

Esta obra está licenciada bajo la Licencia Creative Commons Atribución 4.0 Internacional. Para ver una copia de esta licencia, visite http://creativecommons.org/licenses/by/4.0/ o envíe una carta a Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

## Índice

Actividad propuesta1
Actividades realizadas1
Tabla Clientes2
1. Creación de copia y columnas nuevas2
2. Creación de triggers3
Insert3
Update3
Delete4
3. Realización de pruebas5
Prueba "Insert"5
Prueba "Update"5
Prueba "Delete"6
Tabla Detalle_factura7
1. Creación de copia y columnas nuevas
2. Creación de triggers
Insert7
Update8
Delete8
3. Realización de pruebas9
Prueba "Insert"9
Prueba "Update"9
Prueba "Delete"10
Tabla Facturas11

1. Creación de copia y columnas nuevas	11
2. Creación de triggers	12
Insert	12
Update	12
Delete	13
3. Realización de pruebas	13
Prueba "Insert"	13
Prueba "Update"	13
Prueba "Delete"	14
Tabla Productos	15
1. Creación de copia y columnas nuevas	15
2. Creación de triggers	15
Insert	15
Update	16
Delete	16
3. Realización de pruebas	16
Nota previa	16
Pruebas "insert" y "update"	17
Pruebas "delete"	18

## Actividad propuesta.

En la base de datos de la ferretería los muchachos crea un trigger para cada una de las tablas, en sus tres versiones INSERT, UPDATE, DELETE.

Recuerda los pasos:

- 1. Crea la tabla copia
- 2. Crea el trigger
- 3. Prueba insertando, modificando o eliminando un registro de la tabla original en cada caso

## Actividades realizadas.

Para la realización de las actividades propuestas se llevaron a cabo los siguientes pasos, los cuales estarán agrupados por tablas, es decir, por cada tabla se estará mostrando la creación de la tabla de respaldo, la creación de los trigger o disparadores correspondientes y sus respectivas pruebas.

El funcionamiento de las tablas va a ser muy sencillo, las tablas nuevas tendrán las mismas columnas de las tablas originales más tres columnas nuevas que son fechaEvento, usuarioEvento y evento, y en ellas se recogerán los datos de los distintos eventos llevados a cabo en la tabla original, en el caso de los eventos de insertar la nueva tabla solo recogerá el identificador del registro insertado, la fecha en que se llevó a cabo la inserción, el usuario que la llevó a cabo y el evento que se registra, ya que los datos del registro como tal estarán en la tabla de origen del evento, por otra parte, al llevar a cabo una actualización nuestra tabla de logs recogerá todos los valores anteriores del registro además de la fecha, usuario y evento, por último si el registro es borrado de la tabla de origen se creará un registro en la tabla de logs en donde se mantendrán todos los datos que tenía la tupla así como la fecha y el usuario que lo borró. Usando este sistema podremos disponer de todos los cambios que se hayan realizado sobre un registro.

Para aquellos datos que ya existían se procedió a copiarlos de manera integra y podrán ser reconocidos por que sus columnas fechaEvento, usuarioEvento y evento tendán valores nulos.

## Tabla Clientes.

## 1. Creación de copia y columnas nuevas.

```
CREATE TABLE `triggers_los_muchachos`.`clientes_logs`(
    `Id` INT(2) DEFAULT NULL,
    `NIF` VARCHAR(12) DEFAULT NULL,
    `Nombre` VARCHAR(29) DEFAULT NULL,
    `Dirección` VARCHAR(39) DEFAULT NULL,
    `CP` INT(5) DEFAULT NULL,
    `Población` VARCHAR(10) DEFAULT NULL,
    `Provincia` VARCHAR(22) DEFAULT NULL,
    `Teléfono` INT(9) DEFAULT NULL
);
INSERT INTO `triggers_los_muchachos`.`clientes_logs`(
    `NIF`,
    `Nombre`,
    `Dirección`,
    `CP`,
    `Población`,
    `Provincia`,
    `Teléfono`
SELECT
    `Id`,
    `NIF`,
    `Nombre`,
    `Dirección`,
    `CP`,
    `Población`,
    `Provincia`,
    `Teléfono`
FROM
    `triggers_los_muchachos`.`clientes`;
ALTER TABLE
    `clientes_logs`
    ADD `fechaEvento` DATETIME NOT NULL AFTER `Teléfono`,
    ADD `usuarioEvento` VARCHAR(100) NOT NULL AFTER `fechaEvento`,
    ADD `evento` VARCHAR(10) NOT NULL AFTER `usuarioEvento`;
```

