



## 1. Introducción

### 1.1 Descripción General del Proyecto

Corchea Academy es una plataforma de educación musical en línea que busca democratizar el acceso al aprendizaje musical de alta calidad. El proyecto se centra en el desarrollo de una base de datos robusta y escalable que permita la gestión eficiente de usuarios, cursos, instructores, inscripciones, pagos y demás información relevante para el funcionamiento de la academia.

### 1.2 Alcance del Documento

Este documento describe el proceso de diseño e implementación de la base de datos para Corchea Academy. Incluye la definición del problema, el análisis de los requerimientos, el diseño del modelo de datos, la implementación en MySQL, la creación de funciones, vistas, procedimientos almacenados y triggers, así como una descripción del modelo de negocio que sustenta la plataforma.

## 2. Objetivo

Diseñar e implementar una base de datos relacional en MySQL para gestionar la información de la plataforma de cursos de música en línea Corchea Academy, asegurando la integridad, eficiencia y escalabilidad del sistema.

## 3. Situación Problemática

Corchea Academy se enfrenta al desafío de gestionar una creciente cantidad de información relacionada con sus cursos, usuarios, instructores y transacciones financieras. La falta de un sistema centralizado y eficiente para administrar estos datos dificulta el crecimiento de la plataforma y limita la capacidad de ofrecer una experiencia de usuario óptima.

Actualmente, la información se encuentra dispersa en diferentes archivos y hojas de cálculo, lo que genera inconsistencias, duplicidad de datos y dificulta la generación de reportes y el análisis del rendimiento del negocio. Además, la gestión manual de las inscripciones, pagos y seguimiento del progreso de los estudiantes consume una gran cantidad de tiempo y recursos.

Por lo tanto, se requiere una solución tecnológica que permita centralizar, organizar y gestionar la información de Corchea Academy de manera eficiente, segura y escalable. Esta solución debe facilitar el registro de usuarios, la administración de cursos, el proceso de inscripción y pago, el seguimiento del progreso de los estudiantes, la comunicación con los instructores y la generación de reportes para la toma de decisiones estratégicas.

## 4. Modelo de Negocio

### 4.1 Propuesta de Valor

Corchea Academy ofrece una plataforma de aprendizaje en línea que permite a los usuarios acceder a cursos de música de alta calidad, impartidos por instructores experimentados, a un precio accesible y con la flexibilidad de aprender a su propio ritmo.

### 4.2 Segmentos de Clientes

- **Estudiantes de música:** Personas de todas las edades y niveles que desean aprender a tocar un instrumento, mejorar su técnica o adquirir conocimientos musicales.
- **Músicos aficionados:** Personas que tocan música como hobby y buscan mejorar sus habilidades.
- **Profesores de música:** Instructores que buscan una plataforma para impartir sus cursos en línea y alcanzar a un público más amplio.

### 4.3 Canales de Distribución

- **Sitio web propio:** La plataforma principal para acceder a los cursos, registrarse, realizar pagos y gestionar el aprendizaje.
- **Redes sociales:** Para la promoción de cursos, creación de comunidad y captación de nuevos estudiantes.
- **Publicidad en línea:** Anuncios en buscadores y plataformas de video para alcanzar a un público más amplio.

## 5. Descripciones de Tablas y campos

Tabla	Descripción	Campo	Abreviatura	Tipo de Dato	Tipo de Clave
Usuarios	Almacena la información de los usuarios registrados en la plataforma.	id_usuario	ID Usuario	INT	Primaria
		nombre	Nombre	VARCHAR(100)	
		apellido	Apellido	VARCHAR(100)	
		email	Email	VARCHAR(100)	Único
		contraseña	Contraseña	VARCHAR(255)	
		fecha_registro	Fecha Reg.	DATETIME	
		id_rol	ID Rol	INT	Foránea
Roles	Almacena los roles de usuario (Administrador, Estudiante, etc.).	id_rol	ID Rol	INT	Primaria
		nombre_rol	Nombre Rol	VARCHAR(50)	
Cursos	Almacena la información de los cursos ofrecidos en la plataforma.	id_curso	ID Curso	INT	Primaria
		nombre_curso	Nombre Curso	VARCHAR(100)	
		descripcion	Descripción	TEXT	
		precio	Precio	DECIMAL(10, 2)	
		fecha_creacion	Fecha Creación	DATETIME	
		id_categoria	ID Cat.	INT	Foránea

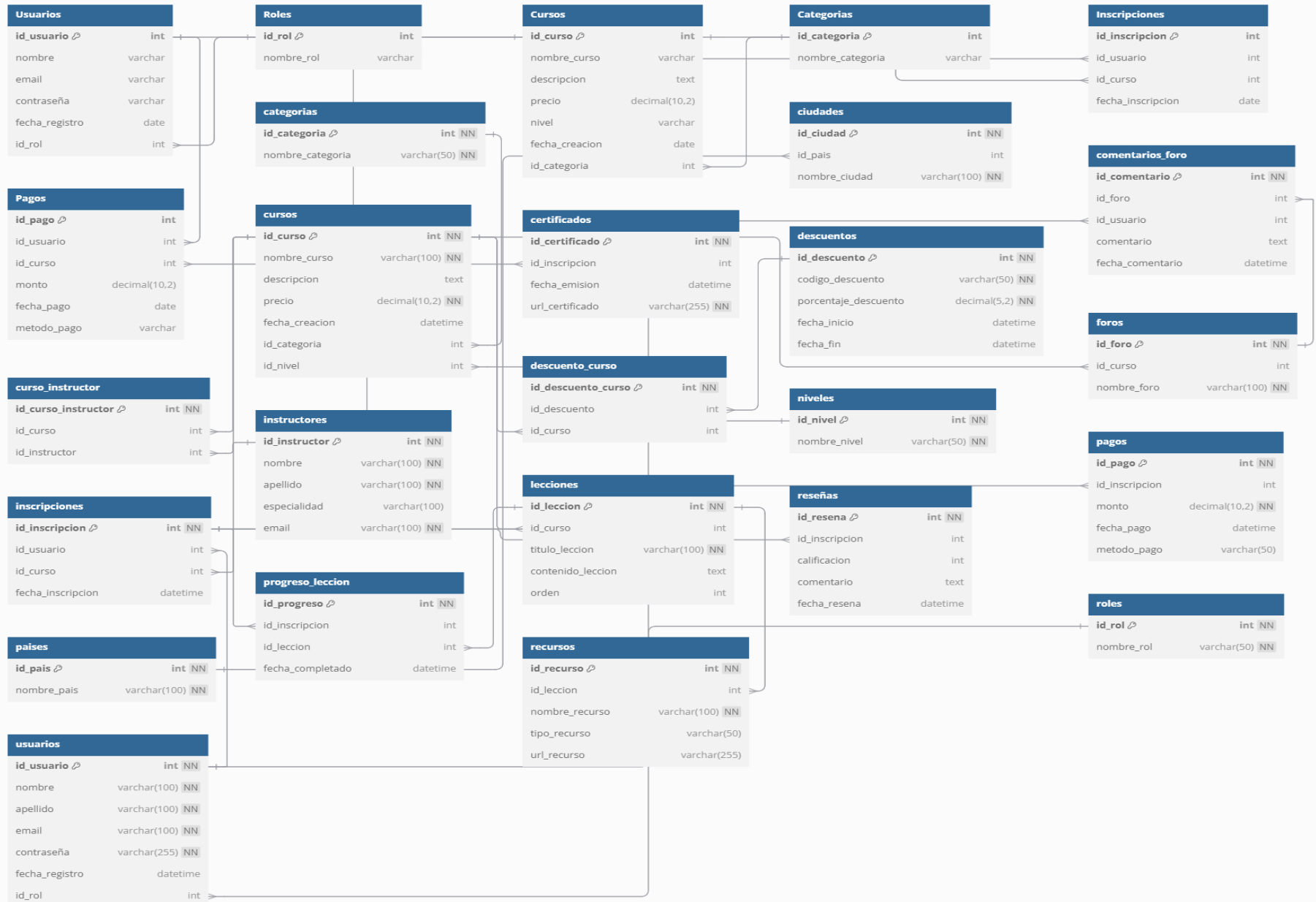
		id_nivel	ID Nivel	INT	Foránea
Categorías	Almacena las categorías de los cursos (ej. Bajo, Batería, Piano).	id_categoria	ID Cat.	INT	Primaria
		nombre_categoria	Nombre Cat.	VARCHAR(50)	
Niveles	Almacena los niveles de dificultad de los cursos (ej. Principiante, Intermedio, Avanzado).	id_nivel	ID Nivel	INT	Primaria
		nombre_nivel	Nombre Nivel	VARCHAR(50)	
Inscripciones	Almacena las inscripciones de los usuarios a los cursos.	id_inscripcion	ID Inscr.	INT	Primaria
		id_usuario	ID Usuario	INT	Foránea
		id_curso	ID Curso	INT	Foránea
		fecha_inscripcion	Fecha Inscr.	DATETIME	
Pagos	Almacena los registros de los pagos realizados por los usuarios.	id_pago	ID Pago	INT	Primaria
		id_inscripcion	ID Inscripción	INT	Foránea
		monto	Monto	DECIMAL(10, 2)	
		fecha_pago	Fecha Pago	DATETIME	
		metodo_pago	Método Pago	VARCHAR(50)	
Reseñas	Almacena las calificaciones y comentarios de los usuarios sobre los cursos (Tabla de Hechos).	id_resena	ID Reseña	INT	Primaria
		id_inscripcion	ID Inscripción	INT	Foránea
		calificacion	Calificación	INT	
		comentario	Comentario	TEXT	

		fecha_resena	Fecha Reseña	DATETIME	
Instructores	Almacena la información de los instructores que imparten los cursos.	id_instructor	ID Instructor	INT	Primaria
		nombre	Nombre	VARCHAR(100)	
		apellido	Apellido	VARCHAR(100)	
		especialidad	Especialidad	VARCHAR(100)	
		email	Email	VARCHAR(100)	Único
Curso_Instructor	Almacena la relación entre cursos e instructores (muchos a muchos).	id_curso_instructor	ID Curso_Instr.	INT	Primaria
		id_curso	ID Curso	INT	Foránea
		id_instructor	ID Instructor	INT	Foránea
Lecciones	Almacena la información de las lecciones que componen cada curso.	id_leccion	ID Lección	INT	Primaria
		id_curso	ID Curso	INT	Foránea
		titulo_leccion	Título Lección	VARCHAR(100)	
		contenido_leccion	Contenido Lección	TEXT	
		orden	Orden	INT	
Progreso_Leccion	Almacena el progreso de los usuarios en las lecciones de cada curso.	id_progreso	ID Progreso	INT	Primaria
		id_inscripcion	ID Inscripción	INT	Foránea
		id_leccion	ID Lección	INT	Foránea
		fecha_completado	Fecha Completado	DATETIME	

