/2022	50	Leonardo Rodrigo	Esto es una nota

EJERCICIO 4

Realizado