**TALLER No. 6**

**TALLER TRIGGER Y CURSOR**

**OBJETIVO:**

* Crear triggers de la base de datos
* Borrar triggers de la base datos
* Declarar y usar cursores.

**ACTIVIDADES A DESARROLLAR:**

1. Construir desencadenadores (trigger)
2. Construir un cursor

**EVIDENCIA(S) A ENTREGAR:**

**EV1 Desarrollar trigger y cursor de acuerdo a:**

Se debe entregar un documento llamado “*TALLERTRIGGERCURSOR.sql*”, que contenga:

* Creación tabla LOGSALARIO
* Creación del trigger TRGSALARIO
* Creación del procedimiento modificarSalario que contenga el cursor llamado CURSOR\_SALARIO

**CONTROL DEL DOCUMENTO**

|  | **Nombre** | **Cargo** | **Dependencia** | **Fecha** |
| --- | --- | --- | --- | --- |
| **Autor (es)** | JOSE FERNANDO GALINDO S. | Instructor | Teleinformática | 28/05/2020 |

**CONTROL DE CAMBIOS** (diligenciar únicamente si realizan ajustes al taller)

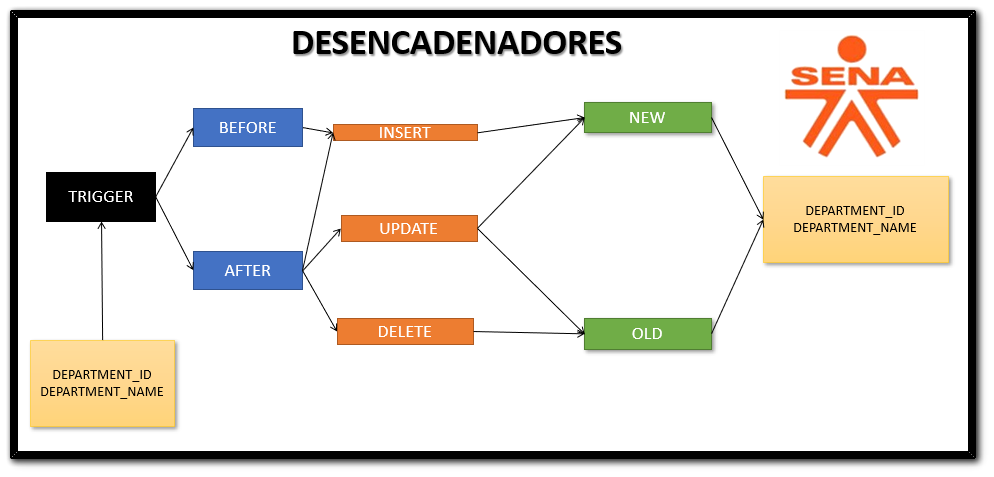
|  | **Nombre** | **Cargo** | **Dependencia** | **Fecha** | **Razón del Cambio** |
| --- | --- | --- | --- | --- | --- |
| **Autor (es)** |  |  |  |  |  |

**PRESENTACIÓN**

Los desencadenadores y cursores son sentencias que utilizan para la gestión de la información en una base de datos.

La incorporación de los trigger se comenzó a utilizar desde la versión mysql 5.0.2, son objetos con nombre asociado a una tabla y es activada cuando se realiza un evento INSERT, UPDATE o DELETE.

**DESENCADENADORES**

****

**Definición**:

| CREATE TRIGGER **Nombre Momento Evento** ON **nomre\_tabla**  FOR EACH ROW  BEGIN  … Cuerpo o llamada  END; |
| --- |

Donde:

**Momento**: **BEFORE** antes de grabar el registro en la tabla, **AFTER** despues de grabar el registro en la tabla.

**Evento**: **INSERT** un nuevo registro, **UPDATE** actualizar el registro, new.nombredecampodelatablaborrar el registro.

Acceso a los datos de la tabla de acuerdo a la DML del evento.

| **Evento** | **Toma de datos** |
| --- | --- |
| INSERT | new.nombredecampodelatabla |
| UPDATE | new.nombredecampodelatabla  old.nombredecampodelatabla |
| DELETE | old.nombredecampodelatabla |