Recursos	Almacena la información de los recursos adicionales para las lecciones (PDF, audios, etc.).	id_recurso	ID Recurso	INT	Primaria
		id_leccion	ID Lección	INT	Foránea
		nombre_recurso	Nombre Recurso	VARCHAR(100)	
		tipo_recurso	Tipo Recurso	VARCHAR(50)	
		url_recurso	URL Recurso	VARCHAR(255)	
Certificados	Almacena la información sobre los certificados que se entregan al finalizar un curso.	id_certificado	ID Certificado	INT	Primaria
		id_inscripcion	ID Inscripción	INT	Foránea
		fecha_emision	Fecha Emisión	DATETIME	
		url_certificado	URL Certificado	VARCHAR(255)	
Foros	Almacena las secciones de foro para discusiones	id_foro	ID Foro	INT	Primaria
		id_curso	ID Curso	INT	Foránea
		nombre_foro	Nombre Foro	VARCHAR(100)	
Comentarios_Foro	Almacena los comentarios dentro de los foros	id_comentario	ID Comentario	INT	Primaria
		id_foro	ID Foro	INT	Foránea
		id_usuario	ID Usuario	INT	Foránea
		comentario	Comentario	TEXT	
		fecha_comentario	Fecha Comentario	DATETIME	
Descuentos		id_descuento	ID Descuento	INT	Primaria

	Almacena la información sobre los descuentos disponibles	codigo_descuento	Código Descuento	VARCHAR(50)	Único
		porcentaje_descuento	Porcentaje Desc.	DECIMAL(5,2)	
		fecha_inicio	Fecha Inicio	DATETIME	
		fecha_fin	Fecha Fin	DATETIME	
Descuento_Curso	Almacena la relación entre cursos y descuentos (muchos a muchos)	id_descuento_curso	ID Descuento_Cur.	INT	Primaria
		id_descuento	ID Descuento	INT	Foránea
		id_curso	ID Curso	INT	Foránea
Paises	Almacena la información de los paises de los usuarios	id_pais	ID País	INT	Primaria
		nombre_pais	Nombre País	VARCHAR(100)	
Ciudades	Almacena la información de las ciudades de los usuarios	id_ciudad	ID Ciudad	INT	Primaria
		id_pais	ID País	INT	Foránea
		nombre_ciudad	Nombre Ciudad	VARCHAR(100)	

## 6. Diagrama Entidad -Relación





## 7. Scripts de creación de objetos en base de datos

### 7.1. FN\_Calcular\_Total\_Pagos\_Usuario

```
USE CorcheaAcademyDB;

DELIMITER $$

DROP FUNCTION IF EXISTS Calcular_Total_Pagos_Usuario;

CREATE FUNCTION Calcular_Total_Pagos_Usuario(id_usuario_input INT)
RETURNS DECIMAL(10, 2)
DETERMINISTIC
BEGIN
    DECLARE total DECIMAL(10, 2);

    -- Se suma el monto de las inscripciones del usuario
    SELECT SUM(p.monto) INTO total
    FROM Pagos p
    JOIN Inscripciones i ON p.id_inscripcion = i.id_inscripcion
    WHERE i.id_usuario = id_usuario_input;

    RETURN COALESCE(total, 0.00);
END$$

DELIMITER ;

-- Ejemplo de uso:
SELECT Calcular_Total_Pagos_Usuario(1) AS Total_Pagado;
```

### 7.2. FN\_Obtener\_Cantidad\_Inscripciones

```
USE CorcheaAcademyDB;

DELIMITER $$

DROP FUNCTION IF EXISTS Obtener_Cantidad_Inscripciones$$
```

```

CREATE FUNCTION Obtener_Cantidad_Inscripciones(id_usuario_input INT)
RETURNS INT
DETERMINISTIC
BEGIN
    DECLARE cantidad INT;
    SELECT COUNT(*) INTO cantidad
    FROM Inscripciones
    WHERE id_usuario = id_usuario_input;
    RETURN cantidad;
END$$

DELIMITER ;

-- Ejemplo de uso:
SELECT Obtener_Cantidad_Inscripciones(1) AS Total_Inscripciones;

```

### 7.3. SP\_Actualizar\_Nivel\_Curso

```

USE CorcheaAcademyDB;

DELIMITER $$

DROP PROCEDURE IF EXISTS SP_Actualizar_Nivel_Curso$$

CREATE PROCEDURE SP_Actualizar_Nivel_Curso(
    IN id_curso_input INT,
    IN id_nivel_nuevo INT
)
BEGIN
    -- Validar si el curso existe
    IF EXISTS (SELECT 1 FROM Cursos WHERE id_curso = id_curso_input) THEN
        -- Validar si el nivel nuevo existe
        IF EXISTS (SELECT 1 FROM Niveles WHERE id_nivel = id_nivel_nuevo) THEN
            -- Actualizar el nivel del curso
            UPDATE Cursos
            SET id_nivel = id_nivel_nuevo

```

```

        WHERE id_curso = id_curso_input;
    ELSE
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'El nivel especificado no existe.';
    END IF;
ELSE
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'El curso especificado no existe.';
END IF;
END$$

DELIMITER ;

-- Ejemplo de uso:
-- CALL SP_Actualizar_Nivel_Curso(1, 2); -- Cambia el nivel del curso con ID 1 al nivel con ID 2

/*
Descripción:
Actualiza el nivel de un curso existente.
Objetivo:
Permitir la modificación del nivel de un curso, validando que tanto el curso como el nuevo nivel existan en la base de
datos.

Tablas involucradas:
- Cursos
- Niveles
*/

```

## 7.4. SP\_Agregar\_Comentario\_Foro

```

USE CorcheaAcademyDB;

DELIMITER $$

DROP PROCEDURE IF EXISTS SP_Agregar_Comentario_Foro$$

CREATE PROCEDURE SP_Agregar_Comentario_Foro(

```

```

    IN id_foro_input INT,
    IN id_usuario_input INT,
    IN comentario_input TEXT
)
BEGIN
    -- Validar si el foro existe
    IF EXISTS (SELECT 1 FROM Foros WHERE id_foro = id_foro_input) THEN
        -- Validar si el usuario existe
        IF EXISTS (SELECT 1 FROM Usuarios WHERE id_usuario = id_usuario_input) THEN
            -- Insertar el comentario en el foro
            INSERT INTO Comentarios_Foro (id_foro, id_usuario, comentario, fecha_comentario)
            VALUES (id_foro_input, id_usuario_input, comentario_input, NOW());
        ELSE
            SIGNAL SQLSTATE '45000'
            SET MESSAGE_TEXT = 'El usuario especificado no existe.';
        END IF;
    ELSE
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'El foro especificado no existe.';
    END IF;
END$$

```

DELIMITER ;

-- Ejemplo de uso:

-- CALL SP\_Agregar\_Comentario\_Foro(1, 1, 'Este es un comentario de prueba.');

/\*

Descripción:

Agrega un nuevo comentario a un foro específico, realizado por un usuario dado.

Objetivo:

Simplificar la lógica para agregar un comentario, validando que el foro y el usuario existen.

Tablas involucradas:

- Foros
- Usuarios
- Comentarios\_Foro

```
*/
```

## 7.5. SP\_Registrar\_Inscripcion

```
USE CorcheaAcademyDB;

DELIMITER $$

DROP PROCEDURE IF EXISTS SP_Registrar_Inscripcion$$

CREATE PROCEDURE SP_Registrar_Inscripcion(
    IN id_usuario_input INT,
    IN id_curso_input INT
)
BEGIN
    -- Validar si el usuario existe y tiene el rol de estudiante
    IF EXISTS (
        SELECT 1 FROM Usuarios
        WHERE id_usuario = id_usuario_input AND id_rol = 3
    ) THEN
        -- Validar si el curso existe
        IF EXISTS (
            SELECT 1 FROM Cursos WHERE id_curso = id_curso_input
        ) THEN
            -- Insertar la inscripción
            INSERT INTO Inscripciones (id_usuario, id_curso, fecha_inscripcion)
            VALUES (id_usuario_input, id_curso_input, CURDATE());
        ELSE
            SIGNAL SQLSTATE '45000'
            SET MESSAGE_TEXT = 'El curso especificado no existe.';
        END IF;
    ELSE
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'El usuario no tiene permisos para inscribirse o no existe.';
    END IF;
END$$
```

```

DELIMITER ;

-- Ejemplo de uso:
-- CALL SP_Registrar_Inscripcion(1, 2);

/*
Descripción:
Inserta una nueva inscripción para un usuario y un curso específicos.
Objetivo:
Simplificar la lógica para registrar una inscripción, validando que el curso existe y que el usuario tiene un rol permitido (Estudiante).

Tablas involucradas:
- Usuarios
- Cursos
- Inscripciones
*/

```

## 7.6. trg\_registrar\_fecha\_completado

```

USE CorcheaAcademyDB;

DELIMITER $$

DROP TRIGGER IF EXISTS trg_registrar_fecha_completado $$

CREATE TRIGGER trg_registrar_fecha_completado
BEFORE UPDATE ON Progreso_Leccion
FOR EACH ROW
BEGIN
    -- Registrar la fecha y hora actuales cuando se actualiza el estado a 'Completado'
    IF NEW.fecha_completado IS NOT NULL AND OLD.fecha_completado IS NULL THEN
        SET NEW.fecha_completado = NOW();
    END IF;
END $$

```

```

DELIMITER ;

/*
Descripción:
Este trigger registra la fecha y hora actuales en el campo `fecha_completado` de la tabla `Progreso_Leccion` cuando un
estudiante completa una lección.
Objetivo:
Automatizar el registro de la fecha de finalización de una lección, facilitando el seguimiento del progreso del estudiante.

Tabla involucrada:
- Progreso_Leccion

Evento:
- BEFORE UPDATE en Progreso_Leccion
*/

```

## 7.7. trg\_verificar\_pago\_inscripcion

```

USE CorcheaAcademyDB;

DELIMITER $$

DROP TRIGGER IF EXISTS trg_verificar_pago_inscripcion $$

CREATE TRIGGER trg_verificar_pago_inscripcion
BEFORE DELETE ON Inscripciones
FOR EACH ROW
BEGIN
    -- Verificar si existe un pago asociado a la inscripción
    IF EXISTS (SELECT 1 FROM Pagos WHERE id_inscripcion = OLD.id_inscripcion) THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'No se puede eliminar la inscripción porque tiene un pago asociado.';
    END IF;
END $$

DELIMITER ;

```

```

/*
Descripción:
Este trigger impide la eliminación de una inscripción si existe un pago asociado a ella.
Objetivo:
Mantener la integridad de los datos, evitando la eliminación accidental de inscripciones que ya han sido pagadas.

Tabla involucrada:
- Inscripciones
- Pagos

Evento:
- BEFORE DELETE en Inscripciones
*/

