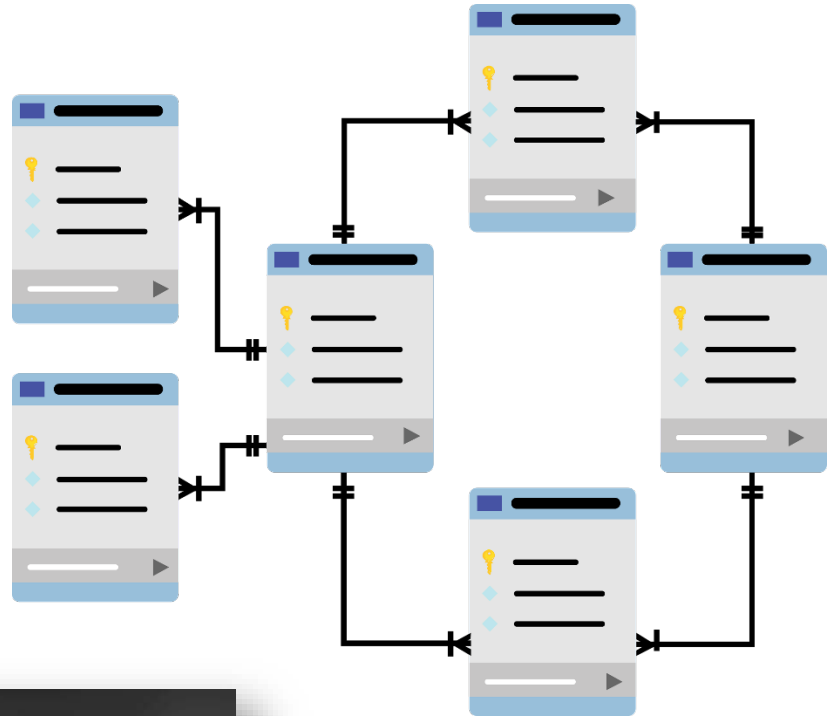


DISEÑO Y CREACIÓN

DE LA BASE DE DATOS



Bases de Datos

Proyecto Indra

Antia Loureiro Ferrío

Índice

1. Diseño de la base de datos:

- a. Modelo Entidad Relación
 - i. Herramienta utilizada*
 - ii. Esquema*
 - iii. Diccionario de datos*
- b. Tablas
 - i. De Entidad*
 - ii. De Interrelación*

2. Creación de la base de datos:

- a. Scripts de creación de la base de datos
- b. Visualización de la estructura de la base de datos
 - i. Modo gráfico*
 - ii. Usando cmd*



Modelo EER
Y
Creación de la BD



1. Diseño de la base de datos:

a. Modelo Entidad Relación

i. *Herramienta utilizada*

Para la realización del modelo entidad relación se ha utilizado la herramienta “Draw.io”, conocida actualmente como [“diagrams.net”](https://diagrams.net).

Drawio.io es una herramienta online de código abierto y gratuita que permite crear diagramas. Además, se puede usar de forma integrada con plataformas como GoogleDrive...

Está adaptada para variedad de temas y categorías, desde diagramas de flujo y UML hasta modelos EER, elementos Cisco, Citrix...

Entre sus utilidades más frecuentes se encuentran:

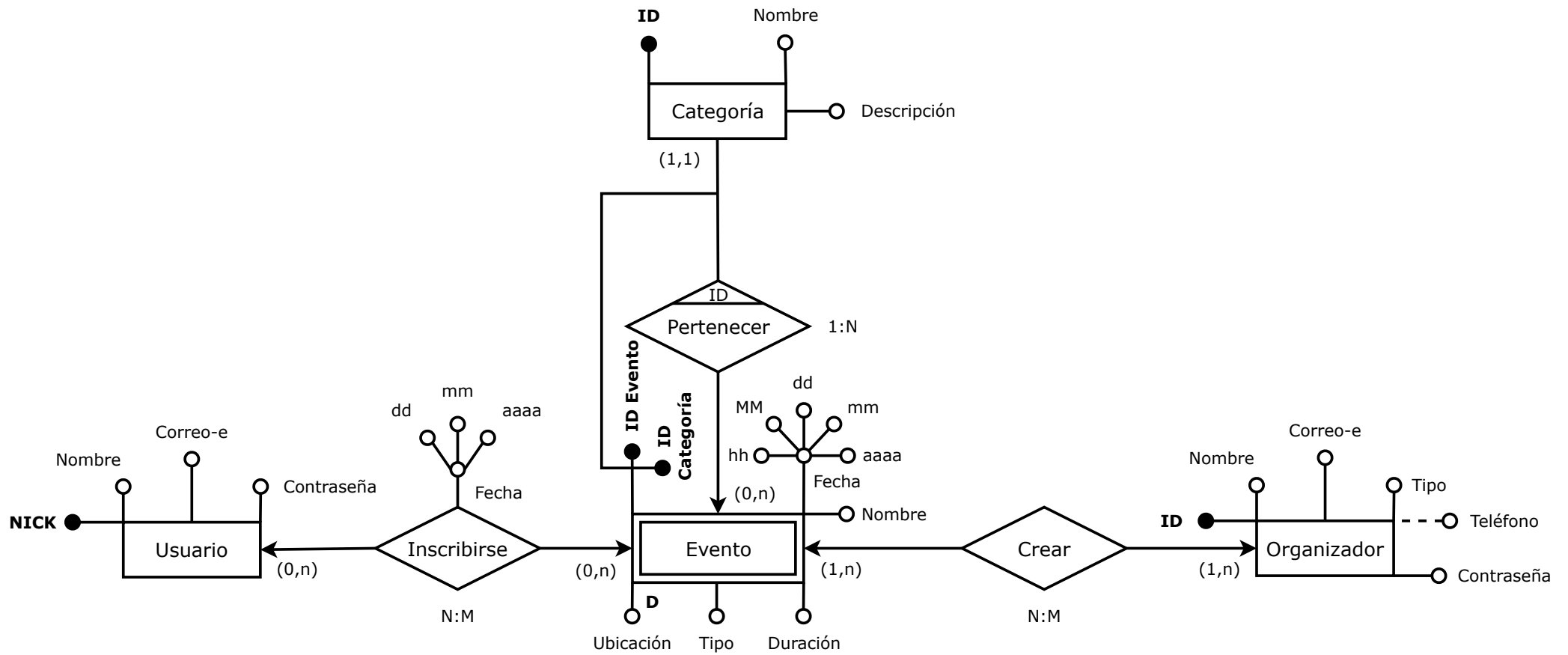
- Diseñar diagramas de flujo
- Mapas conceptuales / mentales
- Organigramas
- Esquemas de red
- Planos de procesos