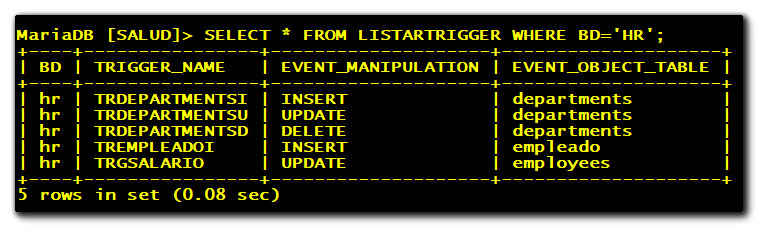
Borrar un trigger en la base de datos:

| DROP TRIGGER nombre\_trigger; |
| --- |

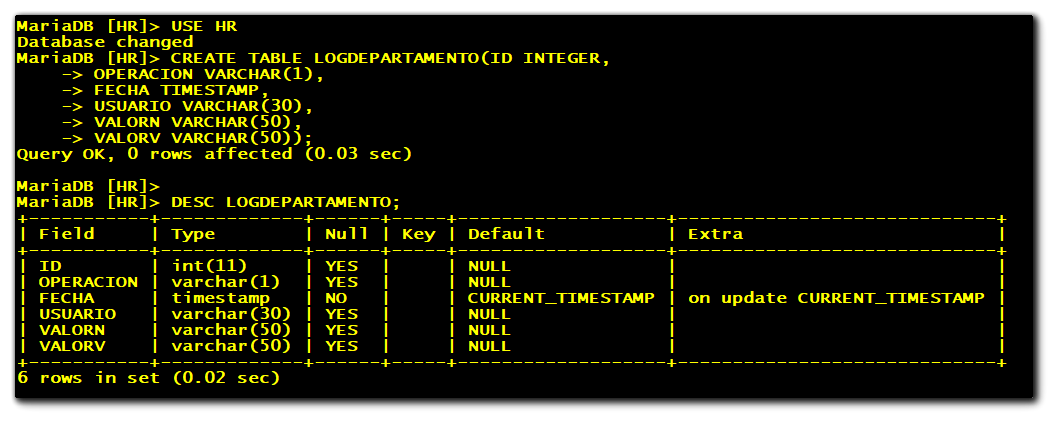
**PRACTICA DEL TALLER**

* Crear la vista “**LISTARTRIGGER**”

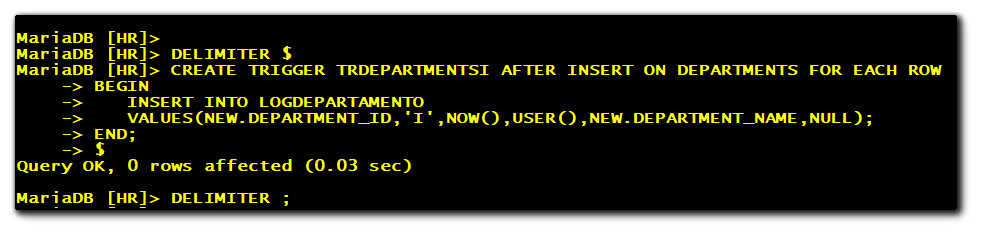
| **CREATE OR REPLACE VIEW LISTARTRIGGER AS**  **SELECT TRIGGER\_SCHEMA BD,TRIGGER\_NAME,EVENT\_MANIPULATION,EVENT\_OBJECT\_TABLE FROM INFORMATION\_SCHEMA.TRIGGERS ;** |
| --- |



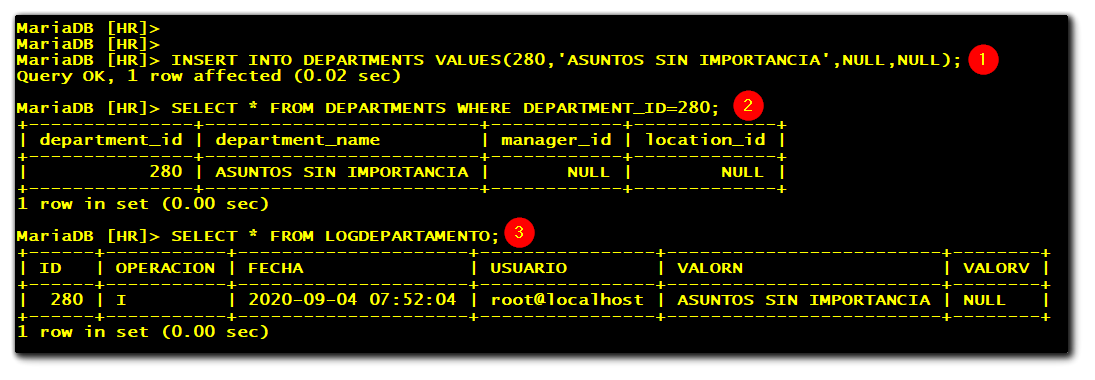
* Crear la tabla “LOGDEPARTAMENTO”