```

## 7.8. Vista\_Comentarios\_Foro

```

USE CorcheaAcademyDB;

DROP VIEW IF EXISTS Vista_Comentarios_Foro;

CREATE VIEW Vista_Comentarios_Foro AS
SELECT
    f.id_foro AS ID_Foro,
    c.id_curso AS ID_Curso,
    c.nombre_curso AS Nombre_Curso,
    cf.id_comentario AS ID_Comentario,
    u.id_usuario AS ID_Usuario,
    u.nombre AS Nombre_Usuario,
    u.apellido AS Apellido_Usuario,
    cf.comentario AS Comentario,
    cf.fecha_comentario AS Fecha_Comentario
FROM
    Foros f
JOIN Comentarios_Foro cf ON f.id_foro = cf.id_foro
JOIN Usuarios u ON cf.id_usuario = u.id_usuario
JOIN Cursos c ON f.id_curso = c.id_curso;

```



```
-- Consulta de la vista
-- SELECT * FROM Vista_Comentarios_Foro;

/*
Descripción:
Esta vista muestra los comentarios realizados en los foros, incluyendo el ID del foro, el ID y nombre del curso al que pertenece el foro, el ID del comentario, el ID, nombre, y apellido del usuario que realizó el comentario, el contenido del comentario y la fecha en que se realizó.
Objetivo:
Facilitar la consulta de los comentarios en los foros, mostrando información consolidada de las tablas Foros, Comentarios_Foro, Usuarios y Cursos.

Tablas involucradas:
- Foros
- Comentarios_Foro
- Usuarios
- Cursos
*/
```

## 7.9. Vista\_Detalle\_Cursos

```
USE CorcheaAcademyDB;

DROP VIEW IF EXISTS Vista_Detalle_Cursos;

CREATE VIEW Vista_Detalle_Cursos AS
SELECT
    c.id_curso AS ID_Curso,
    c.nombre_curso AS Nombre_Curso,
    c.descripcion AS Descripcion,
    c.precio AS Precio,
    cat.nombre_categoria AS Categoria,
    n.nombre_nivel AS Nivel
FROM
    Cursos c
JOIN Categorías cat ON c.id_categoria = cat.id_categoria
JOIN Niveles n ON c.id_nivel = n.id_nivel;
```

```
-- Consulta de la vista
-- SELECT * FROM Vista_Detalle_Cursos;
```

```
/*
```

Descripción:

Esta vista muestra información detallada de los cursos, incluyendo el nombre del curso, descripción, precio, categoría y nivel.

Objetivo:

Facilitar la consulta rápida de información completa de los cursos, combinando datos de las tablas Cursos, Categorías y Niveles.

Tablas involucradas:

- Cursos
- Categorías
- Niveles

```
*/
```

## 7.10. Vista\_Instructores\_Cursos

```
USE CorcheaAcademyDB;
```

```
DROP VIEW IF EXISTS Vista_Instructores_Cursos;
```

```
CREATE VIEW Vista_Instructores_Cursos AS
```

```
SELECT
```

```
    i.id_instructor AS ID_Instructor,  
    i.nombre AS Nombre_Instructor,  
    i.apellido AS Apellido_Instructor,  
    c.id_curso AS ID_Curso,  
    c.nombre_curso AS Nombre_Curso
```

```
FROM
```

```
    Instructores i
```

```
JOIN Curso_Instructor ci ON i.id_instructor = ci.id_instructor
```

```
JOIN Cursos c ON ci.id_curso = c.id_curso;
```

```
-- Consulta de la vista
```

```
-- SELECT * FROM Vista_Instructores_Cursos;
```

```

/*
Descripción:
Esta vista muestra la relación entre instructores y los cursos que imparten, incluyendo el ID y nombre del instructor, y el ID y nombre del curso.
Objetivo:
Facilitar la consulta de qué instructor imparte cada curso, combinando datos de las tablas Instructores, Curso_Instructor y Cursos.

Tablas involucradas:
- Instructores
- Curso_Instructor
- Cursos
*/

```

## 7.11. Vista\_Lecciones\_Por\_Curso

```

USE CorcheaAcademyDB;

DROP VIEW IF EXISTS Vista_Lecciones_Por_Curso;

CREATE VIEW Vista_Lecciones_Por_Curso AS
SELECT
    c.id_curso AS ID_Curso,
    c.nombre_curso AS Nombre_Curso,
    l.id_leccion AS ID_Leccion,
    l.titulo_leccion AS Titulo_Leccion,
    l.orden AS Orden_Leccion
FROM
    Cursos c
JOIN Lecciones l ON c.id_curso = l.id_curso;

-- Consulta de la vista
-- SELECT * FROM Vista_Lecciones_Por_Curso;

```

```
/*
Descripción:
Esta vista muestra las lecciones asociadas a cada curso, incluyendo el ID y nombre del curso, y el ID, título y orden de la lección.
Objetivo:
Facilitar la consulta de las lecciones que componen cada curso, combinando datos de las tablas Cursos y Lecciones.

Tablas involucradas:
- Cursos
- Lecciones
*/
```

## 7.12. Vista\_Resumen\_Pagos

```
USE CorcheaAcademyDB;

DROP VIEW IF EXISTS Vista_Resumen_Pagos;

CREATE VIEW Vista_Resumen_Pagos AS
SELECT
    p.id_pago AS ID_Pago,
    u.id_usuario AS ID_Usuario,
    u.nombre AS Nombre_Usuario,
    u.apellido AS Apellido_Usuario,
    c.id_curso AS ID_Curso,
    c.nombre_curso AS Nombre_Curso,
    p.monto AS Monto_Pagado,
    p.fecha_pago AS Fecha_Pago,
    p.metodo_pago AS Metodo_Pago
FROM
    Pagos p
JOIN Inscripciones i ON p.id_inscripcion = i.id_inscripcion
JOIN Usuarios u ON i.id_usuario = u.id_usuario
JOIN Cursos c ON i.id_curso = c.id_curso;

-- Consulta de la vista
-- SELECT * FROM Vista_Resumen_Pagos;
```

```
/*
Descripción:
Esta vista muestra un resumen de los pagos realizados, incluyendo información del usuario, el curso al que corresponde el pago, el monto, la fecha y el método de pago.
Objetivo:
Facilitar la consulta de los pagos realizados, mostrando información consolidada de las tablas Pagos, Inscripciones, Usuarios y Cursos.

Tablas involucradas:
- Pagos
- Inscripciones
- Usuarios
- Cursos
*/
```

## 8. Scripts de inserción de datos

```
USE CorcheaAcademyDB;

-- Población de la tabla Roles
INSERT INTO Roles (nombre_rol) VALUES
('Administrador'),
('Instructor'),
('Estudiante'),
('Invitado'),
('Editor'),
('Moderador'),
('Coordinador'),
('Soporte Técnico'),
('Gestor de Contenido'),
('Visitante');

-- Población de la tabla Categorías
```

```
INSERT INTO Categorías (nombre_categoria) VALUES
('Bajo'),
('Batería'),
('Piano'),
('Técnica Vocal'),
('Guitarra'),
('Producción Musical'),
('Composición'),
('DJing'),
('Teoría Musical'),
('Improvisación');

-- Población de la tabla Niveles
INSERT INTO Niveles (nombre_nivel) VALUES
('Principiante'),
('Intermedio'),
('Avanzado');

-- Población de la tabla Países
INSERT INTO Países (nombre_pais) VALUES
('Argentina'),
('España'),
('México'),
('Colombia'),
('Estados Unidos'),
('Otro');

-- Población de la tabla Ciudades
INSERT INTO Ciudades (id_pais, nombre_ciudad) VALUES
(1, 'Buenos Aires'),
(1, 'Córdoba'),
(2, 'Madrid'),
(2, 'Barcelona'),
(3, 'Ciudad de México'),
(3, 'Guadalajara'),
(4, 'Bogotá'),
(4, 'Medellín'),
```

```
(5, 'Nueva York'),  
(5, 'Los Ángeles'),  
(6, 'Otra');
```

-- Población de la tabla Usuarios

```
INSERT INTO Usuarios (nombre, apellido, email, contraseña, fecha_registro, id_rol) VALUES  
( 'Carlos', 'Pérez', 'carlos.perez@gmail.com', 'pass1234', '2023-10-01', 3),  
( 'Ana', 'López', 'ana.lopez@gmail.com', 'pass1234', '2023-09-15', 2),  
( 'Luis', 'García', 'luis.garcia@gmail.com', 'pass1234', '2023-08-20', 1),  
( 'María', 'Torres', 'maria.torres@gmail.com', 'pass1234', '2023-09-10', 3),  
( 'Pedro', 'Sánchez', 'pedro.sanchez@gmail.com', 'pass1234', '2023-09-05', 2),  
( 'Sandra', 'Ruiz', 'sandra.ruiz@gmail.com', 'pass1234', '2023-08-30', 3),  
( 'Juan', 'Martínez', 'juan.martinez@gmail.com', 'pass1234', '2023-09-25', 3),  
( 'Lucía', 'Gómez', 'lucia.gomez@gmail.com', 'pass1234', '2023-09-01', 3),  
( 'Raúl', 'Díaz', 'raul.diaz@gmail.com', 'pass1234', '2023-10-02', 3),  
( 'Elena', 'Romero', 'elena.romero@gmail.com', 'pass1234', '2023-09-12', 3),  
( 'Administrador', 'Principal', 'admin@corcheaacademy.com', 'adminpass', '2023-01-01', 1),  
( 'Roberto', 'Gomez', 'roberto.gomez@gmail.com', 'pass1234', '2023-09-15', 3),  
( 'Laura', 'Fernandez', 'laura.fernandez@gmail.com', 'pass1234', '2023-09-18', 3),  
( 'Miguel', 'Herrera', 'miguel.herrera@gmail.com', 'pass1234', '2023-09-20', 3),  
( 'Sofia', 'Vargas', 'sofia.vargas@gmail.com', 'pass1234', '2023-09-22', 3),  
( 'Javier', 'Medina', 'javier.medina@gmail.com', 'pass1234', '2023-09-25', 3),  
( 'Isabel', 'Castro', 'isabel.castro@gmail.com', 'pass1234', '2023-09-28', 3),  
( 'Alejandro', 'Ortega', 'alejandro.ortega@gmail.com', 'pass1234', '2023-10-01', 3),  
( 'Carmen', 'Silva', 'carmen.silva@gmail.com', 'pass1234', '2023-10-03', 3),  
( 'Ricardo', 'Navarro', 'ricardo.navarro@gmail.com', 'pass1234', '2023-10-05', 3);
```