En resumen, podemos concluir que es una herramienta muy utilizada para visualizar ideas, procesos o estructuras de forma clara y profesional.

Antia Loureiro Ferrio - Proyecto Indra

ii. Esquema



iii. Diccionario de datos

Atributos:

Se ha añadido el **nick** al **usuario** para que tenga una clave primaria única que lo identifique.

Lo mismo ocurre con el **organizador**, el **evento** y la **categoría**, a los cuales se les ha añadido un **ID**.

A la **categoría** se le han añadido, además, el **nombre** y una **descripción**. En principio existen 3 categorías: taller, conferencia y actividad sostenible.

Al **organizador** se le añadido una **contraseña** para facilitar el acceso a la plataforma y poder gestionar los eventos actuales y futuros.

Se considera que los datos de contacto del **organizador** son el **correo electrónico** y su **teléfono**, este último es opcional porque puede proporcionarlo o no.

El atributo **tipo** de **organizador** puede tomar tres valores: empresa, institución o persona.

El atributo **tipo** de **evento** puede tomar dos valores: online o presencial.

La **fecha** del **evento** es un **atributo compuesto** formado por 5 campos: hora, minutos, día, mes y año.

La **ubicación** del **evento** es un **atributo derivado**, que solo tomará valor en caso de que el tipo de evento tenga como valor “presencial”.

Se ha añadido el atributo compuesto **fecha** en la interrelación **inscribirse**, ya que, será necesario registrar la fecha de inscripción del usuario en el evento. Por ejemplo, si hay más inscripciones que plazas, habrá que comprobar qué usuarios se han inscrito con anterioridad.

Este atributo está formado por 3 campos: día, mes y año.

Debilidades:

Se considera que un **evento** no puede existir si no pertenece a una **categoría**, por lo que evento es **débil por existencia**, además, también es débil por **identificación**, ya que puede haber varios eventos iguales que pertenezcan a categorías diferentes.

Cardinalidades:

Las cardinalidades se han realizado de forma subjetiva y lo más parecida a la realidad posible teniendo en cuenta la operabilidad de la página web.

- ✓ **Pertenecer:** un evento solo puede pertenecer a una categoría (1,1), y una categoría puede no tener eventos o tener varios (0,n).

- ✓ **Crear:** un evento es creado por, como mínimo 1 organizador o por varios (1,n). Un organizador tiene que crear como mínimo 1 evento para considerarse organizador, sin tener un máximo establecido (1,n).

- ✓ **Inscribirse:** un usuario puede no inscribirse en ningún evento o en varios (0,n) y un evento puede tener a ningún o a varios usuarios inscritos (0,n).

Entidad: Categoría									
Atributo	Descripción	Dominio	Tipo/Longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
ID	Identificador único de la categoría	Cualquier cadena de caracteres que identifique a la categoría de forma única.	Alf / 50			N		S	
Nombre	Nombre de la categoría	Enumerado para el nombre de la categoría, solo hay tres posibilidades: taller, conferencia o actividad ecológica.	Enum (taller, conferencia, actividad)			N		S	
Descripción	Descripción de la categoría	Cadena de caracteres que describe la categoría en cuestión (funcionamiento, finalidad...).	Alf / 255			N		S	
Observaciones y restricciones de la entidad: Solo habrá tres categorías.									



8



10

ii. De Interrelación

Interrelación: Inscribirse									
Atributo	Descripción	Dominio	Tipo/Longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
ID Categoría	Identificador único de la categoría	Cualquier cadena de caracteres que identifique a la categoría de forma única.	Alf / 50			N		S	
ID Evento	Identificador único del evento	Cualquier cadena de caracteres que identifique al evento de forma única.	Alf / 50			N		S	
Nick Usuario	Identificador único del usuario	Cualquier cadena de caracteres que identifique al usuario de forma única.	Alf / 50			N		S	
Fecha	Fecha en la que un usuario se inscribe en un evento	Atributo compuesto que se conforma de 10 caracteres con la siguiente estructura “dd/mm/aaaa”.	Alf / 10			N		S	
Observaciones y restricciones de la entidad: La tabla inscribirse tiene una clave primaria formada por tres claves foráneas, una proveniente del usuario (Nick Usuario), y las otras dos provenientes del evento (ID Categoría y ID Evento). Además, la fecha se utilizará como campo compuesto en lugar de como sus componentes para que sea más simple de manejar.									

