

# Ejercicios triggers

1) Muestra los triggers de la BD sakila de dos formas diferentes. Haz lo mismo con la BD world.

```
use sakila;
show triggers;
```

#	Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_client	collation_connection	Database Collation
1	customer_create_date	INSERT	customer	SET NEW.create_date = NOW()	BEFORE	2018-05-09 15:19:12.10	STRICT_TRANS_TABLES,STRICT...	root@localhost	utf8	utf8_general_ci	latin1_swedish_ci
2	ins_film	INSERT	film	BEGIN INSERT INTO film_text (film_id, ... VALUES (new.film_id, new.titl... END	AFTER	2018-05-09 15:18:53.84	STRICT_TRANS_TABLES,STRICT...	root@localhost	utf8	utf8_general_ci	latin1_swedish_ci
3	upd_film	UPDATE	film	BEGIN IF (old.title != new.title) OR (old.... THEN UPDATE film_text	AFTER	2018-05-09 15:18:53.90	STRICT_TRANS_TABLES,STRICT...	root@localhost	utf8	utf8_general_ci	latin1_swedish_ci

```
use world;
show triggers;
```

#	Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_client	collation_connection	Database Collation

2) Con la información que muestra la tabla TRIGGERS de information\_schema, ¿Qué tipo de evento y cuando disparan los triggers customer\_create\_date, ins\_film, upd\_film, del\_film?

customer\_create\_date: Se ejecuta cuando es una insert en la tabla customer, y hace que create\_date sea la fecha y hora actual.

```
SET NEW.create_date = NOW()
```

ins\_film: Se ejecuta cuando es una insert en la tabla film y inserta en film\_text film\_id, title y description con los mismos valores que la tabla film.

```
BEGIN
INSERT INTO film_text (film_id, title, description)
VALUES (new.film_id, new.title, new.description);
END
```

upd\_film: Se ejecuta cuando es una update en la tabla film y en se actualizara en caso de que cambie el titulo o la descripción o el id la tabla film\_text con los valores anteriores

upd_film	UPDATE	film	<pre> BEGIN   IF (old.title != new.title) OR (old.description != new.description) OR (old.film_id != new.film_id)   THEN     UPDATE film_text       SET title=new.title,           description=new.description,           film_id=new.film_id;   END IF; </pre>
----------	--------	------	---

del\_film: Se ejecuta cuando es una delete en la tabla film y elimina de la tabla film\_text la fila que contuviera el mismo id que la fila eliminada

del_film	DELETE	film	<pre> BEGIN   DELETE FROM film_text WHERE film_id = old.film_id; END </pre>
----------	--------	------	---

3) Visualiza los triggers de la tabla film. ¿Cuántos tiene?

4) Construir un disparador de base de datos que permita auditar las operaciones de borrado de datos que se realicen en la tabla "actor"

4.1, 4.2

```

use sakila;
create table auditaractor (audita varchar(200));
delimiter //
create trigger borrarInsert after delete on sakila.actor
for each row
begin
  insert into auditaractor values(concat("El actor" ,old.actor_id," de apellido " ,old.last_name," fue borrado el día ",now()));
end;
delimiter ;

```

4.3

```

use sakila;
insert into actor (actor_id,first_name,last_name) values(444,"Sergio","Lopez");
insert into actor (actor_id,first_name,last_name) values(445,"Sergio","Lopez");
insert into actor (actor_id,first_name,last_name) values(446,"Sergio","Lopez");

```

```

1 • use sakila;
2 • delete from actor where actor_id=444;
3 • select * from auditaractor;

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
#	audita			
1	El actor444 de apellido Lopez fue borrado el día 2018-05-09 16:11:41			

```
mysql> delete from actor where first_name="Sergio";
Query OK, 2 rows affected (0,08 sec)

mysql> select * from auditaractor;
+-----+-----+
| audita |
+-----+-----+
| El actor444 de apellido Lopez fue borrado el día 2018-05-09 16:11:41 |
| El actor445 de apellido Lopez fue borrado el día 2018-05-09 16:14:58 |
| El actor446 de apellido Lopez fue borrado el día 2018-05-09 16:14:58 |
+-----+-----+
3 rows in set (0,00 sec)
```

5) Construir un nuevo disparador de base de datos de forma que al borrar una actor introduzca en la tabla auditaractor el mensaje 'se ha borrado un actor'. ¿Qué ocurre? ¿Porqué?

```
use sakila;
delimiter //
create trigger borrar after delete on sakila.actor
  for each row
  BEGIN
    insert into auditaractor values("se ha borrado un actor");
  END;
delimiter ;
```

1 •	use sakila;
2 •	insert into actor (actor_id,first_name,last_name) values(444,"Sergio","Lopez");
3 •	delete from actor where actor_id=444;
4 •	select * from auditaractor;

  

#	audita
1	El actor444 de apellido Lopez fue borrado el día 2018-05-09 16:11:41
2	El actor445 de apellido Lopez fue borrado el día 2018-05-09 16:14:58
3	El actor446 de apellido Lopez fue borrado el día 2018-05-09 16:14:58
4	se ha borrado un actor
5	El actor444 de apellido Lopez fue borrado el día 2018-05-09 16:27:38

Funciona los dos disparadores.

6) Busca el trigger rental\_date y explica qué es lo que hace si insertamos en la tabla rental cambia la hora a la hora actual la columna rental\_date

rental_date	INSERT	rental	SET NEW.rental_date = NOW()
-------------	--------	--------	-----------------------------