-- Población de la tabla Cursos

```
INSERT INTO Cursos (nombre_curso, descripcion, precio, fecha_creacion, id_categoria, id_nivel) VALUES  
( 'Aprende Bajo Básico', 'Curso introductorio para aprender bajo eléctrico', 49.99, '2023-09-01', 1, 1),  
( 'Técnicas Avanzadas de Batería', 'Curso avanzado de ritmos y fills en batería', 79.99, '2023-08-15', 2, 3),  
( 'Piano para Principiantes', 'Introducción a acordes y escalas en piano', 59.99, '2023-07-20', 3, 1),  
( 'Técnica Vocal Nivel 1', 'Técnicas básicas para mejorar la voz', 39.99, '2023-06-25', 4, 1),  
( 'Composición Musical', 'Aprende a componer canciones desde cero', 99.99, '2023-08-10', 7, 2),  
( 'Producción Musical en DAW', 'Domina el software de producción musical', 119.99, '2023-09-15', 6, 3),  
( 'Improvisación en Jazz', 'Técnicas de improvisación en jazz', 89.99, '2023-08-30', 9, 2),  
( 'DJing para Principiantes', 'Curso básico para ser DJ', 69.99, '2023-07-05', 8, 1),
```

```
('Teoría Musical Completa', 'Curso completo de teoría musical', 109.99, '2023-09-20', 9, 3),
('Batería para Niños', 'Curso introductorio de batería para niños', 29.99, '2023-10-01', 2, 1),
('Guitarra Acústica Intermedia', 'Progresiones de acordes y arpeggios', 69.99, '2023-09-10', 5, 2),
('Mezcla y Masterización', 'Técnicas profesionales de mezcla y masterización', 129.99, '2023-09-15', 6, 3),
('Armonía Moderna', 'Conceptos avanzados de armonía musical', 79.99, '2023-09-20', 7, 2),
('Sintetizadores y Diseño de Sonido', 'Crea tus propios sonidos con sintetizadores', 89.99, '2023-09-25', 6, 2),
('Canto para Musicales', 'Técnicas de canto aplicadas a teatro musical', 49.99, '2023-10-01', 4, 2),
('Introducción a la Orquestación', 'Aprende a escribir música para orquesta', 99.99, '2023-10-05', 7, 3),
('Guitarra Eléctrica Blues', 'Fundamentos del blues en guitarra eléctrica', 59.99, '2023-10-10', 5, 2),
('Producción de Música Electrónica', 'Crea música electrónica desde cero', 109.99, '2023-10-15', 6, 3),
('Piano Jazz', 'Acordes y escalas para improvisación en jazz', 79.99, '2023-10-20', 3, 2),
('Técnicas de Grabación en Estudio', 'Aprende a grabar instrumentos y voces', 89.99, '2023-10-25', 6, 2);
```

-- Población de la tabla Inscripciones

```
INSERT INTO Inscripciones (id_usuario, id_curso, fecha_inscripcion) VALUES
```

```
(1, 1, '2023-10-05'),
(2, 2, '2023-09-20'),
(3, 3, '2023-08-25'),
(4, 4, '2023-09-12'),
(5, 5, '2023-09-18'),
(6, 6, '2023-09-25'),
(7, 7, '2023-09-30'),
(8, 8, '2023-10-02'),
(9, 9, '2023-10-05'),
(10, 10, '2023-10-10'),
(12, 1, '2023-10-06'),
(13, 3, '2023-10-07'),
(14, 5, '2023-10-08'),
(15, 7, '2023-10-09'),
(16, 9, '2023-10-10'),
(17, 11, '2023-10-11'),
(18, 13, '2023-10-12'),
(19, 15, '2023-10-13'),
(20, 17, '2023-10-14'),
(1, 12, '2023-10-06'),
(3, 14, '2023-10-07'),
(5, 16, '2023-10-08'),
```



```
(7, 18, '2023-10-09'),  
(9, 20, '2023-10-10'),  
(11, 2, '2023-10-11'),  
(13, 4, '2023-10-12'),  
(15, 6, '2023-10-13'),  
(17, 8, '2023-10-14');
```

-- Población de la tabla Pagos

```
INSERT INTO Pagos (id_inscripcion, monto, fecha_pago, metodo_pago) VALUES
```

```
(1, 49.99, '2023-10-06', 'Tarjeta de Crédito'),  
(2, 79.99, '2023-09-21', 'PayPal'),  
(3, 59.99, '2023-08-26', 'Transferencia Bancaria'),  
(4, 39.99, '2023-09-13', 'Tarjeta de Crédito'),  
(5, 99.99, '2023-09-19', 'PayPal'),  
(6, 119.99, '2023-09-26', 'Transferencia Bancaria'),  
(7, 89.99, '2023-10-01', 'PayPal'),  
(8, 69.99, '2023-10-03', 'Tarjeta de Crédito'),  
(9, 109.99, '2023-10-06', 'Transferencia Bancaria'),  
(10, 29.99, '2023-10-11', 'PayPal'),  
(11, 49.99, '2023-10-07', 'Tarjeta de Crédito'),  
(12, 59.99, '2023-10-08', 'PayPal'),  
(13, 99.99, '2023-10-09', 'Tarjeta de Crédito'),  
(14, 79.99, '2023-10-10', 'Transferencia Bancaria'),  
(15, 69.99, '2023-10-11', 'PayPal'),  
(16, 129.99, '2023-10-12', 'Tarjeta de Crédito'),  
(17, 79.99, '2023-10-13', 'Transferencia Bancaria'),  
(18, 89.99, '2023-10-14', 'PayPal'),  
(19, 49.99, '2023-10-15', 'Tarjeta de Crédito'),  
(20, 69.99, '2023-10-07', 'Transferencia Bancaria'),  
(21, 79.99, '2023-10-08', 'PayPal'),  
(22, 89.99, '2023-10-09', 'Tarjeta de Crédito'),  
(23, 109.99, '2023-10-10', 'Transferencia Bancaria'),  
(24, 79.99, '2023-10-11', 'PayPal'),  
(25, 39.99, '2023-10-12', 'Tarjeta de Crédito'),  
(26, 119.99, '2023-10-13', 'Transferencia Bancaria'),  
(27, 69.99, '2023-10-14', 'PayPal'),  
(28, 99.99, '2023-10-15', 'Tarjeta de Crédito');
```

-- Población de la tabla Instructores

```
INSERT INTO Instructores (nombre, apellido, especialidad, email) VALUES
('Pedro', 'Sánchez', 'Bajo Eléctrico', 'pedro.sanchez.instructor@gmail.com'),
('Ana', 'López', 'Batería', 'ana.lopez.instructor@gmail.com'),
('María', 'Torres', 'Técnica Vocal', 'maria.torres.instructor@gmail.com'),
('Pablo', 'Iglesias', 'Guitarra', 'pablo.iglesias.instructor@gmail.com'),
('Luisa', 'Ramírez', 'Piano', 'luisa.ramirez.instructor@gmail.com'),
('Jorge', 'Silva', 'Producción Musical', 'jorge.silva.instructor@gmail.com'),
('Clara', 'Mendoza', 'Composición', 'clara.mendoza.instructor@gmail.com'),
('Diego', 'Fuentes', 'DJing', 'diego.fuentes.instructor@gmail.com'),
('Mariana', 'Ortega', 'Teoría Musical', 'mariana.ortega.instructor@gmail.com'),
('Fernando', 'Soto', 'Improvisación', 'fernando.soto.instructor@gmail.com');
```

-- Población de la tabla Curso\_Instructor

```
INSERT INTO Curso_Instructor (id_curso, id_instructor) VALUES
(1, 1),
(2, 2),
(3, 5),
(4, 3),
(5, 7),
(6, 6),
(7, 10),
(8, 8),
(9, 9),
(10, 2),
(11, 4),
(12, 6),
(13, 9),
(14, 6),
(15, 3),
(16, 7),
(17, 4),
(18, 6),
(19, 5),
(20, 6);
```

```
-- Población de la tabla Lecciones
INSERT INTO Lecciones (id_curso, titulo_leccion, contenido_leccion, orden) VALUES
-- Lecciones para el curso con id_curso = 1
(1, 'Introducción al Bajo', 'Contenido de la lección 1 para el curso 1', 1),
(1, 'Postura y Técnica', 'Contenido de la lección 2 para el curso 1', 2),
-- Lecciones para el curso con id_curso = 2
(2, 'Rudimentos Esenciales', 'Contenido de la lección 1 para el curso 2', 1),
(2, 'Control de Baquetas', 'Contenido de la lección 2 para el curso 2', 2),
-- Lecciones para el curso con id_curso = 3
(3, 'Introducción al Piano', 'Contenido de la lección 1 para el curso 3', 1),
(3, 'Acordes Mayores y Menores', 'Contenido de la lección 2 para el curso 3', 2),
-- Lecciones para el curso con id_curso = 4
(4, 'Técnicas de Respiración', 'Contenido de la lección 1 para el curso 4', 1),
(4, 'Calentamiento Vocal', 'Contenido de la lección 2 para el curso 4', 2),
-- Lecciones para el curso con id_curso = 5
(5, 'Análisis de Canciones', 'Contenido de la lección 1 para el curso 5', 1),
(5, 'Creación de Melodías', 'Contenido de la lección 2 para el curso 5', 2),
-- Lecciones para el curso con id_curso = 6
(6, 'Fundamentos de DAWs', 'Contenido de la lección 1 para el curso 6', 1),
(6, 'Grabación y Edición', 'Contenido de la lección 2 para el curso 6', 2),
-- Lecciones para el curso con id_curso = 7
(7, 'Técnicas de Improvisación', 'Contenido de la lección 1 para el curso 7', 1),
(7, 'Modos Griegos', 'Contenido de la lección 2 para el curso 7', 2),
-- Lecciones para el curso con id_curso = 8
(8, 'Introducción al DJing', 'Contenido de la lección 1 para el curso 8', 1),
(8, 'Beatmatching', 'Contenido de la lección 2 para el curso 8', 2),
-- Lecciones para el curso con id_curso = 9
(9, 'Intervalos y Acordes', 'Contenido de la lección 1 para el curso 9', 1),
(9, 'Armonía Funcional', 'Contenido de la lección 2 para el curso 9', 2),
-- Lecciones para el curso con id_curso = 10
(10, 'Ritmos para Niños', 'Contenido de la lección 1 para el curso 10', 1),
(10, 'Coordinación Básica', 'Contenido de la lección 2 para el curso 10', 2),
-- Lecciones para el curso con id_curso = 11
(11, 'Acordes con Cejilla', 'Contenido de la lección 1 para el curso 11', 1),
(11, 'Arpeggios', 'Contenido de la lección 2 para el curso 11', 2),
-- Lecciones para el curso con id_curso = 12
(12, 'Ecualización', 'Contenido de la lección 1 para el curso 12', 1),
```

```

(12, 'Compresión', 'Contenido de la lección 2 para el curso 12', 2),
-- Lecciones para el curso con id_curso = 13
(13, 'Armonía por Terceras', 'Contenido de la lección 1 para el curso 13', 1),
(13, 'Progresiones de Acordes', 'Contenido de la lección 2 para el curso 13', 2),
-- Lecciones para el curso con id_curso = 14
(14, 'Tipos de Sintetizadores', 'Contenido de la lección 1 para el curso 14', 1),
(14, 'Diseño de Sonido', 'Contenido de la lección 2 para el curso 14', 2),
-- Lecciones para el curso con id_curso = 15
(15, 'Técnica Vocal para Musicales', 'Contenido de la lección 1 para el curso 15', 1),
(15, 'Interpretación Escénica', 'Contenido de la lección 2 para el curso 15', 2),
-- Lecciones para el curso con id_curso = 16
(16, 'Introducción a la Orquestación', 'Contenido de la lección 1 para el curso 16', 1),
(16, 'Instrumentación', 'Contenido de la lección 2 para el curso 16', 2),
-- Lecciones para el curso con id_curso = 17
(17, 'Licks de Blues', 'Contenido de la lección 1 para el curso 17', 1),
(17, 'Improvisación en Blues', 'Contenido de la lección 2 para el curso 17', 2),
-- Lecciones para el curso con id_curso = 18
(18, 'Producción de Beats', 'Contenido de la lección 1 para el curso 18', 1),
(18, 'Sampling', 'Contenido de la lección 2 para el curso 18', 2),
-- Lecciones para el curso con id_curso = 19
(19, 'Acordes de Jazz', 'Contenido de la lección 1 para el curso 19', 1),
(19, 'Escalas para Jazz', 'Contenido de la lección 2 para el curso 19', 2),
-- Lecciones para el curso con id_curso = 20
(20, 'Microfonía', 'Contenido de la lección 1 para el curso 20', 1),
(20, 'Técnicas de Grabación', 'Contenido de la lección 2 para el curso 20', 2);

-- Población de la tabla Progreso_Leccion
INSERT INTO Progreso_Leccion (id_inscripcion, id_leccion, fecha_completado) VALUES
-- Progreso para la inscripción con id_inscripcion = 1 (Curso 1, Lecciones 1 y 2)
(1, 1, '2023-10-06'),
(1, 2, '2023-10-08'),
-- Progreso para la inscripción con id_inscripcion = 2 (Curso 2, Lecciones 3 y 4)
(2, 3, '2023-09-22'),
(2, 4, '2023-09-24'),
-- Progreso para la inscripción con id_inscripcion = 3 (Curso 3, Lecciones 5 y 6)
(3, 5, '2023-08-28'),
(3, 6, '2023-08-30'),