12

2. Creación de la base de datos:

a. Scripts de creación de la base de datos

Creación de la Base de Datos:

“

```
CREATE SCHEMA `antia_loureiro_ferrio_eventos`;
```

”

Creación de las tablas:

Categoría:

“

```
USE `antia_loureiro_ferrio_eventos`;
```

```
CREATE TABLE `categoria` (  
  `ID` VARCHAR(50) NOT NULL,  
  `Nombre` ENUM('taller', 'conferencia', 'actividad') NOT NULL,  
  `Descripcion` VARCHAR(255) NOT NULL,  
  PRIMARY KEY (`ID`)  
);  
”
```

Evento:

“

USE `antia_loureiro_ferrio_eventos`;

CREATE TABLE `evento` (

 `ID_Categoria` VARCHAR(50) NOT NULL,

 `ID_Evento` VARCHAR(50) NOT NULL,

 `Nombre` VARCHAR(100) NOT NULL,

 `Fecha` VARCHAR(16) NOT NULL CHECK (`Fecha` REGEXP '^[0-1][0-9]|2[0-3]):([0-5][0-9])-(0[1-9]|[1-2][0-9]|3[0-1])/([0-1][0-9]|1[0-2])/([0-9]{4})\$'),

 `Duracion` INT NOT NULL,

 `Tipo` ENUM('online', 'presencial') NOT NULL,

 `Ubicacion` VARCHAR(255) NOT NULL ,

 PRIMARY KEY (`ID_Categoria`, `ID_Evento`),

 CONSTRAINT `ID_Categoria`

 FOREIGN KEY (`ID_Categoria`)

 REFERENCES `antia_loureiro_ferrio_eventos`.`categoria` (`ID`)

 ON DELETE NO ACTION

 ON UPDATE NO ACTION

);

”

Para hacer que la ubicación sea un atributo derivado, se ha establecido como obligatorio y, posteriormente, se realiza un trigger para conseguir que, en caso de que el evento sea de tipo “presencial”, esta ubicación sí pueda tomar valor y, en caso de que el tipo de evento sea “online”, la ubicación será null.

“

DELIMITER \$\$

DROP TRIGGER IF EXISTS ubicacion_evento\$\$

CREATE TRIGGER ubicacion_evento

BEFORE INSERT ON evento

FOR EACH ROW

BEGIN

IF NEW.tipo NOT LIKE 'presencial' THEN

SET NEW.ubicacion = null;

END IF;

END;

\$\$

DELIMITER ;

”

Para que el evento sea débil de categoría, antes de que se inserte un nuevo evento, hay que comprobar que el id de la categoría a la que pertenece ya existe en la base de datos.

“

DELIMITER \$\$

DROP TRIGGER IF EXISTS evento_debil_categoria\$\$

CREATE TRIGGER evento_debil_categoria

BEFORE INSERT ON evento

FOR EACH ROW

BEGIN

 DECLARE categoriaExiste INT;

 SELECT COUNT(*) INTO categoriaExiste

 FROM categoria

 WHERE ID_Categoria = NEW.ID_Categoria;

 IF categoriaExiste = 0 THEN

 SIGNAL SQLSTATE '45000'

 SET MESSAGE_TEXT = 'No se puede insertar el evento porque la categoría no existe.';

 END IF;

END;

\$\$

DELIMITER ;

”

Usuario:

“

USE `antia_loureiro_ferrio_eventos`;

```
CREATE TABLE `usuario` (  
  `Nick` VARCHAR(50) NOT NULL,  
  `Nombre` VARCHAR(50) NOT NULL,  
  `Correo-e` VARCHAR(255) NOT NULL CHECK (`Correo-e` REGEXP  
'^[^@]+@[^@]+\.[^@]+$'),  
  `Contraseña` VARCHAR(20) NOT NULL CHECK (  
    LENGTH(`Contraseña`) BETWEEN 8 AND 20 AND  
    `Contraseña` REGEXP '.*[A-Z].*[A-Z]' AND  
    `Contraseña` REGEXP '.*[a-z].*[a-z]' AND  
    `Contraseña` REGEXP '.*[0-9].*[0-9]' AND  
    `Contraseña` REGEXP '.*[^a-zA-Z0-9].*[^a-zA-Z0-9]'  
  ),  
  PRIMARY KEY (`Nick`)  
);  
”
```

Organizador:

“

USE `antia_loureiro_ferrio_eventos`;

```
CREATE TABLE `organizador` (  
  `ID` VARCHAR(50) NOT NULL,  
  `Nombre` VARCHAR(50) NOT NULL,  
  `Correo-e` VARCHAR(255) NOT NULL CHECK (`Correo-e` REGEXP  
'^[^@]+@[^@]+\.[^@]+$'),  
  `Contraseña` VARCHAR(20) NOT NULL CHECK (  
    LENGTH(`Contraseña`) BETWEEN 8 AND 20 AND  
    `Contraseña` REGEXP '.*[A-Z].*[A-Z]' AND  
    `Contraseña` REGEXP '.*[a-z].*[a-z]' AND  
    `Contraseña` REGEXP '.*[0-9].*[0-9]' AND  
    `Contraseña` REGEXP '.*[^a-zA-Z0-9].*[^a-zA-Z0-9]'  
  ),  
  `Tipo` ENUM('institucion', 'empresa', 'persona') NOT NULL,  
  `Telefono` CHAR(9) NULL,  
  PRIMARY KEY (`ID`)  
);  
”
```

Inscribirse:

“

```
USE `antia_loureiro_ferrio_eventos`;
```

```
CREATE TABLE `inscribirse` (  
  `ID_Categoria` VARCHAR(50) NOT NULL,  
  `ID_Evento` VARCHAR(50) NOT NULL,  
  `Nick_Usuario` VARCHAR(50) NOT NULL,  
  `Fecha` VARCHAR(10) NOT NULL CHECK (`Fecha` REGEXP '^0[1-9]||[1-2][0-9]|3[0-1]/0[1-9]|1[0-2]/[0-9]{4}$'),  
  PRIMARY KEY (`ID_Categoria`, `ID_Evento`, `Nick_Usuario`),  
  INDEX `Nick_Usuario_idx` (`Nick_Usuario` ASC) VISIBLE,  
  CONSTRAINT `ID_Evento_Categoria_Inscribirse`  
    FOREIGN KEY (`ID_Categoria`, `ID_Evento`)  
    REFERENCES `antia_loureiro_ferrio_eventos`.`evento`  
    (`ID_Categoria`, `ID_Evento`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `Nick_Usuario_Inscribirse`  
    FOREIGN KEY (`Nick_Usuario`)  
    REFERENCES `antia_loureiro_ferrio_eventos`.`usuario` (`Nick`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION  
);  
”
```

Crear:

“

USE `antia_loureiro_ferrio_eventos`;

CREATE TABLE `crear` (

 `ID_Categoria` VARCHAR(50) NOT NULL,

 `ID_Evento` VARCHAR(50) NOT NULL,

 `ID_Organizador` VARCHAR(50) NOT NULL,

 PRIMARY KEY (`ID_Categoria`, `ID_Evento`, `ID_Organizador`),

 INDEX `ID_Organizador_idx` (`ID_Organizador` ASC) VISIBLE,

 CONSTRAINT `ID_Evento_Categoria_Crear`

 FOREIGN KEY (`ID_Categoria`, `ID_Evento`)

 REFERENCES `antia_loureiro_ferrio_eventos`.`evento`

 (`ID_Categoria`, `ID_Evento`)

 ON DELETE NO ACTION

 ON UPDATE NO ACTION,

 CONSTRAINT `ID_Organizador_Crear`

 FOREIGN KEY (`ID_Organizador`)

 REFERENCES `antia_loureiro_ferrio_eventos`.`organizador` (`ID`)

 ON DELETE NO ACTION

 ON UPDATE NO ACTION

);

”

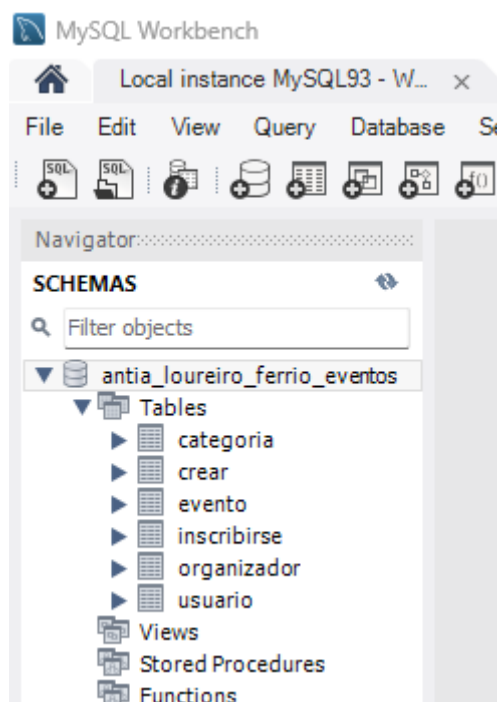
b. Visualización de la estructura de la base de datos

Se visualizará la estructura de la base de datos de forma gráfica mediante la herramienta “MySQL Workbench”.

La base de datos se ha creado de forma local en el equipo, por lo que también se podrá visualizar desde “MySQL Workbench” además de mediante cmd, como se mostrará posteriormente.



Comprobación de que **la base de datos** se ha creado correctamente:



Comprobación de la tabla categoría:

En el apartado “[Columns](#)” se pueden observar las columnas por las que está formada una tabla.

En el apartado “[Foreign Keys](#)” se verán las claves foráneas que posee la tabla y a qué atributo de qué tabla hacen referencia.

Ambos apartados se encuentran en la parte inferior izquierda de la pantalla que observamos a continuación.

The screenshot shows the 'Table Structure' window for a table named 'categoria' in the 'antia_loureiro_ferrio_eventos' schema. The table is using the 'InnoDB' engine and 'utf8mb4' character set. The columns are defined as follows:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
ID	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Nombre	ENUM('taller', 'conferencia', 'actividad')	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Descripción	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

The 'Columns' and 'Foreign Keys' tabs are highlighted with red boxes. The 'Columns' tab is currently selected, showing the column definitions. The 'Foreign Keys' tab is also visible, showing no foreign keys are currently defined for this table.