## 2. Creación de triggers.

#### Insert

Volver al índice

#### **Update**

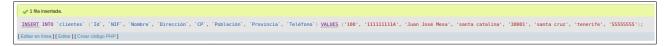
```
-- Trigger para los eventos update.
DELIMITER
CREATE TRIGGER clientes_AU AFTER UPDATE
    clientes FOR EACH ROW
    INSERT INTO clientes_logs(
        `Id`,
`NIF`,
        `Nombre`,
        `Dirección`,
        `CP`,
        `Población`,
         `Provincia`,
        `Teléfono`,
        `fechaEvento`,
        `usuarioEvento`,
        `evento`
VALUES(
    OLD. `Id`,
    OLD. `NIF`,
    OLD. `Nombre`,
    OLD. `Dirección`,
    OLD. `CP`,
    OLD. `Población`,
    OLD.`Provincia`,
    OLD. `Teléfono`,
    NOW(), CURRENT_USER(), 'UPDATED');
    END &&
DELIMITER
```

#### **Delete**

```
-- Trigger para los eventos delete.
DELIMITER
    &&
CREATE TRIGGER clientes_AD AFTER DELETE
    clientes FOR EACH ROW
    INSERT INTO clientes_logs(
        `Id`,
        `NIF`,
        `Nombre`,
        `Dirección`,
        `CP`,
        `Población`,
        `Provincia`,
        `Teléfono`,
        `fechaEvento`,
        `usuarioEvento`,
        `evento`
    )
VALUES(
    OLD. `Id`,
    OLD. `NIF`,
    OLD. `Nombre`,
    OLD. `Dirección`,
    OLD. `CP`,
   OLD. `Población`,
    OLD.`Provincia`,
    OLD. `Teléfono`,
    NOW(), CURRENT_USER(), 'DELETED');
    END &&
DELIMITER
```

## 3. Realización de pruebas.

#### Prueba "Insert"

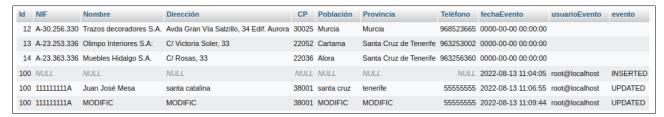


ld	NI	IF	Nombre	Dirección	СР	Población	Provincia	Teléfono	fechaEvento	usuarioEvento	evento
1	2 A-	30.256.330	Trazos decoradores S.A.	Avda Gran Via Salzillo, 34 Edif. Aurora	30025	Murcia	Murcia	968523665	0000-00-00 00:00:00		
1	3 A-	23.253.336	Olimpo Interiores S.A:	C/ Victoria Soler, 33	22052	Cartama	Santa Cruz de Tenerife	963253002	0000-00-00 00:00:00		
1	4 A-	23.363.336	Muebles Hidalgo S.A.	C/ Rosas, 33	22036	Alora	Santa Cruz de Tenerife	963256360	0000-00-00 00:00:00		
10	00 N	ULL	NULL	NULL	NULL	NULL	NULL	NULL	2022-08-13 11:04:05	root@localhost	INSERTED