```

```
-- Progreso para la inscripción con id_inscripcion = 4 (Curso 4, Lecciones 7 y 8)
(4, 7, '2023-09-15'),
(4, 8, '2023-09-17'),
-- Progreso para la inscripción con id_inscripcion = 5 (Curso 5, Lecciones 9 y 10)
(5, 9, '2023-09-21'),
(5, 10, '2023-09-23'),
-- Progreso para la inscripción con id_inscripcion = 6 (Curso 6, Lecciones 11 y 12)
(6, 11, '2023-09-28'),
(6, 12, '2023-09-30'),
-- Progreso para la inscripción con id_inscripcion = 7 (Curso 7, Lecciones 13 y 14)
(7, 13, '2023-10-03'),
(7, 14, '2023-10-05'),
-- Progreso para la inscripción con id_inscripcion = 8 (Curso 8, Lecciones 15 y 16)
(8, 15, '2023-10-05'),
(8, 16, '2023-10-07'),
-- Progreso para la inscripción con id_inscripcion = 9 (Curso 9, Lecciones 17 y 18)
(9, 17, '2023-10-08'),
(9, 18, '2023-10-10'),
-- Progreso para la inscripción con id_inscripcion = 10 (Curso 10, Lecciones 19 y 20)
(10, 19, '2023-10-13'),
(10, 20, '2023-10-15'),
-- Progreso para la inscripción con id_inscripcion = 11 (Curso 1, Lecciones 1 y 2)
(11, 1, '2023-10-08'),
(11, 2, '2023-10-10'),
-- Progreso para la inscripción con id_inscripcion = 12 (Curso 12, Lecciones 23 y 24)
(12, 23, '2023-10-09'),
(12, 24, '2023-10-11'),
-- Progreso para la inscripción con id_inscripcion = 13 (Curso 14, Lecciones 27 y 28)
(13, 27, '2023-10-10'),
(13, 28, '2023-10-12'),
-- Progreso para la inscripción con id_inscripcion = 14 (Curso 16, Lecciones 31 y 32)
(14, 31, '2023-10-11'),
(14, 32, '2023-10-13'),
-- Progreso para la inscripción con id_inscripcion = 15 (Curso 18, Lecciones 35 y 36)
(15, 35, '2023-10-12'),
(15, 36, '2023-10-14'),
-- Progreso para la inscripción con id_inscripcion = 16 (Curso 20, Lecciones 39 y 40)
```

```
(16, 39, '2023-10-13'),
(16, 40, '2023-10-15'),
-- Progreso para la inscripción con id_inscripcion = 17 (Curso 2, Lecciones 3 y 4)
(17, 3, '2023-10-14'),
(17, 4, '2023-10-16'),
-- Progreso para la inscripción con id_inscripcion = 18 (Curso 4, Lecciones 7 y 8)
(18, 7, '2023-10-15'),
(18, 8, '2023-10-17'),
-- Progreso para la inscripción con id_inscripcion = 19 (Curso 6, Lecciones 11 y 12)
(19, 11, '2023-10-16'),
(19, 12, '2023-10-18'),
-- Progreso para la inscripción con id_inscripcion = 20 (Curso 8, Lecciones 15 y 16)
(20, 15, '2023-10-08'),
(20, 16, '2023-10-10'),
-- Progreso para la inscripción con id_inscripcion = 21 (Curso 10, Lecciones 19 y 20)
(21, 19, '2023-10-09'),
(21, 20, '2023-10-11'),
-- Progreso para la inscripción con id_inscripcion = 22 (Curso 11, Lecciones 21 y 22)
(22, 21, '2023-10-10'),
(22, 22, '2023-10-12'),
-- Progreso para la inscripción con id_inscripcion = 23 (Curso 13, Lecciones 25 y 26)
(23, 25, '2023-10-11'),
(23, 26, '2023-10-13'),
-- Progreso para la inscripción con id_inscripcion = 24 (Curso 15, Lecciones 29 y 30)
(24, 29, '2023-10-12'),
(24, 30, '2023-10-14'),
-- Progreso para la inscripción con id_inscripcion = 25 (Curso 17, Lecciones 33 y 34)
(25, 33, '2023-10-13'),
(25, 34, '2023-10-15'),
-- Progreso para la inscripción con id_inscripcion = 26 (Curso 19, Lecciones 37 y 38)
(26, 37, '2023-10-14'),
(26, 38, '2023-10-16'),
-- Progreso para la inscripción con id_inscripcion = 27 (Curso 3, Lecciones 5 y 6)
(27, 5, '2023-10-15'),
(27, 6, '2023-10-17'),
-- Progreso para la inscripción con id_inscripcion = 28 (Curso 5, Lecciones 9 y 10)
(28, 9, '2023-10-16'),
```

```
(28, 10, '2023-10-18');
```

```
-- Población de la tabla Recursos (Corregido)
```

```
INSERT INTO Recursos (id_leccion, nombre_recurso, tipo_recurso, url_recurso) VALUES  
(1, 'PDF de Introducción al Bajo', 'PDF', 'http://ejemplo.com/recurso1.pdf'),  
(3, 'Audio de Rudimentos', 'Audio', 'http://ejemplo.com/recurso2.mp3'),  
(5, 'Video de Acordes Básicos', 'Video', 'http://ejemplo.com/recurso3.mp4'),  
(7, 'PDF de Ejercicios de Respiración', 'PDF', 'http://ejemplo.com/recurso4.pdf'),  
(9, 'Audio de Estructura de una Canción', 'Audio', 'http://ejemplo.com/recurso5.mp3'),  
(11, 'Video de Introducción a la Interfaz', 'Video', 'http://ejemplo.com/recurso6.mp4'),  
(13, 'PDF de Escalas y Modos', 'PDF', 'http://ejemplo.com/recurso7.pdf'),  
(15, 'Audio de Configuración del Equipo', 'Audio', 'http://ejemplo.com/recurso8.mp3'),  
(17, 'Video de Conceptos Básicos', 'Video', 'http://ejemplo.com/recurso9.mp4'),  
(19, 'PDF de Ritmos Básicos', 'PDF', 'http://ejemplo.com/recurso10.pdf'),  
(21, 'Audio de Acordes y Arpeggios', 'Audio', 'http://ejemplo.com/recurso11.mp3'),  
(23, 'Video de Técnicas de Mezcla', 'Video', 'http://ejemplo.com/recurso12.mp4'),  
(25, 'PDF de Armonía Diatónica', 'PDF', 'http://ejemplo.com/recurso13.pdf'),  
(27, 'Audio de Síntesis Sustractiva', 'Audio', 'http://ejemplo.com/recurso14.mp3'),  
(29, 'Video de Técnicas de Interpretación', 'Video', 'http://ejemplo.com/recurso15.mp4'),  
(31, 'PDF de Escritura para Cuerdas', 'PDF', 'http://ejemplo.com/recurso16.pdf'),  
(33, 'Audio de Escalas de Blues', 'Audio', 'http://ejemplo.com/recurso17.mp3'),  
(35, 'Video de Efectos y Procesamiento', 'Video', 'http://ejemplo.com/recurso18.mp4'),  
(37, 'Audio de Progresiones de Acordes', 'Audio', 'http://ejemplo.com/recurso19.mp3'),  
(39, 'Video de Técnicas de Microfoneo', 'Video', 'http://ejemplo.com/recurso20.mp4');
```

```
-- Población de la tabla Certificados
```

```
INSERT INTO Certificados (id_inscripcion, fecha_emision, url_certificado) VALUES  
(1, '2023-10-15', 'http://ejemplo.com/certificado1.pdf'),  
(2, '2023-09-30', 'http://ejemplo.com/certificado2.pdf'),  
(3, '2023-09-05', 'http://ejemplo.com/certificado3.pdf'),  
(4, '2023-09-25', 'http://ejemplo.com/certificado4.pdf'),  
(5, '2023-10-01', 'http://ejemplo.com/certificado5.pdf'),  
(6, '2023-10-10', 'http://ejemplo.com/certificado6.pdf'),  
(7, '2023-10-15', 'http://ejemplo.com/certificado7.pdf'),  
(8, '2023-10-20', 'http://ejemplo.com/certificado8.pdf'),  
(9, '2023-10-25', 'http://ejemplo.com/certificado9.pdf'),
```

```
(10, '2023-10-30', 'http://ejemplo.com/certificado10.pdf'),
(11, '2023-10-18', 'http://ejemplo.com/certificado11.pdf'),
(12, '2023-10-19', 'http://ejemplo.com/certificado12.pdf'),
(13, '2023-10-20', 'http://ejemplo.com/certificado13.pdf'),
(14, '2023-10-21', 'http://ejemplo.com/certificado14.pdf'),
(15, '2023-10-22', 'http://ejemplo.com/certificado15.pdf'),
(16, '2023-10-23', 'http://ejemplo.com/certificado16.pdf'),
(17, '2023-10-24', 'http://ejemplo.com/certificado17.pdf'),
(18, '2023-10-25', 'http://ejemplo.com/certificado18.pdf'),
(19, '2023-10-26', 'http://ejemplo.com/certificado19.pdf'),
(20, '2023-10-27', 'http://ejemplo.com/certificado20.pdf'),
(21, '2023-10-28', 'http://ejemplo.com/certificado21.pdf'),
(22, '2023-10-29', 'http://ejemplo.com/certificado22.pdf'),
(23, '2023-10-30', 'http://ejemplo.com/certificado23.pdf'),
(24, '2023-10-31', 'http://ejemplo.com/certificado24.pdf'),
(25, '2023-11-01', 'http://ejemplo.com/certificado25.pdf'),
(26, '2023-11-02', 'http://ejemplo.com/certificado26.pdf'),
(27, '2023-11-03', 'http://ejemplo.com/certificado27.pdf'),
(28, '2023-11-04', 'http://ejemplo.com/certificado28.pdf');
```