Comprobación de la tabla evento:

evento - Table

Table Name:

evento

Schema:

antia_loureiro_ferrio_eventos

Charset/Collation:








utf8mb4

utf8mb4_0900_ai_ci

Engine:

InnoDB

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
 ID_Categoria	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 ID_Evento	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 Nombre	VARCHAR(100)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 Fecha	VARCHAR(16)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 Duracion	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 Tipo	ENUM('online', 'presencial')	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 Ubicacion	VARCHAR(255)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name:

CharSet/Collation:

Default CharSet

Default Collation

Comments:

Data Type:

Default:

Storage:

☐ Virtual
 ☐ Stored

☐ Primary Key
 ☐ Not Null
 ☐ Unique

☐ Binary
 ☐ Unsigned
 ☐ Zero Fill

☐ Auto Increment
 ☐ Generated

Columns

Indexes

Foreign Keys

Triggers

Partitioning

Options

Apply

Revert

evento - Table




Table Name:

Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation:

Engine:

Comments:

Foreign Key Name	Referenced Table	Column	Referenced Column
ID_Categoria	'antia_loureiro_ferrio_eventos'. 'categoria'	<input checked="" type="checkbox"/> ID_Categoria <input type="checkbox"/> ID_Evento <input type="checkbox"/> Nombre <input type="checkbox"/> Fecha <input type="checkbox"/> Duracion <input type="checkbox"/> Tipo <input type="checkbox"/> Ubicacion	ID

Foreign Key Options

On Update:
On Delete:
☐ Skip in SQL generation

Foreign Key Comment

Columns

Indexes

Foreign Keys

Triggers

Partitioning

Options

Apply

Revert

Comprobación de la tabla usuario:

usuario - Table

Table Name: Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation: Engine:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Nick	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Nombre	VARCHAR(50)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Correo-e	VARCHAR(255)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Contraseña	VARCHAR(20)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name: Data Type:

Charset/Collation: Default:

Comments:

Storage: ☐ Virtual ☐ Stored ☐ Primary Key ☐ Not Null ☐ Unique ☐ Binary ☐ Unsigned ☐ Zero Fill ☐ Auto Increment ☐ Generated

Columns Indexes Foreign Keys Triggers Partitioning Options

Apply Revert

Comprobación de la tabla organizador:

organizador - Table

Table Name: Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation: Engine:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
ID	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Nombre	VARCHAR(50)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Correo-e	VARCHAR(255)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Contraseña	VARCHAR(20)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Tipo	ENUM('Institucion', 'empresa', 'persona')	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Telefono	CHAR(9)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Column Name: Data Type:

Charset/Collation: Default:

Comments:


Storage: ☐ Virtual ☐ Stored ☐ Primary Key ☐ Not Null ☐ Unique ☐ Binary ☐ Unsigned ☐ Zero Fill ☐ Auto Increment ☐ Generated

Columns Indexes Foreign Keys Triggers Partitioning Options

Apply Revert

Comprobación de la tabla inscribirse:

inscribirse - Table


Table Name: Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation: Engine:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
ID_Categoria	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ID_Evento	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Nick_Usuario	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Fecha	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name:
Data Type:

Charset/Collation:
Default:

Comments:


Storage: ☐ Virtual ☐ Stored
☐ Primary Key ☐ Not Null ☐ Unique
☐ Binary ☐ Unsigned ☐ Zero Fill
☐ Auto Increment ☐ Generated

Columns Indexes Foreign Keys Triggers Partitioning Options

Apply Revert

Acceder al apartado de “Foreign Keys” como se especificó anteriormente. Para ver la referencia de cada clave, hay que seleccionarla:

inscribirse - Table



Table Name: Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation: Engine:

Comments:

Foreign Key Name	Referenced Table	Column	Referenced Column
ID_Evento_Categoria_In...	`antia_loureiro_ferrio_eventos`.`evento`	<input checked="" type="checkbox"/> ID_Categoria	ID_Categoria
Nick_Usuario_Inscribirse	`antia_loureiro_ferrio_eventos`.`usuario`	<input checked="" type="checkbox"/> ID_Evento	ID_Evento
		<input type="checkbox"/> Nick_Usuario	
		<input type="checkbox"/> Fecha	

inscribirse - Table


Table Name: Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation: Engine:

Comments:

Foreign Key Name	Referenced Table	Column	Referenced Column
ID_Evento_Categoria_In...	`antia_loureiro_ferrio_eventos`.`evento`	<input type="checkbox"/> ID_Categoria	
Nick_Usuario_Inscribirse	`antia_loureiro_ferrio_eventos`.`usuario`	<input type="checkbox"/> ID_Evento	
		<input checked="" type="checkbox"/> Nick_Usuario	Nick
		<input type="checkbox"/> Fecha	

Comprobación de la tabla crear:

crear - Table

Table Name: Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation: Engine:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
ID_Categoria	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ID_Evento	VARCHAR(50)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ID_Organizador	VARCHAR(50)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name: Data Type:

Charset/Collation: Default:

Comments:

Storage: ☐ Virtual ☐ Stored

☐ Primary Key ☐ Not Null ☐ Unique

☐ Binary ☐ Unsigned ☐ Zero Fill

☐ Auto Increment ☐ Generated

Columns Indexes Foreign Keys Triggers Partitioning Options

Apply Revert

Al igual que con la tabla inscribirse, se comprobarán las claves foráneas de la siguiente forma:

crear - Table

Table Name: Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation: Engine:

Comments:

Foreign Key Name	Referenced Table
ID_Evento_Categoria_Crear	`antia_loureiro_ferrio_eventos`.`evento`
ID_Organizador_Crear	`antia_loureiro_ferrio_eventos`.`organizador`

Column	Referenced Column
<input checked="" type="checkbox"/> ID_Categoria	ID_Categoria
<input checked="" type="checkbox"/> ID_Evento	ID_Evento
<input type="checkbox"/> ID_Organizador	

crear - Table

Table Name: Schema: **antia_loureiro_ferrio_eventos**

Charset/Collation: Engine:

Comments:

Foreign Key Name	Referenced Table
ID_Evento_Categoria_Crear	`antia_loureiro_ferrio_eventos`.`evento`
ID_Organizador_Crear	`antia_loureiro_ferrio_eventos`.`organizador`

Column	Referenced Column
<input type="checkbox"/> ID_Categoria	
<input type="checkbox"/> ID_Evento	
<input checked="" type="checkbox"/> ID_Organizador	ID

Usando CMD:

Comprobación de que la base de datos se ha creado correctamente:

```
Símbolo del sistema - mysql
C:\Program Files\MySQL\MySQL Server 9.3\bin>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 13
Server version: 9.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

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> SHOW DATABASES;
+-----+
| Database |
+-----+
| antia_loureiro_ferrio_eventos |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
6 rows in set (0.013 sec)

mysql>
```

Comprobación de la tabla categoría:

Una vez seleccionada la base de datos, ya se podrán mostrar todas las tablas, es decir, solo se realizará “USE antia_loureiro_ferrio_eventos;” una vez.

```
Símbolo del sistema - mysql
mysql> USE antia_loureiro_ferrio_eventos;
Database changed
mysql>
mysql> DESCRIBE categoria;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ID | varchar(50) | NO | PRI | NULL | |
| Nombre | enum('taller','conferencia','actividad') | NO | | NULL | |
| Descripcion | varchar(255) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.010 sec)

mysql>
```

Comprobación de la tabla evento:

```
mysql> DESCRIBE evento;
```

Field	Type	Null	Key	Default	Extra
ID_Categoria	varchar(50)	NO	PRI	NULL	
ID_Evento	varchar(50)	NO	PRI	NULL	
Nombre	varchar(100)	NO		NULL	
Fecha	varchar(16)	NO		NULL	
Duracion	int	NO		NULL	
Tipo	enum('online','presencial')	NO		NULL	
Ubicacion	varchar(255)	NO		NULL	

```
7 rows in set (0.011 sec)

mysql>
```

Comprobación de la tabla usuario:

```
mysql> DESCRIBE usuario;
```

Field	Type	Null	Key	Default	Extra
Nick	varchar(50)	NO	PRI	NULL	
Nombre	varchar(50)	NO		NULL	
Correo-e	varchar(255)	NO		NULL	
Contraseña	varchar(20)	NO		NULL	

```
4 rows in set (0.008 sec)
```

Comprobación de la tabla organizador:

```
mysql> DESCRIBE organizador;
```

Field	Type	Null	Key	Default	Extra
ID	varchar(50)	NO	PRI	NULL	
Nombre	varchar(50)	NO		NULL	
Correo-e	varchar(255)	NO		NULL	
Contraseña	varchar(20)	NO		NULL	
Tipo	enum('institucion','empresa','persona')	NO		NULL	
Telefono	char(9)	YES		NULL	

```
6 rows in set (0.010 sec)

mysql>
```

Comprobación de la tabla inscribirse:

```
mysql> DESCRIBE inscribirse;
```

Field	Type	Null	Key	Default	Extra
ID_Categoria	varchar(50)	NO	PRI	NULL	
ID_Evento	varchar(50)	NO	PRI	NULL	
Nick_Usuario	varchar(50)	NO	PRI	NULL	
Fecha	varchar(10)	NO		NULL	

```
4 rows in set (0.011 sec)

mysql>
```

Comprobación de la tabla crear:

```
Símbolo del sistema - mysql × + v
mysql> DESCRIBE crear;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ID_Categoria   | varchar(50)   | NO   | PRI | NULL    |       |
| ID_Evento      | varchar(50)   | NO   | PRI | NULL    |       |
| ID_Organizador | varchar(50)   | NO   | PRI | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.011 sec)

mysql>
```

Comprobación de la base de datos tiene **todas las tablas creadas anteriormente**:

```
Símbolo del sistema - mysql × + v
mysql> SHOW TABLES;
+-----+
| Tables_in_antia_loureiro_ferrio_eventos |
+-----+
| categoria                                |
| crear                                    |
| evento                                   |
| inscribirse                             |
| organizador                             |
| usuario                                  |
+-----+
6 rows in set (0.015 sec)

mysql>
```


Trigger Ubicación:

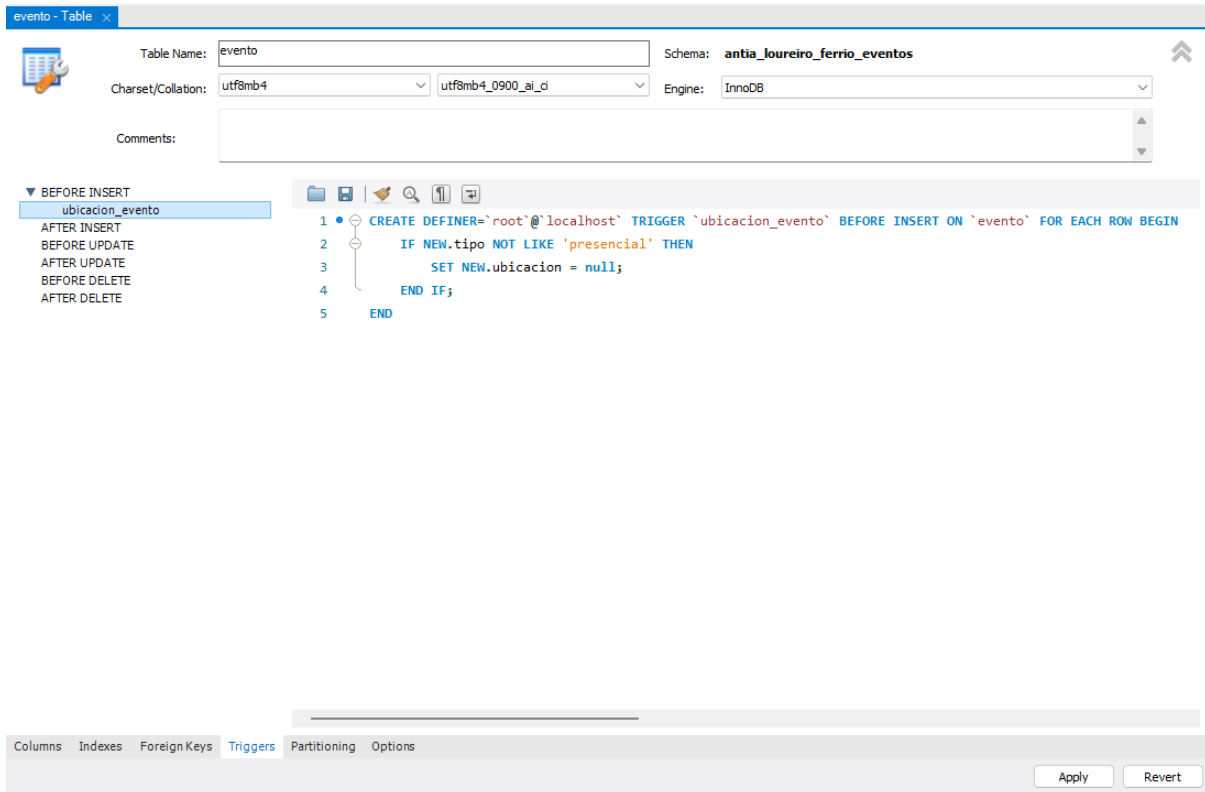
Se ha insertado el trigger en la base de datos, el cual comprueba el correcto valor de la ubicación:

```
Símbolo del sistema - mysql × + v
mysql> USE antia_loureiro_ferrio_eventos;
Database changed
mysql> DELIMITER $$
mysql> DROP TRIGGER IF EXISTS ubicacion_evento$$
Query OK, 0 rows affected (0.050 sec)

mysql> CREATE TRIGGER ubicacion_evento
-> BEFORE INSERT ON evento
-> FOR EACH ROW
-> BEGIN
->     IF NEW.tipo NOT LIKE 'presencial' THEN
->         SET NEW.ubicacion = null;
->     END IF;
-> END;
-> $$
Query OK, 0 rows affected (0.151 sec)

mysql> DELIMITER ;
mysql>
```

Para comprobar que el trigger se ha insertado correctamente, podemos acceder a la sección “Triggers” dentro de la tabla eventos:



The screenshot shows the MySQL Workbench interface for the 'evento' table. The 'Triggers' tab is selected, displaying the trigger 'ubicacion_evento'. The trigger is defined as a BEFORE INSERT trigger on the 'evento' table. The trigger logic is as follows:

```
1 CREATE DEFINER='root'@'localhost' TRIGGER `ubicacion_evento` BEFORE INSERT ON `evento` FOR EACH ROW BEGIN
2   IF NEW.tipo NOT LIKE 'presencial' THEN
3     SET NEW.ubicacion = null;
4   END IF;
5 END
```

The interface also shows the table name 'evento', schema 'antia_loureiro_ferrio_eventos', charset 'utf8mb4', collation 'utf8mb4_0900_ai_ci', and engine 'InnoDB'. The 'Triggers' tab is highlighted in the bottom navigation bar, and the 'Apply' and 'Revert' buttons are visible at the bottom right.

