# Instituto Tecnológico Superior de Jerez.



**Jerez de García Salinas a 15 de Noviembre del 2019.**

**Cristofer Casas Murillo.**

cristofer32513@gmail.com

**S17070157.**

**INGENIERÍA EN SISTEMAS COMPUTACIONALES.**

**Taller de Base de Datos.**

**5to. SEMESTRE.**

**Triggers SQL.**

**ISC. Salvador Acevedo Sandoval.**

Iniciamos el servidor con el usuario root.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u root -p

C:\Users\casas>mysql -u root -p
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.17 MySQL Community Server - GPL

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Creamos un nuevo usuario.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u root -p
mysql> CREATE USER 'Cristofer'@'localhost' IDENTIFIED BY 'casas';
Query OK, 0 rows affected (0.77 sec)

mysql>
```

Agregamos privilegios al usuario creado anteriormente.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u root -p
mysql> GRANT ALL PRIVILEGES ON * . * TO 'Cristofer'@'localhost';
Query OK, 0 rows affected (0.30 sec)

mysql>
```

Actualizamos los privilegios.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u root -p
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.04 sec)

mysql>
```

Cerramos la sesión del servidor del  usuario root.

```
C:\WINDOWS\system32\cmd.exe
mysql> EXIT
Bye

C:\Users\casas>
```

# Practica 1.

Iniciamos sesión con el usuario creado anteriormente.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
C:\Users\casas>mysql -u Cristofer -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.17 MySQL Community Server - GPL

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Creamos la base de datos.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE DATABASE base_ejemplo;
Query OK, 1 row affected (0.22 sec)

mysql>
```

Seleccionamos la base de datos.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> USE base_ejemplo;
Database changed
mysql>
```

Creamos la tabla.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE productos (
    ->     id INT NOT NULL AUTO_INCREMENT,
    ->     nombre VARCHAR(20) NOT NULL,
    ->     coste FLOAT NOT NULL DEFAULT 0.0,
    ->     precio FLOAT NOT NULL DEFAULT 0.0,
    ->     PRIMARY KEY(id)
    -> );
Query OK, 0 rows affected (1.07 sec)

mysql>
```

Agregamos inserciones de prueba.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> INSERT INTO productos (nombre, coste, precio) VALUES ('Producto A', 4, 8), ('Producto B', 2, 4), ('Producto C', 40, 80);
Query OK, 3 rows affected (0.22 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql>
```

Creamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELIMITER $$
mysql> CREATE TRIGGER actualizarPrecioProducto
    -> BEFORE UPDATE ON productos
    -> FOR EACH ROW
    -> BEGIN
    ->   IF NEW.coste <> OLD.coste
    ->     THEN
    ->       SET NEW.precio = NEW.coste * 2;
    ->   END IF ;
    -> END$$
Query OK, 0 rows affected (0.71 sec)

mysql> DELIMITER ;
mysql>
```

Probamos el trigger.

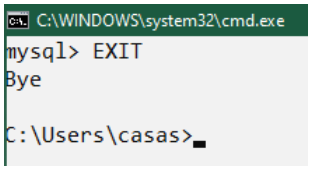
```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> UPDATE productos SET coste = 5 WHERE id = 1;
Query OK, 1 row affected (0.35 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> SELECT * FROM productos;
+----+------------+-------+--------+
| id | nombre     | coste | precio |
+----+------------+-------+--------+
|  1 | Producto A |     5 |     10 |
|  2 | Producto B |     2 |      4 |
|  3 | Producto C |    40 |     80 |
+----+------------+-------+--------+
3 rows in set (0.01 sec)

mysql>
```

Borramos el trigger.