-- Población de la tabla Foros

```
INSERT INTO Foros (id_curso, nombre_foro) VALUES
```

```
(1, 'Foro de Introducción al Bajo'),
(2, 'Foro de Técnicas Avanzadas de Batería'),
(3, 'Foro de Piano para Principiantes'),
(4, 'Foro de Técnica Vocal Nivel 1'),
(5, 'Foro de Composición Musical'),
(6, 'Foro de Producción Musical en DAW'),
(7, 'Foro de Improvisación en Jazz'),
(8, 'Foro de DJing para Principiantes'),
(9, 'Foro de Teoría Musical Completa'),
(10, 'Foro de Batería para Niños'),
(11, 'Foro de Guitarra Acústica Intermedia'),
(12, 'Foro de Mezcla y Masterización'),
(13, 'Foro de Armonía Moderna'),
(14, 'Foro de Sintetizadores y Diseño de Sonido'),
(15, 'Foro de Canto para Musicales');
```



```
(16, 'Foro de Introducción a la Orquestación'),
(17, 'Foro de Guitarra Eléctrica Blues'),
(18, 'Foro de Producción de Música Electrónica'),
(19, 'Foro de Piano Jazz'),
(20, 'Foro de Técnicas de Grabación en Estudio');

-- Población de la tabla Comentarios_Foro
INSERT INTO Comentarios_Foro (id_foro, id_usuario, comentario, fecha_comentario) VALUES
(1, 1, '¡Excelente curso! He aprendido mucho sobre la introducción al bajo.', '2023-10-07'),
(1, 12, 'Tengo una duda sobre la postura correcta. ¿Podrían ayudarme?', '2023-10-08'),
(2, 2, 'Este curso de batería es justo lo que necesitaba. ¡Gracias!', '2023-09-23'),
(2, 14, '¿Alguien sabe cómo mejorar la velocidad en los redobles?', '2023-09-25'),
(3, 3, 'Me encantan las lecciones de piano. Son muy claras y fáciles de seguir.', '2023-08-30'),
(3, 16, '¿Qué tipo de piano recomiendan para principiantes?', '2023-09-02'),
(4, 4, 'Los ejercicios de respiración me han ayudado mucho con mi técnica vocal.', '2023-09-18'),
(4, 18, '¿Cómo puedo aumentar mi rango vocal?', '2023-09-20'),
(5, 5, 'Componer canciones es un desafío, pero este curso lo hace más fácil.', '2023-09-25'),
(5, 20, 'Estoy atascado en la parte del puente de mi canción. ¿Algún consejo?', '2023-09-28'),
(6, 6, 'El curso de producción musical es muy completo. Lo recomiendo.', '2023-10-02'),
(6, 11, '¿Qué DAW recomiendan para empezar?', '2023-10-05'),
(7, 7, 'La improvisación en jazz es un arte. Gracias por las lecciones.', '2023-10-08'),
(7, 13, '¿Cómo puedo mejorar mi fraseo en jazz?', '2023-10-10'),
(8, 8, 'Ser DJ es mi sueño, y este curso me está ayudando a lograrlo.', '2023-10-10'),
(8, 15, '¿Qué equipo es esencial para un DJ principiante?', '2023-10-12'),
(9, 9, 'La teoría musical es fundamental. Este curso es excelente.', '2023-10-12'),
(9, 17, 'Tengo problemas para entender los modos griegos. ¿Alguien me explica?', '2023-10-15'),
(10, 10, 'Mi hijo está disfrutando mucho las clases de batería para niños.', '2023-10-18'),
(10, 19, '¿A qué edad pueden empezar los niños a tocar la batería?', '2023-10-20'),
(11, 1, 'Me ha gustado el curso de guitarra, lo recomiendo', '2023-10-07'),
(11, 3, 'Tengo una duda sobre el cambio de acordes', '2023-10-08'),
(12, 2, 'Excelente curso para mejorar mis producciones!', '2023-09-23'),
(12, 4, '¿Hay alguna forma de mejorar el kick?', '2023-09-25'),
(13, 5, 'Muy interesante el curso, aprendí mucho sobre los acordes', '2023-09-25'),
(13, 7, 'Estoy atascado en la parte de la melodía. ¿Algún tip?', '2023-09-28'),
(14, 8, 'Excelente curso, pude hacer mis primeros sonidos', '2023-10-10'),
(14, 10, '¿Qué especificaciones de hardware recomiendan para sintetizadores?', '2023-10-12'),
(15, 12, 'Me encantó el curso de técnica vocal, quede maravillado', '2023-10-12'),
```

```
(15, 14, 'Tengo problemas con la afinacion. ¿Alguien me ayuda?', '2023-10-15'),
(16, 15, 'Muy buen curso de orquestacion, lo recomiendo', '2023-10-18'),
(16, 17, '¿A que se refieren cuando hablan de orquestacion?', '2023-10-20'),
(17, 18, 'Me gusta el blues y gracias a este curso mejore mis habilidades', '2023-10-07'),
(17, 20, 'Tengo una duda sobre como tocar blues en guitarra electrica', '2023-10-08'),
(18, 1, 'Pude mejorar mi produccion musical con este curso!', '2023-09-23'),
(18, 3, '¿En este curso enseñan que son los beats?', '2023-09-25'),
(19, 5, 'Aprendi mucho sobre los acordes en piano, muy agradecido', '2023-09-25'),
(19, 7, 'Estoy practicando la digitacion en el piano', '2023-09-28'),
(20, 9, 'El curso de grabacion me ayudo a mejorar mi sonido', '2023-10-12'),
(20, 12, '¿Me pueden decir que es el sonido?', '2023-10-15');
```

-- Población de la tabla Descuentos

```
INSERT INTO Descuentos (codigo_descuento, porcentaje_descuento, fecha_inicio, fecha_fin) VALUES
('DESC10', 10.00, '2023-10-01', '2023-10-31'),
('DESC20', 20.00, '2023-11-01', '2023-11-30'),
('DESC50', 50.00, '2023-12-24', '2023-12-25'),
('BLACKFRIDAY', 60.00, '2023-11-24', '2023-11-24'),
('NAVIDAD25', 25.00, '2023-12-01', '2023-12-31');
```

-- Población de la tabla Descuento\_Curso

```
INSERT INTO Descuento_Curso (id_descuento, id_curso) VALUES
(1, 1),
(1, 3),
(1, 5),
(2, 2),
(2, 4),
(2, 6),
(3, 8),
(3, 10),
(3, 12),
(4, 7),
(4, 9),
(4, 11),
(5, 13),
(5, 14),
(5, 15),
```

```
(5, 16),  
(5, 17),  
(5, 18),  
(5, 19),  
(5, 20);
```

```
-- Población de la tabla Países
```

```
INSERT INTO Países (nombre_pais) VALUES
```

```
('Argentina'),  
( 'España'),  
( 'México'),  
( 'Colombia'),  
( 'Estados Unidos'),  
( 'Otro');
```

```
-- Población de la tabla Ciudades
```

```
INSERT INTO Ciudades (id_pais, nombre_ciudad) VALUES
```

```
(1, 'Buenos Aires'),  
(1, 'Córdoba'),  
(2, 'Madrid'),  
(2, 'Barcelona'),  
(3, 'Ciudad de México'),  
(3, 'Guadalajara'),  
(4, 'Bogotá'),  
(4, 'Medellín'),  
(5, 'Nueva York'),  
(5, 'Los Ángeles'),  
(6, 'Otra');
```

## 9. Herramientas y tecnologías usadas

MySQL (Motor de bases de datos version: latest)

MySQL Workbench (Interfaz grafica)

Dbdiagram (generar Diagrama E-R)

# 10. Ejecución de scripts en ambiente local

## 10.1. Creación de Tablas

The screenshot displays a SQL IDE interface with a Navigator on the left and a main editor on the right. The Navigator shows a tree structure of schemas, with 'corcheaacademydb\_dev' selected. The main editor shows a SQL script with the following content:

```
-- Base de datos para Corchea Academy
1 CREATE DATABASE CorcheaAcademyDB_DEV;
2 USE CorcheaAcademyDB_DEV;
3
4
5 -- Tabla Usuarios
6 CREATE TABLE Usuarios (
7     id_usuario INT AUTO_INCREMENT PRIMARY KEY,
8     nombre VARCHAR(100) NOT NULL,
9     apellido VARCHAR(100) NOT NULL,
```

The Output pane at the bottom shows the execution results of the script. The table below summarizes the data from the 'Action Output' section:

#	Time	Action	Message	Duration / Fetch
1	12:08:08	DROP DATABASE 'corcheaacademydb_dev'	7 row(s) affected	0.312 sec
2	12:09:00	CREATE DATABASE CorcheaAcademyDB_DEV	1 row(s) affected	0.015 sec
3	12:09:00	USE CorcheaAcademyDB_DEV	0 row(s) affected	0.000 sec
4	12:09:00	CREATE TABLE Usuarios ( id_usuario INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(100) NOT NULL, apellido VAR...	0 row(s) affected	0.079 sec
5	12:09:00	CREATE TABLE Roles ( id_rol INT AUTO_INCREMENT PRIMARY KEY, nombre_rol VARCHAR(50) NOT NULL )	0 row(s) affected	0.046 sec
6	12:09:00	CREATE TABLE Categorías ( id_categoria INT AUTO_INCREMENT PRIMARY KEY, nombre_categoria VARCHAR(50) NOT NULL )	0 row(s) affected	0.032 sec
7	12:09:00	CREATE TABLE Niveles ( id_nivel INT AUTO_INCREMENT PRIMARY KEY, nombre_nivel VARCHAR(50) NOT NULL )	0 row(s) affected	0.032 sec
8	12:09:00	CREATE TABLE Cursos ( id_curso INT AUTO_INCREMENT PRIMARY KEY, nombre_curso VARCHAR(100) NOT NULL, descripcio...	0 row(s) affected	0.062 sec
9	12:09:00	CREATE TABLE Inscripciones ( id_inscripcion INT AUTO_INCREMENT PRIMARY KEY, id_usuario INT, id_curso INT, fecha_insc...	0 row(s) affected	0.047 sec
10	12:09:00	CREATE TABLE Pagos ( id_pago INT AUTO_INCREMENT PRIMARY KEY, id_inscripcion INT, monto DECIMAL(10, 2) NOT NULL, ...	0 row(s) affected	0.062 sec
11	12:09:00	CREATE TABLE Reseñas ( id_resena INT AUTO_INCREMENT PRIMARY KEY, id_inscripcion INT, calificacion INT CHECK (califica...	0 row(s) affected	0.047 sec
12	12:09:00	CREATE TABLE Instructores ( id_instructor INT AUTO_INCREMENT PRIMARY KEY, nombre VARCHAR(100) NOT NULL, apellido ...	0 row(s) affected	0.063 sec
13	12:09:00	CREATE TABLE Curso_Instructor ( id_curso_instructor INT AUTO_INCREMENT PRIMARY KEY, id_curso INT, id_instructor INT, ...	0 row(s) affected	0.062 sec
14	12:09:01	CREATE TABLE Lecciones ( id_leccion INT AUTO_INCREMENT PRIMARY KEY, id_curso INT, titulo_leccion VARCHAR(100) NOT...	0 row(s) affected	0.078 sec
15	12:09:01	CREATE TABLE Progreso_Leccion ( id_progreso INT AUTO_INCREMENT PRIMARY KEY, id_inscripcion INT, id_leccion INT, fe...	0 row(s) affected	0.047 sec
16	12:09:01	CREATE TABLE Recursos ( id_recurso INT AUTO_INCREMENT PRIMARY KEY, id_leccion INT, nombre_recurso VARCHAR(100) ...	0 row(s) affected	0.063 sec
17	12:09:01	CREATE TABLE Certificados ( id_certificado INT AUTO_INCREMENT PRIMARY KEY, id_inscripcion INT, fecha_emision DATETIM...	0 row(s) affected	0.047 sec
18	12:09:01	CREATE TABLE Foros ( id_foro INT AUTO_INCREMENT PRIMARY KEY, id_curso INT, nombre_foro VARCHAR(100) NOT NULL, ...	0 row(s) affected	0.140 sec
19	12:09:01	CREATE TABLE Comentarios_Foro ( id_comentario INT AUTO_INCREMENT PRIMARY KEY, id_foro INT, id_usuario INT, coment...	0 row(s) affected	0.125 sec
20	12:09:01	CREATE TABLE Descuentos ( id_descuento INT AUTO_INCREMENT PRIMARY KEY, codigo_descuento VARCHAR(50) UNIQUE N...	0 row(s) affected	0.141 sec
21	12:09:01	CREATE TABLE Descuento_Curso ( id_descuento_curso INT AUTO_INCREMENT PRIMARY KEY, id_descuento INT, id_curso IN...	0 row(s) affected	0.094 sec
22	12:09:01	CREATE TABLE Paises ( id_pais INT AUTO_INCREMENT PRIMARY KEY, nombre_pais VARCHAR(100) NOT NULL )	0 row(s) affected	0.031 sec
23	12:09:01	CREATE TABLE Ciudades ( id_ciudad INT AUTO_INCREMENT PRIMARY KEY, id_pais INT, nombre_ciudad VARCHAR(100) NOT ...	0 row(s) affected	0.062 sec
24	12:09:01	ALTER TABLE Usuarios ADD CONSTRAINT fk_usuario_rol FOREIGN KEY (id_rol) REFERENCES Roles(id_rol)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.110 sec
25	12:09:01	CREATE INDEX idx_email ON Usuarios(email)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.047 sec
26	12:09:01	CREATE INDEX idx_nombre_curso ON Cursos(nombre_curso)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.047 sec
27	12:09:02	CREATE INDEX idx_fecha_inscripcion ON Inscripciones(fecha_inscripcion)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.046 sec

## 10.2. Población de tablas

**Navigation**

**SCHEMAS**

Filter objects

- coderrhouse\_games
- corcheaacademydb
- corcheaacademydb\_dev**
  - Tables
    - categorias
    - certificados
    - ciudades
    - comentarios\_foro
    - curso\_instructor
    - cursos
    - descuento\_curso
    - descuentos
    - foros
    - inscripciones
    - instructores
    - lecciones
    - niveles
    - pagos
    - paises
    - progreso\_leccion
    - recursos
    - reseñas
    - roles
    - usuarios
  - Views
  - Stored Procedures
  - Functions
- sys
- world

SQL File 7
SQL File 7

Limit to 1000 rows

```

1 USE CorcheaAcademyDB_dev;
2
3 -- Población de la tabla Roles
4 INSERT INTO Roles (nombre_rol) VALUES
5 ('Administrador'),
6 ('Instructor'),
7 ('Estudiante'),
8 ('Invitado'),
9 ('Editor'),
10 ('Moderador'),
11 ('Coordinador'),
12 ('Soporte Técnico'),

```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	13:23:53	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	13:23:53	INSERT INTO Roles (nombre_rol) VALUES (Administrador), (Instructor), (Estudiante), (Invitado), (Editor), (Moderador), (Coordinador), (Soporte	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0	0.000 sec
3	13:23:53	INSERT INTO Categorias (nombre_categoria) VALUES (Bajo), (Batería), (Piano), (Técnica Vocal), (Guitarra), (Producción Musical), (Composici...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0	0.000 sec
4	13:23:53	INSERT INTO Niveles (nombre_nivel) VALUES (Principiante), (Intermedio), (Avanzado)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
5	13:23:53	INSERT INTO Paises (nombre_pais) VALUES (Argentina), (España), (México), (Colombia), (Estados Unidos), (Otro)	6 row(s) affected Records: 6 Duplicates: 0 Warnings: 0	0.015 sec
6	13:23:53	INSERT INTO Ciudades (id_pais, nombre_ciudad) VALUES (1, 'Buenos Aires'), (1, 'Córdoba'), (2, 'Madrid'), (2, 'Barcelona'), (3, 'Ciudad de México'), (...	11 row(s) affected Records: 11 Duplicates: 0 Warnings: 0	0.000 sec
7	13:23:53	INSERT INTO Usuarios (nombre, apellido, email, contraseña, fecha_registro, id_rol) VALUES (Carlos), (Pérez), 'carlos.perez@gmail.com', 'pass1234'...	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
8	13:23:53	INSERT INTO Cursos (nombre_curso, descripcion, precio, fecha_creacion, id_categoria, id_nivel) VALUES (Aprende Bajo Básico), (Curso introduct...	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
9	13:23:53	INSERT INTO Inscripciones (id_usuario, id_curso, fecha_inscripcion) VALUES (1, 1, '2023-10-05'), (2, 2, '2023-09-20'), (3, 3, '2023-08-25'), (4, 4, '2...	28 row(s) affected Records: 28 Duplicates: 0 Warnings: 0	0.000 sec
10	13:23:53	INSERT INTO Pagos (id_inscripcion, monto, fecha_pago, metodo_pago) VALUES (1, 49.99, '2023-10-06', 'Tarjeta de Crédito'), (2, 79.99, '2023-09-...	28 row(s) affected Records: 28 Duplicates: 0 Warnings: 0	0.000 sec
11	13:23:53	INSERT INTO Instructores (nombre, apellido, especialidad, email) VALUES (Pedro), (Sánchez), (Bajo Eléctrico), 'pedro.sanchez.instructor@gmail.co...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0	0.000 sec
12	13:23:53	INSERT INTO Curso_Instructor (id_curso, id_instructor) VALUES (1, 1), (2, 2), (3, 5), (4, 3), (5, 7), (6, 6), (7, 10), (8, 8), (9, 9), (10, 2), (11, 4), (12, 6), (...	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
13	13:23:53	INSERT INTO Lecciones (id_curso, titulo_leccion, contenido_leccion, orden) VALUES -- Lecciones para el curso con id_curso = 1 (1, 'Introducción...	40 row(s) affected Records: 40 Duplicates: 0 Warnings: 0	0.000 sec
14	13:23:53	INSERT INTO Progreso_Leccion (id_inscripcion, id_leccion, fecha_completado) VALUES -- Progreso para la inscripción con id_inscripcion = 1 (Cur...	56 row(s) affected Records: 56 Duplicates: 0 Warnings: 0	0.000 sec
15	13:23:53	INSERT INTO Recursos (id_leccion, nombre_recurso, tipo_recurso, url_recurso) VALUES (1, 'PDF de Introducción al Bajo', 'PDF', 'http://ejemplo.c...	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
16	13:23:53	INSERT INTO Certificados (id_inscripcion, fecha_emision, url_certificado) VALUES (1, '2023-10-15', 'http://ejemplo.com/certificado1.pdf'), (2, '2023...	28 row(s) affected Records: 28 Duplicates: 0 Warnings: 0	0.015 sec
17	13:23:53	INSERT INTO Foros (id_curso, nombre_foro) VALUES (1, 'Foro de Introducción al Bajo'), (2, 'Foro de Técnicas Avanzadas de Batería'), (3, 'Foro de...	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
18	13:23:53	INSERT INTO Comentarios_Foro (id_foro, id_usuario, comentario, fecha_comentario) VALUES (1, 1, '¡Excelente curso! He aprendido mucho sobre l...	40 row(s) affected Records: 40 Duplicates: 0 Warnings: 0	0.000 sec
19	13:23:53	INSERT INTO Descuentos (codigo_descuento, porcentaje_descuento, fecha_inicio, fecha_fin) VALUES (DESC10, 10.00, '2023-10-01', '2023-10-...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.000 sec
20	13:23:53	INSERT INTO Descuento_Curso (id_descuento, id_curso) VALUES (1, 1), (1, 3), (1, 5), (2, 2), (2, 4), (2, 6), (3, 8), (3, 10), (3, 12), (4, 7), (4, 9), (4, 11)...	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
21	13:23:53	INSERT INTO Paises (nombre_pais) VALUES (Argentina), (España), (México), (Colombia), (Estados Unidos), (Otro)	6 row(s) affected Records: 6 Duplicates: 0 Warnings: 0	0.000 sec
22	13:23:53	INSERT INTO Ciudades (id_pais, nombre_ciudad) VALUES (1, 'Buenos Aires'), (1, 'Córdoba'), (2, 'Madrid'), (2, 'Barcelona'), (3, 'Ciudad de México'), (...	11 row(s) affected Records: 11 Duplicates: 0 Warnings: 0	0.000 sec