Trigger Debilidad Evento:

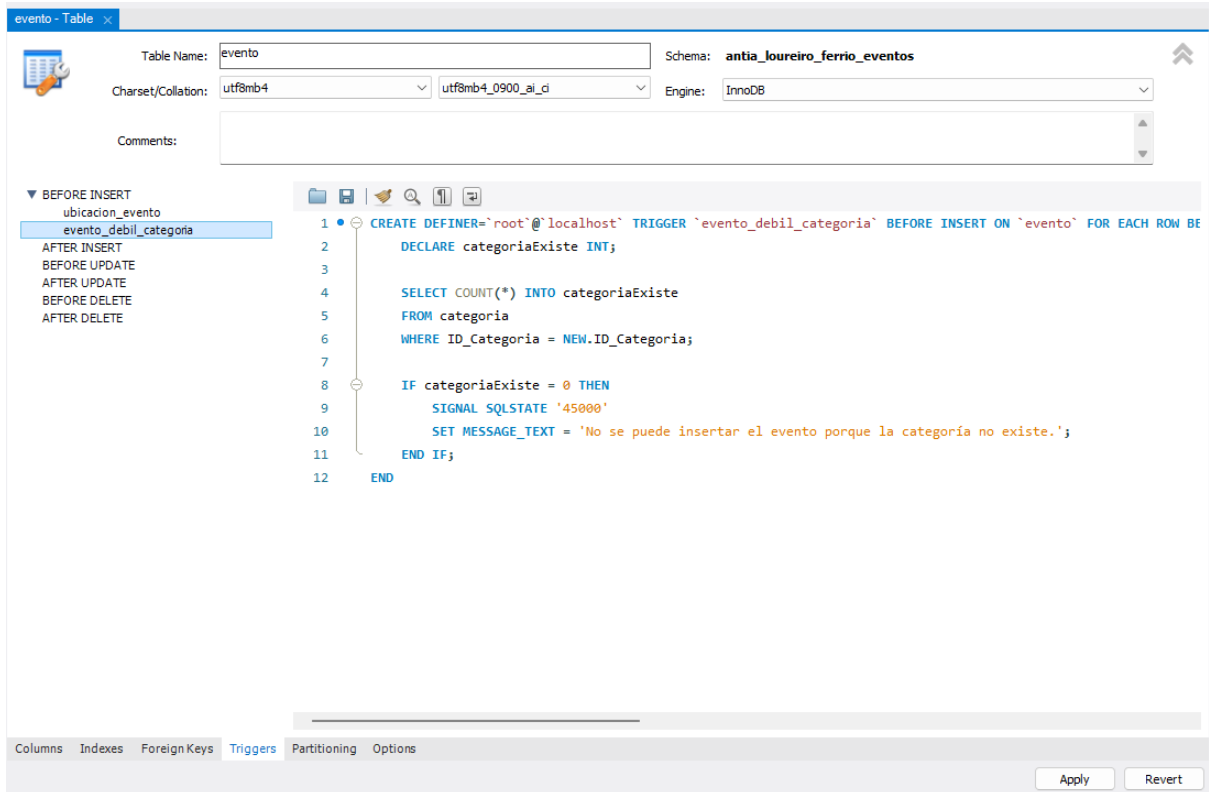
Se ha insertado el trigger en la base de datos, este se asegura de que, si la categoría a la que hace referencia el evento no existe, este evento no se podrá insertar:

```
Símbolo del sistema - mysql · X + v
mysql> DELIMITER $$
mysql>
mysql> DROP TRIGGER IF EXISTS evento_debil_categoria$$
Query OK, 0 rows affected, 1 warning (0.026 sec)

mysql> CREATE TRIGGER evento_debil_categoria
-> BEFORE INSERT ON evento
-> FOR EACH ROW
-> BEGIN
->     DECLARE categoriaExiste INT;
->
->     SELECT COUNT(*) INTO categoriaExiste
->     FROM categoria
->     WHERE ID_Categoria = NEW.ID_Categoria;
->
->     IF categoriaExiste = 0 THEN
->         SIGNAL SQLSTATE '45000'
->         SET MESSAGE_TEXT = 'No se puede insertar el evento porque la categoría no existe.';
->     END IF;
-> END;
-> $$
Query OK, 0 rows affected (0.073 sec)

mysql>
mysql> DELIMITER ;
mysql>
```

Para comprobar que el trigger se ha insertado correctamente, podemos acceder a la sección “Triggers” dentro de la tabla eventos:



The screenshot shows the MySQL Workbench interface for the 'evento' table in the 'antia_loureiro_ferrio_eventos' schema. The 'Triggers' tab is active, showing a trigger named 'evento_debil_categoria' defined as a BEFORE INSERT trigger for each row.

Table Name: evento
Schema: antia_loureiro_ferrio_eventos
Charset/Collation: utf8mb4 utf8mb4_0900_ai_ci
Engine: InnoDB

Comments:

Triggers:

- BEFORE INSERT
- ubicacion_evento
- evento_debil_categoria
- AFTER INSERT
- BEFORE UPDATE
- AFTER UPDATE
- BEFORE DELETE
- AFTER DELETE

```
1 CREATE DEFINER='root'@'localhost' TRIGGER `evento_debil_categoria` BEFORE INSERT ON `evento` FOR EACH ROW BE
2 DECLARE categoriaExiste INT;
3
4 SELECT COUNT(*) INTO categoriaExiste
5 FROM categoria
6 WHERE ID_Categoria = NEW.ID_Categoria;
7
8 IF categoriaExiste = 0 THEN
9     SIGNAL SQLSTATE '45000'
10     SET MESSAGE_TEXT = 'No se puede insertar el evento porque la categoría no existe.';
11 END IF;
12 END
```

Columns Indexes Foreign Keys Triggers Partitioning Options

Apply Revert