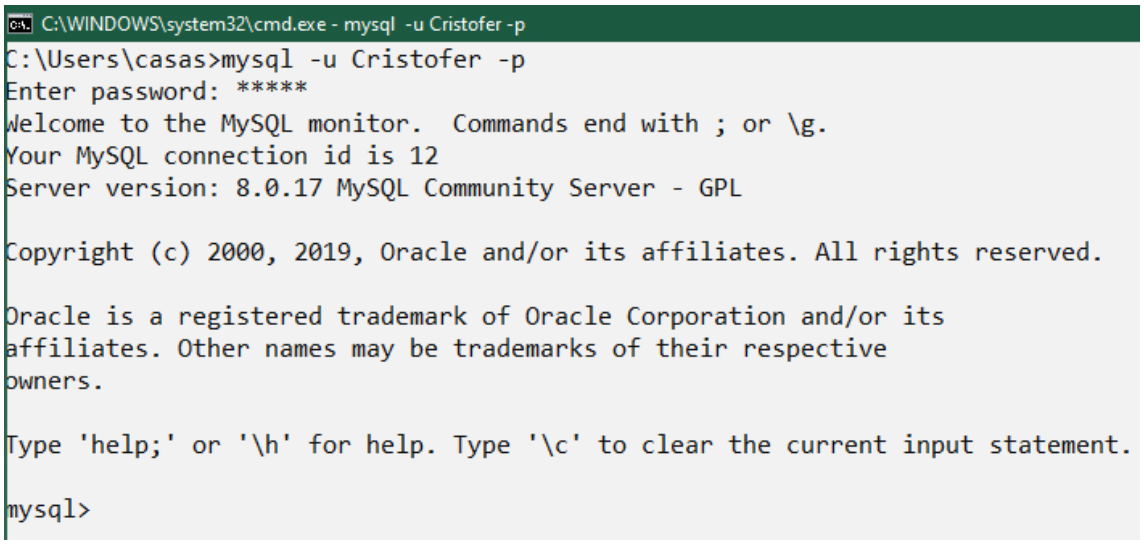
```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DROP TRIGGER actualizarPrecioProducto;
Query OK, 0 rows affected (0.16 sec)

mysql>
```

Cerramos sesión del servidor.

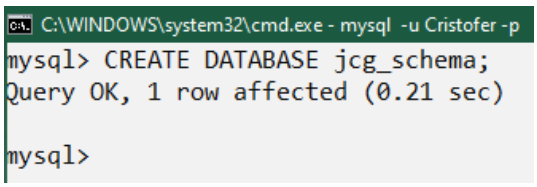



# Practica 2

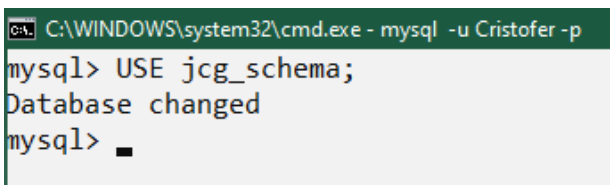
Iniciamos sesión con el usuario creado anteriormente.




Creamos la base de datos.




Seleccionamos la base de datos.



Creamos las tablas.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE author_audit (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     authorId INT NOT NULL,
    ->     name VARCHAR(50) NOT NULL,
    ->     changedate DATETIME DEFAULT NULL,
    ->     action VARCHAR(50) DEFAULT NULL
    -> );
Query OK, 0 rows affected (0.60 sec)

mysql>
```

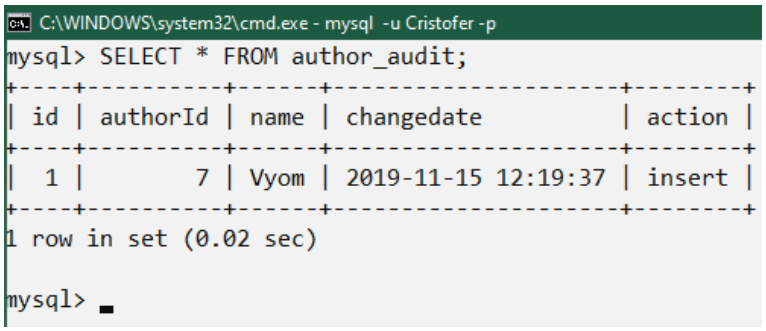
```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE author (
    -> id INT PRIMARY KEY,
    -> name VARCHAR(50) NOT NULL,
    -> post_count INT NOT NULL
    -> );
Query OK, 0 rows affected (0.76 sec)

mysql>
```

Creamos el trigger.

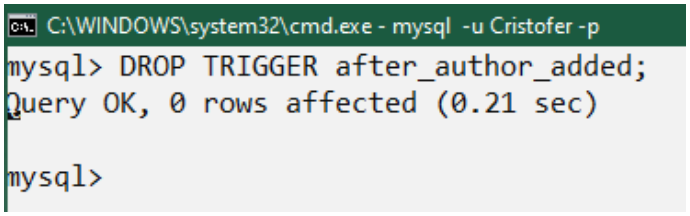
```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELIMITER $$
mysql> CREATE TRIGGER after_author_added
    -> AFTER INSERT ON Author
    -> FOR EACH ROW
    -> BEGIN
    ->     INSERT INTO author_audit
    ->     SET action = 'insert',
    ->         authorId = NEW.id,
    ->         name = NEW.name,
    ->         changedate = NOW();
    -> END$$
Query OK, 0 rows affected (0.18 sec)

