

Examen Automatización de tareas

SGBD 2ºASIR

Nombre: Andrés Tenllado Pérez, Base de datos: catalogo

Ejercicio 1 :

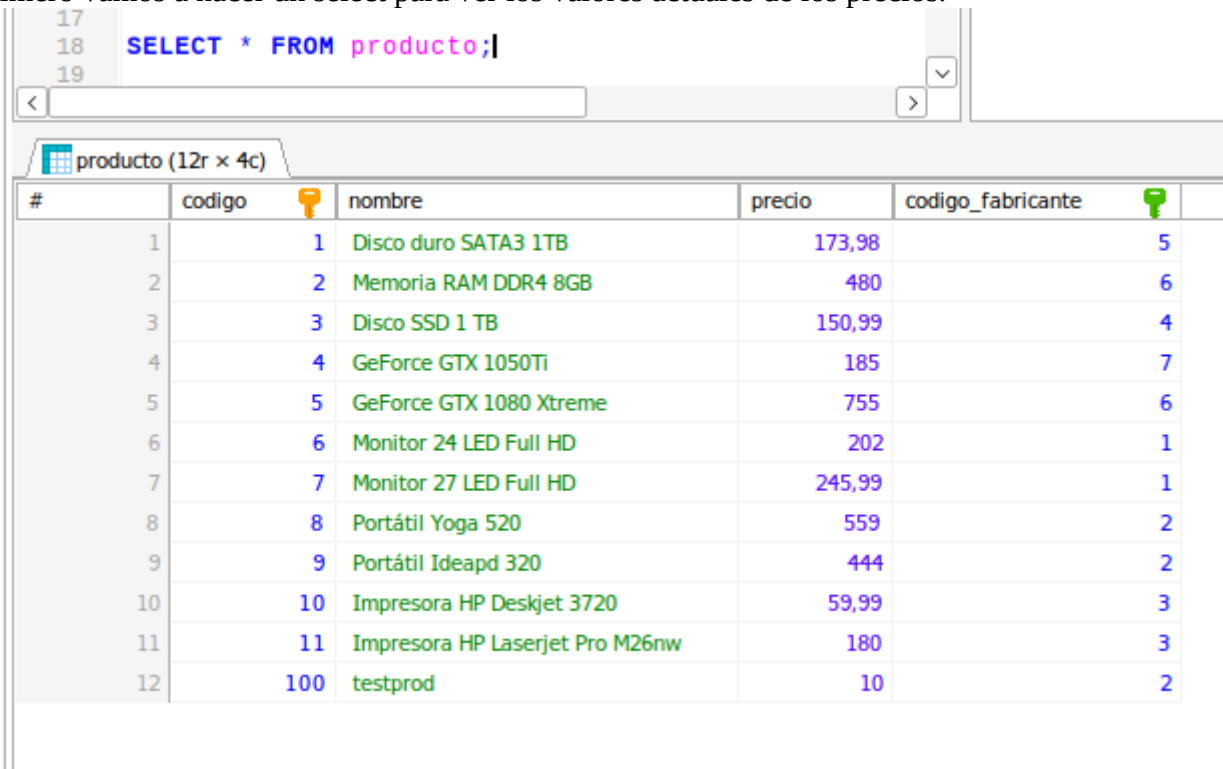
La idea es actualizar el precio con el porcentaje a incrementar para ello hemos usado un update y actualizamos el precio multiplicando por $(1 + (\text{porcentaje}/100))$

Codigo:

```
delimiter //  
  
CREATE PROCEDURE ej1(  
  IN cod VARCHAR(50),  
  IN porcentaje DECIMAL(10,2)  
)  
BEGIN  
  UPDATE producto  
  SET precio = precio * (1 + (porcentaje/100))  
  WHERE codigo LIKE cod;  
END //  
delimiter ;
```

Comprobacion:

Primero vamos a hacer un select para ver los valores actuales de los precios.



The screenshot shows a database management interface. At the top, a SQL query is entered in a text area: `SELECT * FROM producto;`. Below the query, a table titled "producto (12r x 4c)" displays the results. The table has five columns: "#", "codigo", "nombre", "precio", and "codigo_fabricante". The data is as follows:

#	codigo	nombre	precio	codigo_fabricante
1	1	Disco duro SATA3 1TB	173,98	5
2	2	Memoria RAM DDR4 8GB	480	6
3	3	Disco SSD 1 TB	150,99	4
4	4	GeForce GTX 1050Ti	185	7
5	5	GeForce GTX 1080 Xtreme	755	6
6	6	Monitor 24 LED Full HD	202	1
7	7	Monitor 27 LED Full HD	245,99	1
8	8	Portátil Yoga 520	559	2
9	9	Portátil Ideapad 320	444	2
10	10	Impresora HP Deskjet 3720	59,99	3
11	11	Impresora HP Laserjet Pro M26nw	180	3
12	100	testprod	10	2

Ahora vamos a ejecutar la llamada al procedimiento para incrementar el precio del producto 6 al doble y luego volvemos a hacer el select para ver si ha cambiado:

```
17
18 SELECT * FROM producto;
19
20 CALL ej1('6',100);
21
```

#	codigo	nombre	precio	codigo_fabricante
1	1	Disco duro SATA3 1TB	173,98	5
2	2	Memoria RAM DDR4 8GB	480	6
3	3	Disco SSD 1 TB	150,99	4
4	4	GeForce GTX 1050Ti	185	7
5	5	GeForce GTX 1080 Xtreme	755	6
6	6	Monitor 24 LED Full HD	404	1
7	7	Monitor 27 LED Full HD	245,99	1
8	8	Portátil Yoga 520	559	2
9	9	Portátil Ideapd 320	444	2
10	10	Impresora HP Deskjet 3720	59,99	3
11	11	Impresora HP Laserjet Pro M26nw	180	3
12	100	testprod	10	2

Como podemos comprobar el procedimiento funciona correctamente.