#### Volver al índice

### Prueba "Update"

ld	NIF	Nombre	Dirección	СР	Población	Provincia	Teléfono	fechaEvento	usuarioEvento	evento
12	A-30.256.330	Trazos decoradores S.A.	Avda Gran Via Salzillo, 34 Edif. Aurora	30025	Murcia	Murcia	968523665	00:00:00		
13	A-23.253.336	Olimpo Interiores S.A:	C/ Victoria Soler, 33	22052	Cartama	Santa Cruz de Tenerife	963253002	00:00:00		
14	A-23.363.336	Muebles Hidalgo S.A.	C/ Rosas, 33	22036	Alora	Santa Cruz de Tenerife	963256360	00:00:00		
100	NULL	NULL	NULL	NULL	NULL	NULL	NULL	2022-08-13 11:04:05	root@localhost	INSERTED
100	111111111A	Juan José Mesa	santa catalina	38001	santa cruz	tenerife	5555555	2022-08-13 11:06:55	root@localhost	UPDATED



#### Volver al índice

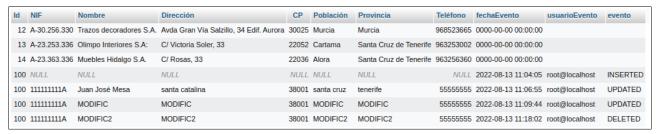
Podemos observar que solo registramos los valores viejos en la tabla logs ya que los valores actualizados estarán en el registro de la tabla de origen.

#### Prueba "Delete"

```
✓ 1 fila afectada. (La consulta tardó 0.0005 segundos.)

DELETE FROM `clientes` WHERE id = 100;

[Editar en línea] [Editar] [Crear código PHP]
```



## Tabla Detalle\_factura.

#### 1. Creación de copia y columnas nuevas.

```
CREATE TABLE `triggers_los_muchachos`.`detalle_factura_logs`(
    `Id` INT(1) DEFAULT NULL,
    `Codigo_producto` VARCHAR(5) DEFAULT NULL,
    `Cantidad` INT(3) DEFAULT NULL
INSERT INTO `triggers_los_muchachos`.`detalle_factura_logs`(`Id`, `Codigo_producto`,
`Cantidad`)
SELECT
    `Id`,
    `Codigo_producto`,
    `Cantidad`
    `triggers_los_muchachos`.`detalle_factura`;
ALTER TABLE
    `detalle_factura_logs`
    ADD `fechaEvento` DATETIME NOT NULL AFTER `Cantidad`,
    ADD `usuarioEvento` VARCHAR(100) NOT NULL AFTER `fechaEvento`,
    ADD `evento` VARCHAR(10) NOT NULL AFTER `usuarioEvento`;
```

Volver al índice

## 2. Creación de triggers.

#### Insert

```
-- Trigger para evento insert

DELIMITER

&&

CREATE TRIGGER detalle_factura_AI AFTER INSERT ON

detalle_factura FOR EACH ROW

BEGIN

INSERT INTO detalle_factura_logs(`Id`, `fechaEvento`, `usuarioEvento`, `evento`
)

VALUES(

NEW.Id,

NOW(), CURRENT_USER, 'INSERTED');

END &&

DELIMITER

;
```

## **Update**

Volver al índice

#### **Delete**

```
-- Trigger para los eventos delete.
DELIMITER
CREATE TRIGGER detalle_factura_AD AFTER DELETE
   detalle_factura FOR EACH ROW
BEGIN
    INSERT INTO detalle_factura_logs(
        `Id`,
        `Codigo_producto`,
        `Cantidad`,
        `fechaEvento`,
        `usuarioEvento`,
        `evento`
   )
VALUES(
   OLD. `Id`,
   OLD. `Codigo_producto`,
   OLD. `Cantidad`,
   NOW(),
   CURRENT_USER(),
    'DELETED');
END &&
DELIMITER
```