mysql> DELIMITER ;
mysql>
```

Probamos el trigger.

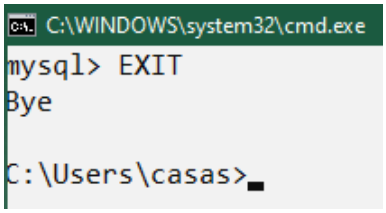
```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> INSERT INTO Author (id, name, post_count) VALUES (7, 'Vyom', 27);
Query OK, 1 row affected (0.18 sec)

mysql>
```




Borramos el trigger.

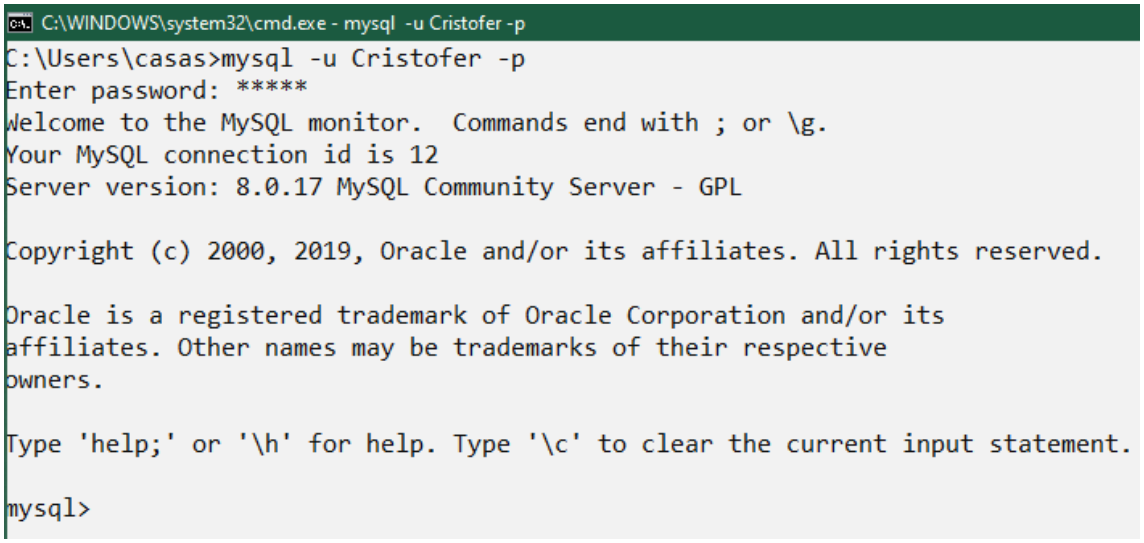



Cerramos sesión del servidor.




# Practica 3

Iniciamos sesión con el usuario creado anteriormente.



Creamos la base de datos.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE DATABASE practica3;
Query OK, 1 row affected (0.70 sec)

mysql>
```

Seleccionamos la base de datos.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> USE practica3;
Database changed
mysql>
```

**1.- Antes del UPDATE:**

Creamos las tablas.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE customer (
    -> acc_no INTEGER PRIMARY KEY,
    -> cust_name VARCHAR(20),
    -> avail_balance DECIMAL);
Query OK, 0 rows affected (0.53 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE mini_statement (
    -> acc_no INTEGER,
    -> avail_balance DECIMAL,
    -> FOREIGN KEY(acc_no)
    -> REFERENCES customer(acc_no)
    -> ON DELETE CASCADE);
Query OK, 0 rows affected (0.57 sec)

mysql>
```

Insertamos registros.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> INSERT INTO customer VALUES (1000, "Fanny", 7000);
Query OK, 1 row affected (0.18 sec)

mysql> INSERT INTO customer VALUES (1001, "Peter", 12000);
Query OK, 1 row affected (0.15 sec)

mysql>
```

Creamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELIMITER //
mysql> CREATE TRIGGER update_cus
    -> BEFORE UPDATE ON customer
    -> FOR EACH ROW
    -> BEGIN
    -> INSERT INTO mini_statement VALUES (OLD.acc_no, OLD.avail_balance);
    -> END //
Query OK, 0 rows affected (0.18 sec)

mysql> DELIMITER ;
mysql>
```

Probamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> UPDATE customer SET avail_balance = avail_balance + 3000 WHERE acc_no = 1001;
Query OK, 1 row affected (0.19 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> UPDATE customer SET avail_balance = avail_balance + 3000 WHERE acc_no = 1000;
Query OK, 1 row affected (0.25 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> SELECT * FROM mini_statement;
+--------+---------------+
| acc_no | avail_balance |
+--------+---------------+
|   1001 |         12000 |
|   1000 |          7000 |
+--------+---------------+
2 rows in set (0.10 sec)

mysql>
```

**2.- Después del UPDATE:**

Creamos la tabla.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE micro_statement (
    -> acc_no INTEGER,
    -> avail_balance DECIMAL,
    -> FOREIGN KEY(acc_no)
    -> REFERENCES customer(acc_no)
    -> ON DELETE CASCADE);
Query OK, 0 rows affected (0.81 sec)

mysql>
```