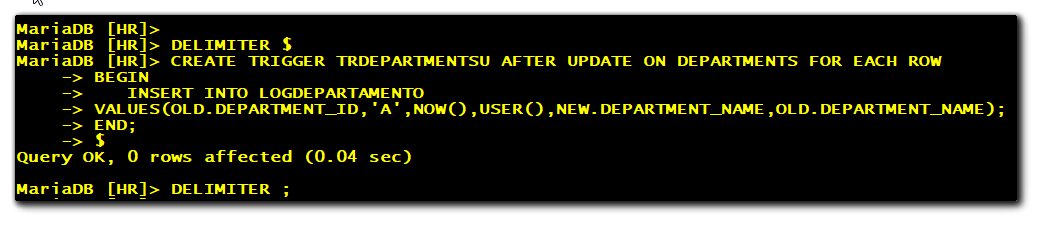
* Crear el TRIGGER para el evento INSERT



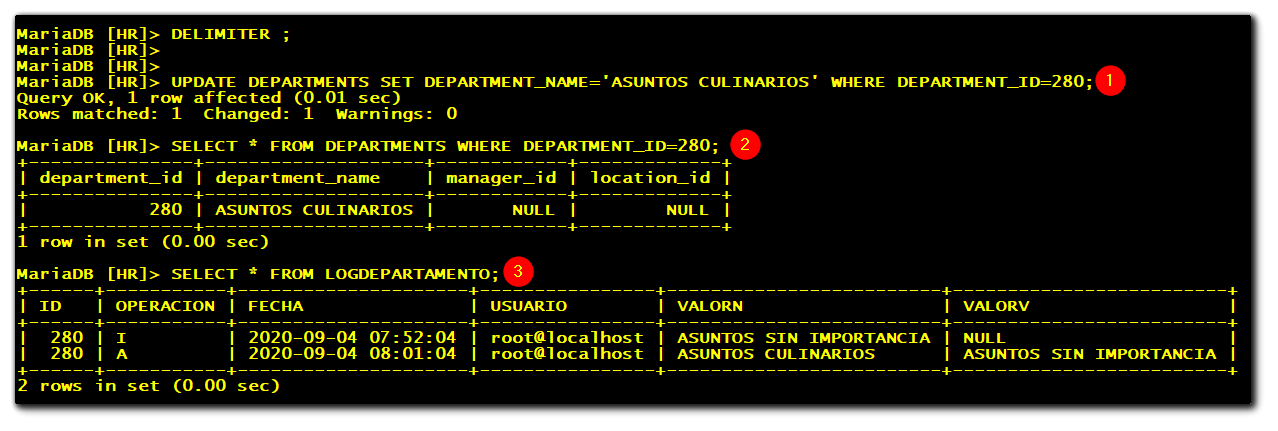
1. Insertamos en la tabla DEPARTMENTS
2. Verificamos que se haya creado en la tabla DEPARTMENTS.
3. Si el desencadenador funcionó, verificamos en la tabla LOGDEPARTAMENTO.



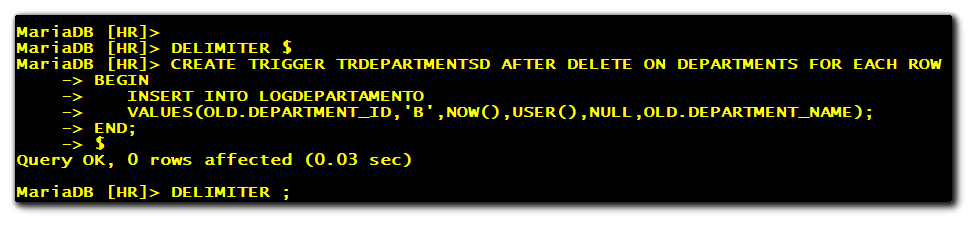
* Crear el TRIGGER para el evento UPDATE