## 3. Realización de pruebas.

#### Prueba "Insert"

I				
8 PPB-1	100	00:00:00-00-00:00:00		
9 PPB-1	100	0000-00-00 00:00:00		
100 NULL	NULL	2022-08-13 12:20:43 roo	ot@localhost	INSERTED

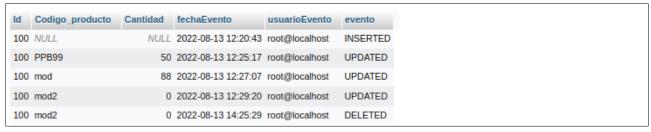
Volver al índice

## Prueba "Update"

8 PPB-1	100 0000-00-00 00:00:00	
9 PPB-1	100 0000-00-00 00:00:00	
100 NULL	NULL 2022-08-13 12:20:43 root@localhost	INSERTED
100 PPB99	50 2022-08-13 12:25:17 root@localhost	UPDATED

ld	Codigo_producto	Cantidad	fechaEvento	usuarioEvento	evento
100	mod	88	2022-08-13 12:27:07	root@localhost	UPDATED
100	mod2	0	2022-08-13 12:29:20	root@localhost	UPDATED

#### Prueba "Delete"



## Tabla Facturas.

## 1. Creación de copia y columnas nuevas.

```
CREATE TABLE `triggers_los_muchachos`.`facturas_logs`(
    `Id` INT(1) DEFAULT NULL,
    `fecha` VARCHAR(8) DEFAULT NULL,
    `Id _cliente` INT(2) DEFAULT NULL,
    `Descuento` INT(2) DEFAULT NULL,
    `iva` INT(1) DEFAULT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8; SET
    SQL_MODE = 'NO_AUTO_VALUE_ON_ZERO';
INSERT INTO `triggers_los_muchachos`.`facturas_logs`(
    `Id`,
    `fecha`,
    `Id _cliente`,
    `Descuento`,
    `iva`
SELECT
    `Id`,
    `fecha`,
    `Id _cliente`,
    `Descuento`,
    `iva`
    `triggers_los_muchachos`.`facturas`;
ALTER TABLE
    facturas_logs ADD `fechaEvento` DATETIME NOT NULL,
    ADD `usuarioEvento` VARCHAR(100) NOT NULL,
    ADD `evento` VARCHAR(10) NOT NULL;
```

## 2. Creación de triggers.

#### Insert

```
- Trigger para evento insert
DELIMITER
CREATE TRIGGER facturas_AI AFTER INSERT ON
   facturas FOR EACH ROW
   INSERT INTO facturas_logs(
        `Id`,
        `fechaEvento`,
        `usuarioEvento`,
        `evento`
VALUES(
   NEW.Id,
   NOW(),
   CURRENT_USER,
    'INSERTED');
END &&
DELIMITER
```

Volver al índice

#### **Update**

```
-- Trigger para los eventos update.

DELIMITER

&&

CREATE TRIGGER facturas_AU AFTER UPDATE

ON

facturas FOR EACH ROW

BEGIN

INSERT INTO facturas_logs('Id', 'fecha', 'Id _cliente', 'Descuento', 'iva',
'fechaEvento', 'usuarioEvento', 'evento'
)

VALUES(

OLD. `Id',
OLD. `Id _,
OLD. `Id _cliente',
OLD. `Descuento',
OLD. `iva',
NOW(),
CURRENT_USER(),
'UPDATED');
END &&
DELIMITER

;
```

#### Delete

```
-- Trigger para los eventos delete.

DELIMITER

&&

CREATE TRIGGER facturas_AD AFTER DELETE

ON

facturas FOR EACH ROW

BEGIN

INSERT INTO facturas_logs(`Id`, `fecha`, `Id _cliente`, `Descuento`, `iva`, `fechaEvento`, `usuarioEvento`, `evento`

)

VALUES(OLD.`Id`, OLD.`fecha`, OLD.`Id _cliente`, OLD.`Descuento`, OLD.`iva`, NOW(), CURRENT_USER(), 'DELETED');

END &&

DELIMITER

;
```