Insertamos un registro en la tabla “customer”.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> INSERT INTO customer VALUES (1002, "Janitor", 4500);
Query OK, 1 row affected (0.09 sec)

mysql>
```

Creamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELIMITER //
mysql> CREATE TRIGGER update_after
    -> AFTER UPDATE ON customer
    -> FOR EACH ROW
    -> BEGIN
    -> INSERT INTO micro_statement VALUES(new.acc_no, new.avail_balance);
    -> END //
Query OK, 0 rows affected (0.25 sec)

mysql> DELIMITER ;
mysql>
```

Probamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> UPDATE customer SET avail_balance = avail_balance + 1500 WHERE acc_no = 1002;
Query OK, 1 row affected (0.14 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> SELECT * FROM micro_statement;
+--------+---------------+
| acc_no | avail_balance |
+--------+---------------+
|   1002 |          6000 |
+--------+---------------+
1 row in set (0.00 sec)

mysql>
```

### 3.- Antes del INSERT:

Crear la tabla.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE contacts (
    -> contact_id INT (11) NOT NULL AUTO_INCREMENT,
    -> last_name VARCHAR (30) NOT NULL,
    -> first_name VARCHAR (25),
    -> birthday DATE,
    -> created_date DATE,
    -> created_by VARCHAR(30),
    -> CONSTRAINT contacts_pk
    -> PRIMARY KEY (contact_id));
Query OK, 0 rows affected, 1 warning (0.73 sec)

mysql>
```

Crear el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELIMITER //
mysql> CREATE TRIGGER contacts_before_insert
    -> BEFORE INSERT ON contacts
    -> FOR EACH ROW
    -> BEGIN
    -> DECLARE vUser VARCHAR(50);
    -> SELECT USER() INTO vUser;
    -> SET NEW.created_date = SYSDATE();
    -> SET NEW.created_by = vUser;
    -> END //
Query OK, 0 rows affected (0.55 sec)

mysql> DELIMITER ;
mysql>
```

Probar el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> INSERT INTO contacts VALUES (1, "Newton", "Enigma",
    -> str_to_date ("19-08-1999", "%d-%m-%Y"),
    -> str_to_date ("17-03-2018", "%d-%m-%Y"), "xyz");
Query OK, 1 row affected (0.28 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> SELECT * FROM contacts;
+------------+-----------+------------+------------+--------------+----------------------+
| contact_id | last_name | first_name | birthday   | created_date | created_by           |
+------------+-----------+------------+------------+--------------+----------------------+
|          1 | Newton    | Enigma     | 1999-08-19 | 2019-11-15   | Cristofer@localhost |
+------------+-----------+------------+------------+--------------+----------------------+
1 row in set (0.00 sec)

mysql>
```

**4.- Después del INSERT:**

Creamos las tablas.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE contacts2(
    -> contact_id int (11) NOT NULL AUTO_INCREMENT,
    -> last_name VARCHAR(30) NOT NULL,
    -> first_name VARCHAR(25),
    -> birthday DATE,
    -> CONSTRAINT contacts2_pk
    -> PRIMARY KEY (contact_id)
    -> );
Query OK, 0 rows affected, 1 warning (0.45 sec)
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE contacts_audit(
    -> contact_id INTEGER,
    -> created_date DATE,
    -> created_by VARCHAR (30)
    -> );
Query OK, 0 rows affected (0.49 sec)
```

Creamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELIMITER //
mysql> CREATE TRIGGER contacts_after_insert
    -> AFTER INSERT ON contacts2
    -> FOR EACH ROW
    -> BEGIN
    -> DECLARE vUser VARCHAR(50);
    -> SELECT USER() INTO vUser;
    -> INSERT INTO contacts_audit(contact_id, created_date, created_by) VALUES(NEW.contact_id, SYSDATE(), vUser);
    -> END //
Query OK, 0 rows affected (0.18 sec)
```

Probamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> INSERT INTO contacts2 VALUES (1, "Kumar", "Rupesh", str_to_date("20-06-1999", "%d-%m-%Y"));
Query OK, 1 row affected (1.42 sec)
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> SELECT * FROM contacts_audit;
+------------+--------------+----------------------+
| contact_id | created_date | created_by           |
+------------+--------------+----------------------+
|          1 | 2019-11-15   | Cristofer@localhost |
+------------+--------------+----------------------+
1 row in set (0.00 sec)

mysql>
```

## 5.- Antes del DELETE:

Crear las tablas.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE contacts3(
    -> contact_id int (11) NOT NULL AUTO_INCREMENT,
    -> last_name VARCHAR (30) NOT NULL,
    -> first_name VARCHAR (25),
    -> birthday DATE,
    -> created_date DATE,
    -> created_by VARCHAR(30),
    -> CONSTRAINT contacts_pk
    -> PRIMARY KEY (contact_id));
Query OK, 0 rows affected, 1 warning (0.41 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE contacts_audit2(
    -> contact_id INTEGER,
    -> deleted_date DATE,
    -> deleted_by VARCHAR(20));
Query OK, 0 rows affected (0.81 sec)

mysql>
```

Crear el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELIMITER //
mysql> CREATE TRIGGER contacts_before_delete
    -> BEFORE DELETE ON contacts3
    -> FOR EACH ROW
    -> BEGIN
    -> DECLARE vUser VARCHAR(50);
    -> SELECT USER() INTO vUser;
    -> INSERT INTO contacts_audit2(contact_id, deleted_date, deleted_by)
    -> VALUES(OLD.contact_id, SYSDATE(), vUser );
    -> END //
Query OK, 0 rows affected (0.16 sec)

mysql> DELIMITER ;
mysql>
```

Insertar un registro.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> INSERT INTO contacts3 VALUES (1, "Bond", "Ruskin",
    -> str_to_date ("19-08-1995", "%d-%m-%Y"),
    -> str_to_date ("27-04-2018", "%d-%m-%Y"), "xyz");
Query OK, 1 row affected (0.18 sec)

mysql>
```

Probar el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELETE FROM contacts3 WHERE last_name="Bond";
Query OK, 1 row affected (0.09 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> SELECT * FROM contacts_audit2;
+------------+--------------+---------------------+
| contact_id | deleted_date | deleted_by          |
+------------+--------------+---------------------+
|          1 | 2019-11-15   | Cristofer@localhost |
+------------+--------------+---------------------+
1 row in set (0.00 sec)

mysql>
```

## 6.- Después del DELETE:

Creamos las tablas.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE contacts4(
    -> contact_id int (11) NOT NULL AUTO_INCREMENT,
    -> last_name VARCHAR (30) NOT NULL,
    -> first_name VARCHAR (25),
    -> birthday DATE,
    -> created_date DATE,
    -> created_by VARCHAR (30),
    -> CONSTRAINT contacts4_pk
    -> PRIMARY KEY (contact_id)
    -> );
Query OK, 0 rows affected, 1 warning (0.55 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> CREATE TABLE contacts_audit3(
    -> contact_id INTEGER,
    -> deleted_date DATE,
    -> deleted_by VARCHAR(20)
    -> );
Query OK, 0 rows affected (0.48 sec)

mysql>
```

Creamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELIMITER //
mysql> CREATE TRIGGER contacts_after_delete
    -> AFTER DELETE ON contacts4
    -> FOR EACH ROW
    -> BEGIN
    -> DECLARE vUser varchar(50);
    -> SELECT USER() INTO vUser;
    -> INSERT INTO contacts_audit3(contact_id, deleted_date, deleted_by)
    -> VALUES(OLD.contact_id, SYSDATE(), vUser );
    -> END //
Query OK, 0 rows affected (0.22 sec)

mysql> DELIMITER ;
mysql>
```

Insertamos un registro.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> INSERT INTO contacts4 VALUES (1, "Newton", "Isaac",
    -> str_to_date ("19-08-1985", "%d-%m-%Y"),
    -> str_to_date ("23-07-2018", "%d-%m-%Y"), "xyz");
Query OK, 1 row affected (0.18 sec)

mysql>
```

Probamos el trigger.

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> DELETE FROM contacts4 WHERE first_name="Isaac";
Query OK, 1 row affected (0.16 sec)

mysql>
```

```
C:\WINDOWS\system32\cmd.exe - mysql  -u Cristofer -p
mysql> SELECT * FROM contacts_audit3;
+------------+--------------+---------------------+
| contact_id | deleted_date | deleted_by          |
+------------+--------------+---------------------+
|          1 | 2019-11-15   | Cristofer@localhost |
+------------+--------------+---------------------+
1 row in set (0.00 sec)

mysql>
```

Cerramos sesión del servidor.

```
C:\WINDOWS\system32\cmd.exe
mysql> EXIT
Bye

C:\Users\casas>
```