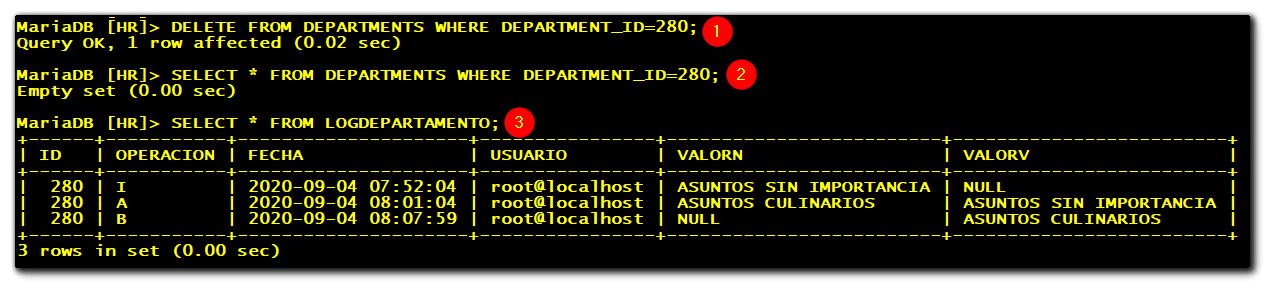
1. Actualizamos en la tabla DEPARTMENTS el registro 280, cambiamos “ASUNTOS SIN IMPORTANCIA” POR “ASUNTOS CULINARIOS”
2. Verificamos que se modificado en la tabla DEPARTMENTS.
3. Si el desencadenador funcionó, verificamos en la tabla LOGDEPARTAMENTO.

****

* Crear el TRIGGER para el evento DELETE

****

1. Borramos en la tabla DEPARTMENTS el registro 280.
2. Verificamos que se borró en la tabla DEPARTMENTS.
3. Si el desencadenador funcionó, verificamos en la tabla LOGDEPARTAMENTO.

****

**CURSORES**

Para la práctica de cursores debemos tener en cuenta las siguientes partes:

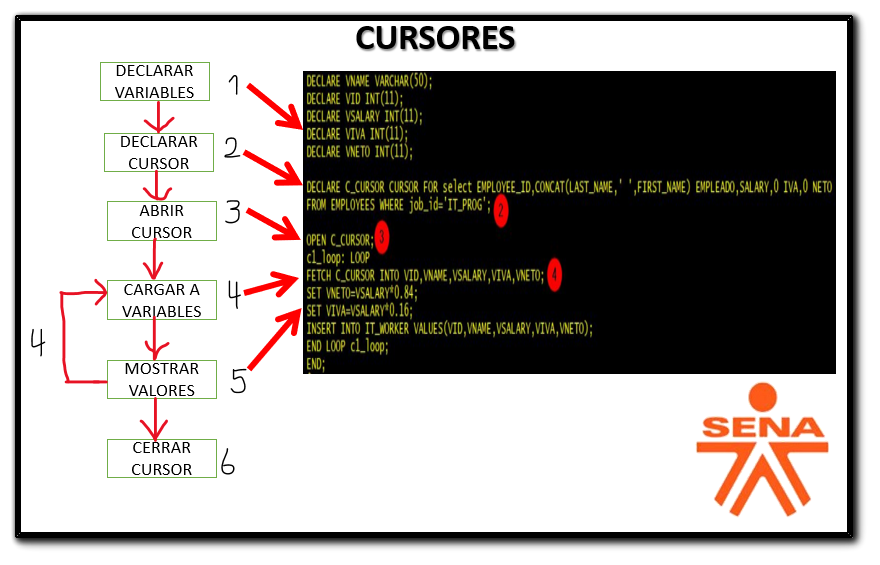
1- DECLARAR EL CURSOR

2- ABRIR EL CURSOR

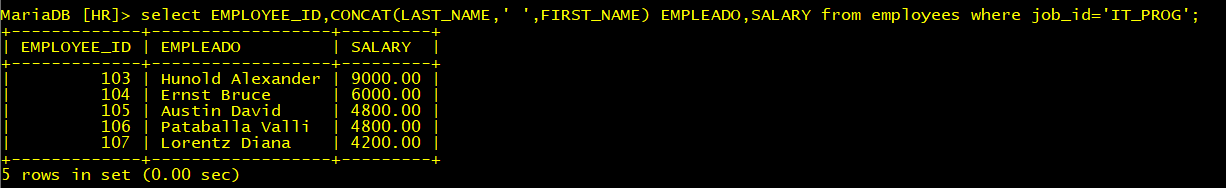
3- CARGAR DATOS

4- MOSTRAR DATOS

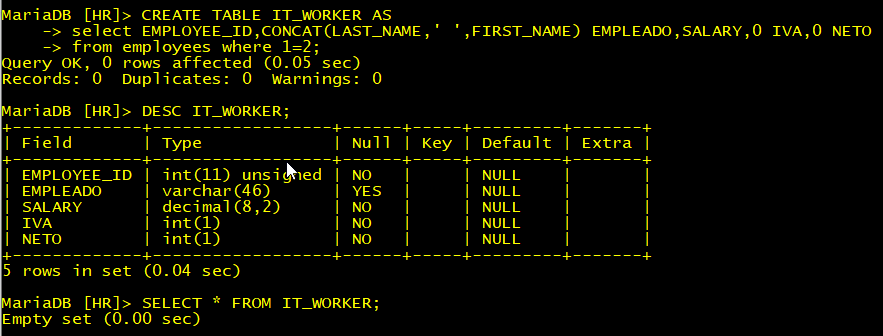
5- CERRAR EL CURSOR



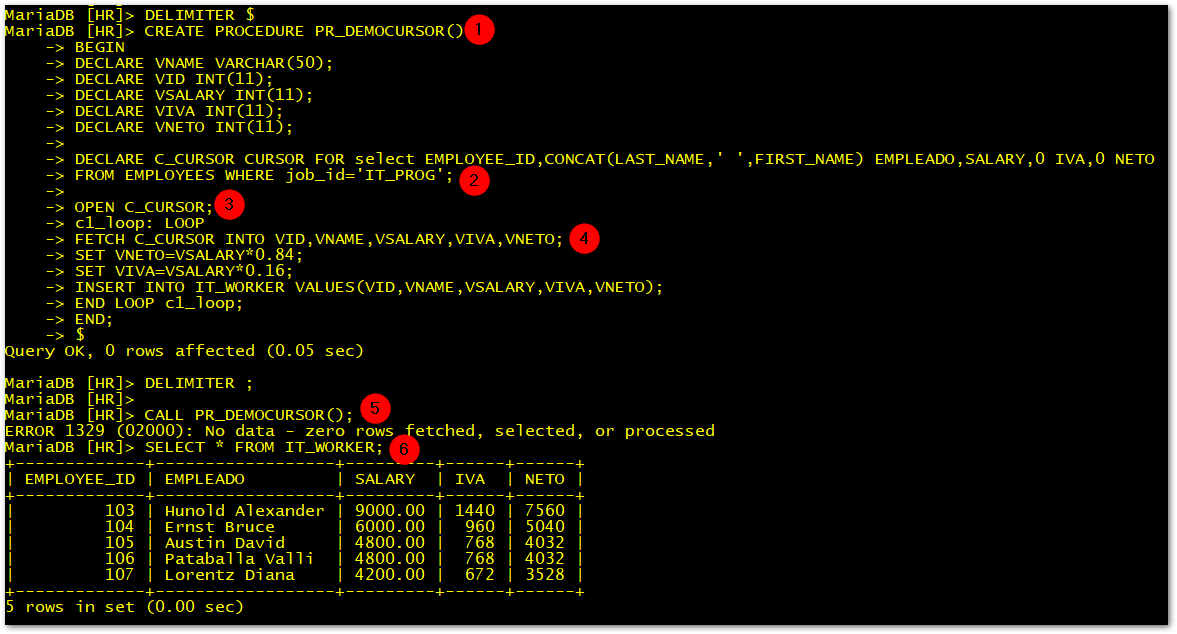
Para esto, crearemos un procedimiento llamado “PR\_DEMOCURSOR” que contiene un cursor que carga las filas de la tabla EMPLOYEES que su JOB\_ID pertenezcan a “IT\_PROG”, asi:

****

Creamos la tabla “IT\_WORKER” solo la estructura



La idea es de correr el procedimiento “PR\_DEMOCURSOR” y utilizar el cursor llamado “C\_CURSOR” que calcula el IVA y el valor neto de su salario.



**RETO A REALIZAR**

Realizar un procedimiento llamado modificarSalario que utilice un cursor con el id, el salario y el cargo que ocupa. De acuerdo al cargo se hará un incremento así:

| **CARGO** | **INCREMENTO** |
| --- | --- |
| **IT\_PROG** | **1.5%** |
| **HR\_REP** | **1.6%** |
| **SA\_REP** | **1.7%** |
| **ST\_CLERK** | **1.5%** |
| **SH\_CLERK** | **1.5%** |
| **AC\_ACCOUNT** | **1.7%** |
| **FI\_ACCOUNT** | **1.6%** |
| **El resto de cargos** | **1.2%** |

Cuando se actualice el salario en la tabla EMPLOYEES se debe construir un trigger llamado TRGSALARIO donde se guardar el id, el salario antiguo, el nuevo salario, el cargo, la fecha y el usuario que realizo la transacción en la tabla LOGSALARIO.

**Realizado por el instructor José Fernando Galindo Suárez**

[**jgalindos@sena.edu.co**](mailto:jgalindos@sena.edu.co) **2020**