Volver al índice

## 3. Realización de pruebas.

#### Prueba "Insert"

```
7 25-03-22 7 5 7 0000-00-00 00:00:00
8 25-03-22 5 5 7 0000-00-00 00:00:00
9 25-03-22 5 25 7 0000-00-00 00:00:00
100 NULL NULL NULL NULL 2022-08-13 15:22:44 root@localhost INSERTED
```

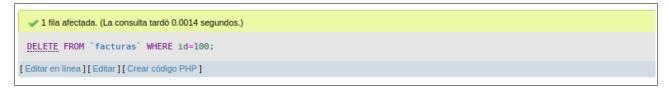
Volver al índice

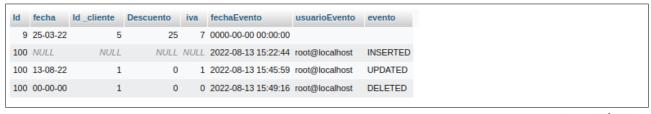
## Prueba "Update"

8 25-03-22	5	5	7	0000-00-00 00:00:00		
9 25-03-22	5	25	7	0000-00-00 00:00:00		
100 NULL	NULL	NULL	NULL	2022-08-13 15:22:44	root@localhost	INSERTED
100 13-08-22	1	0	1	2022-08-13 15:45:59	root@localhost	UPDATED

Volver al índice

#### Prueba "Delete"





#### **Tabla Productos.**

#### 1. Creación de copia y columnas nuevas.

```
CREATE TABLE `triggers_los_muchachos`.`productos_logs`(
    `Código` VARCHAR(5) DEFAULT NULL,
    `Producto` VARCHAR(32) DEFAULT NULL,
    `Precio` VARCHAR(5) DEFAULT NULL
) ENGINE = InnoDB DEFAULT CHARSET = utf8; SET
   SQL_MODE = 'NO_AUTO_VALUE_ON_ZERO';
INSERT INTO `triggers_los_muchachos`.`productos_logs`(`Código`, `Producto`, `Precio`)
SELECT
    `Código`
    `Producto`,
    `Precio`
    `triggers_los_muchachos`.`productos`;
ALTER TABLE
   productos_logs ADD `fechaEvento` DATETIME NOT NULL,
    ADD `usuarioEvento` VARCHAR(100) NOT NULL,
   ADD `evento` VARCHAR(10) NOT NULL;
```

Volver al índice

## 2. Creación de triggers.

#### Insert

```
-- Trigger para evento insert

DELIMITER

&&

CREATE TRIGGER productos_AI AFTER INSERT ON

productos FOR EACH ROW

BEGIN

INSERT INTO productos_logs(`Código`, `fechaEvento`, `usuarioEvento`, `evento`)

VALUES(

NEW.Código,

NOW(),

CURRENT_USER,

'INSERTED');

END &&

DELIMITER

;
```

#### **Update**

Volver al índice

#### Delete

```
-- Trigger para los eventos delete.

DELIMITER

&&

CREATE TRIGGER productos_AD AFTER DELETE

ON

productos FOR EACH ROW

BEGIN

INSERT INTO productos_logs(`Código`, `Producto`, `Precio`, `fechaEvento`, `usuarioEvento`, `evento`

)

VALUES(OLD.`Código`, OLD.`Producto`, OLD.`Precio`, NOW(), CURRENT_USER(), 'DELETED');

END &&

DELIMITER

;
```

Volver al índice

## 3. Realización de pruebas.

## Nota previa.

Al momento de hacer esta inserción me percaté que el campo precio estaba definido como varchar, aceptando en consecuencia, un formato errado, además se están expresando los decimales con una coma en lugar de ser

con un punto que sería la sintaxis correcta por lo que las pruebas de insert y update estarán entremezcladas ya que haremos un update del campo precio usando la instrucción replace y luego un cambio del tipo en la columna correspondiente.