Ejercicio 2:

Lo que tenemos que hacer es guardar el precio del objeto que coincide con el codigo en una variable y si supera los 500 euros devolvemos TRUE y en caso contrario FALSE.

Codigo:

```
delimiter //
CREATE FUNCTION ej2(cod VARCHAR(50))
RETURNS BOOL
DETERMINISTIC
BEGIN
    DECLARE n_precio DECIMAL(10,2);

    SELECT precio INTO n_precio FROM producto
    WHERE codigo LIKE cod;

    IF n_precio > 500 THEN
        RETURN TRUE;
    ELSE
        RETURN FALSE;
    END IF;
END //
delimiter ;
```

Comprobacion:

Vamos a mirar los codigos 8 (vale mas de 500) y 9 (vale menos de 500)

```
42
43 SELECT ej2('8') AS Resultado;
44
```

#	Resultado
1	1

```
42
43 SELECT ej2('9') AS Resultado;
44
```

#	Resultado
1	0

El valor 1 representar TRUE y el valor 0 representa FALSE.

Ejercicio 3:

Para crear el trigger tenemos que contar el numero de productos que tienen el mismo codigo_fabricante que el producto nuevo que se va a insertar, caso de ser superior a 100 no se inserta y sale el mensaje de error.

```
delimiter //
CREATE TRIGGER ej3
BEFORE INSERT ON producto
FOR EACH ROW
BEGIN
    DECLARE total INT;

    SELECT COUNT(*) INTO total FROM producto
    WHERE codigo_fabricante = NEW.codigo_fabricante;

    IF total >= 100 THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Error. Ese fabricante ya suministra 100
productos';
    END IF;

END //
delimiter ;
```

Comprobacion:

Para comprobar que funcione en vez de 100 he cambiado el valor a 3 porque no hay muchas entradas en la base de datos. Para comprobar que funciona vamos a insertar productos con el mismo codigo de fabricante hasta que salte el error. El código de fabricante esta presente 2 veces osea que si insertamos 2 veces mas dara error.

```
70
71 INSERT INTO producto
72 VALUES(
73 101,
74 'testprod',
75 10,
76 '1'
77 );
```

Filtro:

```
341 /* Error de SQL (1304): PROCEDURE ej1 already exists */
342 /* */
343 /* Filas afectadas: 0 Filas encontradas: 0 Advertencias: 0 Duración para 1 de 12 consultas: 0,000 seg. */
344 CREATE TRIGGER ej3 BEFORE INSERT on producto FOR EACH ROW BEGIN DECLARE total INT; SELECT COUNT(*)
345 /* */
346 /* Filas afectadas: 0 Filas encontradas: 0 Advertencias: 0 Duración para 1 consulta: 0,009 seg. */
347 INSERT INTO producto VALUES(101,'testprod',10,'1');
348 /* */
349 /* Filas afectadas: 1 Filas encontradas: 0 Advertencias: 0 Duración para 1 consulta: 0,007 seg. */
```

r71 : c1 (1,2 KiB) Conectado: 02 MySQL 8.0.41 Activo durante: 02:14 h Hora del servid Preparado.

Volvemos a intentar insertar

```
70
71 INSERT INTO producto
72 VALUES(
73 102,
74 'testprod',
75 10,
76 '1'
77 );
```

Unnamed: Error

Error de SQL (1644): Error. Ese fabricante ya suministra 100 productos

Aceptar

Encontrar ayuda acerca de este error (=> [ecosia.org](https://dev.mysql.com/doc/err/en/))

Filtro:

```
382 SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS` WHERE `EVENT_SCHEMA`='
383 /* El nodo de tabla "ej3" no se encontró en el árbol. */
384 /* Entrando a la sesión "Unnamed" */
385 CREATE TRIGGER ej3 BEFORE INSERT on producto FOR EACH ROW BEGIN DECLARE total INT; SELECT COUNT(*)
386 /* */
387 /* Filas afectadas: 0 Filas encontradas: 0 Advertencias: 0 Duración para 1 consulta: 0,010 seg. */
388 INSERT INTO producto VALUES(102,'testprod',10,'1');
389 /* Error de SQL (1644): Error. Ese fabricante ya suministra 100 productos */
390 /* */
```

r71 : c1 (1,2 KiB) Conectado: 02 MySQL 8.0.41 Activo durante: 02:15 h Hora del servid 00:00:07.9: Ejecutando

Podemos observar como se ha producido el error. Ahora vamos a ver si de verdad no se ha insertado el producto con código 102.

17

18

19

SELECT * FROM producto;

<

>

▼

producto (13r x 4c)

#	codigo	nombre	precio	codigo_fabricante
1	1	Disco duro SATA3 1TB	173,98	5
2	2	Memoria RAM DDR4 8GB	480	6
3	3	Disco SSD 1 TB	150,99	4
4	4	GeForce GTX 1050Ti	185	7
5	5	GeForce GTX 1080 Xtreme	755	6
6	6	Monitor 24 LED Full HD	404	1
7	7	Monitor 27 LED Full HD	245,99	1
8	8	Portátil Yoga 520	559	2
9	9	Portátil Ideapd 320	444	2
10	10	Impresora HP Deskjet 3720	59,99	3
11	11	Impresora HP Laserjet Pro M26nw	180	3
12	100	testprod	10	2
13	101	testprod	10	1

Como podemos ver no se ha insertado.