Volver al índice

#### Pruebas "insert" y "update"

```
✓ 9 filas afectadas. (La consulta tardó 0.0005 segundos.)

UPDATE `productos` SET `Precio`= Replace(Precio,',','.');

[Editar en línea] [Editar] [Crear código PHP]
```

```
Código
         Producto
                                      Precio
PPB-1
         Pintura plástica blanca 1 Kg.
                                     3.3
PPB-2
         Pintura plástica blanca 5 Kg.
                                     6.45
PPB-3 Pintura plástica blanca 10 Kg 11.3
PPA-1 Pintura plástica azul 5
                                     8.65
PPA-2
         Pintura plástica azul 10 Kg. 15.4
PPM-1 Pintura plástica amarillo 5 Kg. 10.45
PPM-2 Pintura plástica amarillo 10 Kg. 19.4
PAB-1 Pintura acrilica blanca 1 Kg
                                     8.45
PAR-1
         Pintura acrílica roja 1 Kg
                                     15.4
PAN-1
         Pintura Acrilica Negra 1 Kg
```

```
✓ Los cambios en la Tabla productos se hicieron exitosamente.

ALTER TABLE `productos` CHANGE `Precio` `Precio` DECIMAL(5,2) NULL DEFAULT NULL;

[Editar en línea][Editar][Crear código PHP]
```

I MIV-T	i intura acrinca roja 1 Ng	13.40 0000-00-00 00.00.00	
PAN-1	NULL	NULL 2022-08-13 22:40:58 root@localhost INS	SERTED
PPB-1	Pintura plástica blanca 1 Kg.	3.30 2022-08-13 22:57:19 root@localhost UP	DATED
PPB-2	Pintura plástica blanca 5 Kg.	6.45 2022-08-13 22:57:19 root@localhost UP	DATED
PPB-3	Pintura plástica blanca 10 Kg	11.30 2022-08-13 22:57:19 root@localhost UP	DATED
PPA-1	Pintura plástica azul 5	8.65 2022-08-13 22:57:19 root@localhost UP	DATED
PPA-2	Pintura plástica azul 10 Kg.	15.40 2022-08-13 22:57:19 root@localhost UP	DATED
PPM-1	Pintura plástica amarillo 5 Kg.	10.45 2022-08-13 22:57:19 root@localhost UP	DATED
PPM-2	Pintura plástica amarillo 10 Kg.	19.40 2022-08-13 22:57:19 root@localhost UP	DATED
PAB-1	Pintura acrílica blanca 1 Kg	8.45 2022-08-13 22:57:19 root@localhost UP	DATED
PAR-1	Pintura acrílica roja 1 Kg	15.40 2022-08-13 22:57:19 root@localhost UP	DATED
PAN-1	Pintura Acrilica Negra 1 Kg	15.40 2022-08-13 22:57:19 root@localhost UP	DATED

## Volver al índice

## Pruebas "delete"

PPA-2	Pintura plástica azul 10 Kg.	15.40 2022-08-13 22:57:19 root@localhost	UPDATED
PPM-1	Pintura plástica amarillo 5 Kg.	10.45 2022-08-13 22:57:19 root@localhost	UPDATED
PPM-2	Pintura plástica amarillo 10 Kg.	19.40 2022-08-13 22:57:19 root@localhost	UPDATED
PAB-1	Pintura acrilica blanca 1 Kg	8.45 2022-08-13 22:57:19 root@localhost	UPDATED
PAR-1	Pintura acrilica roja 1 Kg	15.40 2022-08-13 22:57:19 root@localhost	UPDATED
PAN-1	Pintura Acrílica Negra 1 Kg	15.40 2022-08-13 22:57:19 root@localhost	UPDATED
PAN-1	Pintura Acrilica Negra 1 Kg	15.40 2022-08-13 23:13:04 root@localhost	DELETED