## 10.3. Creación de Funcion1

The screenshot displays a SQL IDE interface with the following components:

- Navigator:** Shows the database schema structure. The 'corcheaacademydb\_dev' database is selected, showing various tables and functions. The function 'f() Obtener\_Cantidad\_Inscripciones' is highlighted.
- SQL Editor:** Contains the SQL script for creating the function. The script is as follows:

```
1 USE CorcheaAcademyDB_dev;
2
3 DELIMITER $$
4
5 DROP FUNCTION IF EXISTS Obtener_Cantidad_Inscripciones$$
6
7 CREATE FUNCTION Obtener_Cantidad_Inscripciones(id_usuario_input INT)
8 RETURNS INT
9 DETERMINISTIC
10 BEGIN
11     DECLARE cantidad INT;
12     SELECT COUNT(*) INTO cantidad
13     FROM Inscripciones
14     WHERE id_usuario = id_usuario_input;
15     RETURN cantidad;
16 END$$
17
```
- Result Grid:** Shows the execution results of the SQL script. The first result is 'Total\_Inscripciones' with a value of 2.
- Output:** Displays the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
1	14:00:21	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	14:00:21	DROP FUNCTION IF EXISTS Obtener_Cantidad_Inscripciones	0 row(s) affected, 1 warning(s): 1305 FUNCTION corcheaacademydb_dev.Obtener_Cantidad_Inscripciones does not exist	0.016 sec
3	14:00:21	CREATE FUNCTION Obtener_Cantidad_Inscripciones(id_usuario_input INT) RETURNS INT DETERMINISTIC BEGIN DECLARE cantidad INT; SELECT COU...	0 row(s) affected	0.000 sec
4	14:00:21	- Ejemplo de uso: SELECT Obtener_Cantidad_Inscripciones(1) AS Total_Inscripciones LIMIT 0, 1000	1 row(s) returned	0.016 sec / 0.000 sec

## 10.4. Creación Función 2

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- coderhouse\_gamers
- corcheaacademydb
- corcheaacademydb\_dev
  - Tables
    - categorias
    - certificados
    - ciudades
    - comentarios\_foro
    - curso\_instructor
    - cursos
    - descuento\_curso
    - descuentos
    - foros
    - inscripciones
    - instructores
    - lecciones
    - niveles
    - pagos
    - paises
    - progreso\_leccion
    - recursos
    - reseñas
    - roles
    - usuarios
  - Views
  - Stored Procedures
  - Functions
    - f() Calcular\_Total\_Pagos\_Usuario
    - f() Obtener\_Cantidad\_Inscripciones
- sys
- world

SQL File 6\* SQL File 7\* SQL File 11\* SQL File 13\* x

Limit to 1000 rows

```
1 USE CorcheaAcademyDB_dev;
2
3 DELIMITER $$
4
5 DROP FUNCTION IF EXISTS Calcular_Total_Pagos_Usuario$$
6
7 CREATE FUNCTION Calcular_Total_Pagos_Usuario(id_usuario_input INT)
8 RETURNS DECIMAL(10, 2)
9 DETERMINISTIC
10 BEGIN
11     DECLARE total DECIMAL(10, 2);
12
13     -- Se suma el monto de las inscripciones del usuario
14     SELECT SUM(p.monto) INTO total
15     FROM Pagos p
16     JOIN Inscripciones i ON p.id_inscripcion = i.id_inscripcion
17     WHERE i.id_usuario = id_usuario_input;
```

Result Grid

Filter Rows: | Export: | Wrap Cell Contents: |

Total_Pagado
119.98

Result 1 x

Read Only

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	14:06:02	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	14:06:02	DROP FUNCTION IF EXISTS Calcular_Total_Pagos_Usuario	0 row(s) affected, 1 warning(s): 1305 FUNCTION corcheaacademydb_dev.Calcular_Total_Pagos_Usuario does not exist	0.000 sec
3	14:06:02	CREATE FUNCTION Calcular_Total_Pagos_Usuario(id_usuario_input INT) RETURNS DECIMAL(10, 2) DETERMINISTIC BEGIN DECLARE total DECIMAL(10, 2)...	0 row(s) affected	0.016 sec
4	14:06:02	-- Ejemplo de uso: SELECT Calcular_Total_Pagos_Usuario(1) AS Total_Pagado LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

Administration Schemas

Information

No object selected

Object Info Session



## 10.5. Creación Vista 1

The screenshot displays the SQL Server Enterprise Manager interface. The Navigator pane on the left shows the database structure, with the 'corcheaacademydb\_dev' database selected. The central pane shows the following SQL script:

```
1 • USE CorcheaAcademyDB_dev;
2
3 • DROP VIEW IF EXISTS Vista_Detalle_Cursos;
4
5 • CREATE VIEW Vista_Detalle_Cursos AS
6 SELECT
7     c.id_curso AS ID_Curso,
8     c.nombre_curso AS Nombre_Curso,
9     c.descripcion AS Descripcion,
10    c.precio AS Precio,
11    cat.nombre_categoria AS Categoria,
12    n.nombre_nivel AS Nivel
13 FROM
14     Cursos c
15 JOIN Categorias cat ON c.id_categoria = cat.id_categoria
16 JOIN Niveles n ON c.id_nivel = n.id_nivel;
17
18 -- Consulta de la vista
19 -- SELECT * FROM Vista_Detalle_Cursos;
20
21 /*
22 Descripción:
23 Esta vista muestra información detallada de los cursos, incluyendo el nombre del curso, descripción, precio, categoría y nivel.
24 Objetivo:
25 Facilitar la consulta rápida de información completa de los cursos, combinando datos de las tablas Cursos, Categorías y Niveles.
26
27 Tablas involucradas:
```

The Output pane at the bottom shows the execution results:

#	Time	Action	Message	Duration / Fetch
✓ 1	14:19:34	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
⚠ 2	14:19:34	DROP VIEW IF EXISTS Vista_Detalle_Cursos	0 row(s) affected, 1 warning(s): 1051 Unknown table 'corcheaacademydb_dev.vista_detalle_cursos'	0.015 sec
✓ 3	14:19:34	CREATE VIEW Vista_Detalle_Cursos AS SELECT c.id_curso AS ID_Curso, c.nombre_curso AS Nombre_Curso, c.descripcion AS Descripci...	0 row(s) affected	0.016 sec

Schema: corcheaacademydb\_dev

## 10.6. Creación Vista 2

The screenshot displays the SQL Developer environment. On the left, the Schema Navigator shows the database structure for 'corcheaacademydb\_dev', including tables like 'categorias', 'certificados', 'ciudades', and 'instructores'. The main SQL Editor window contains the following SQL script:

```
1 • USE CorcheaAcademyDB_dev;
2
3 • DROP VIEW IF EXISTS Vista_Instructores_Cursos;
4
5 • CREATE VIEW Vista_Instructores_Cursos AS
6   SELECT
7     i.id_instructor AS ID_Instructor,
8     i.nombre AS Nombre_Instructor,
9     i.apellido AS Apellido_Instructor,
10    c.id_curso AS ID_Curso,
11    c.nombre_curso AS Nombre_Curso
12  FROM
13    Instructores i
14  JOIN Curso_Instructor ci ON i.id_instructor = ci.id_instructor
15  JOIN Cursos c ON ci.id_curso = c.id_curso;
16
17  -- Consulta de la vista
18  -- SELECT * FROM Vista_Instructores_Cursos;
19
20  /*
21   Descripción:
22   Esta vista muestra la relación entre instructores y los cursos que imparten, incluyendo el ID y nombre del instructor, y el ID y nombre del curso.
23   Objetivo:
24   Facilitar la consulta de qué instructor imparte cada curso, combinando datos de las tablas Instructores, Curso_Instructor y Cursos.
25
26   Tablas involucradas:
27   - Instructores
```

At the bottom, the Output window shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	14:20:32	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	14:20:32	DROP VIEW IF EXISTS Vista_Instructores_Cursos	0 row(s) affected, 1 warning(s): 1051 Unknown table 'corcheaacademydb_dev.vista_instructores_cursos'	0.000 sec
3	14:20:32	CREATE VIEW Vista_Instructores_Cursos AS SELECT i.id_instructor AS ID_Instructor, i.nombre AS Nombre_Instructor, i.apellido AS Apellid...	0 row(s) affected	0.000 sec

The schema at the bottom is set to 'corcheaacademydb\_dev'.

## 10.7. Creación Vista 3

The screenshot displays the SQL Server Enterprise Manager interface. The Navigator pane on the left shows the database structure, with the 'corcheaacademydb\_dev' schema selected. The central SQL editor contains the following SQL script:

```
1 • USE CorcheaAcademyDB_dev;
2
3 • DROP VIEW IF EXISTS Vista_Resumen_Pagos;
4
5 • CREATE VIEW Vista_Resumen_Pagos AS
6   SELECT
7     p.id_pago AS ID_Pago,
8     u.id_usuario AS ID_Usuario,
9     u.nombre AS Nombre_Usuario,
10    u.apellido AS Apellido_Usuario,
11    c.id_curso AS ID_Curso,
12    c.nombre_curso AS Nombre_Curso,
13    p.monto AS Monto_Pagado,
14    p.fecha_pago AS Fecha_Pago,
15    p.metodo_pago AS Metodo_Pago
16  FROM
17    Pagos p
18  JOIN Inscripciones i ON p.id_inscripcion = i.id_inscripcion
19  JOIN Usuarios u ON i.id_usuario = u.id_usuario
20  JOIN Cursos c ON i.id_curso = c.id_curso;
21
22 -- Consulta de la vista
23 -- SELECT * FROM Vista_Resumen_Pagos;
24
25 /*
26 Descripción:
27 Esta vista muestra un resumen de los pagos realizados, incluyendo información del usuario, el curso al que corresponde el pago, el monto, la fecha y el método de pago.
```

The Output pane at the bottom shows the execution results of the SQL script:

#	Time	Action	Message	Duration / Fetch
1	14:21:37	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	14:21:37	DROP VIEW IF EXISTS Vista_Resumen_Pagos	0 row(s) affected, 1 warning(s): 1051 Unknown table 'corcheaacademydb_dev.vista_resumen_pagos'	0.000 sec
3	14:21:37	CREATE VIEW Vista_Resumen_Pagos AS SELECT p.id_pago AS ID_Pago, u.id_usuario AS ID_Usuario, u.nombre AS Nombre_Usuario, ...	0 row(s) affected	0.000 sec

Schema: corcheaacademydb\_dev

## 10.8. Creación Vista 4

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'Navigator' pane shows the 'corcheaacademydb\_dev' database with its tables and views. The central pane contains the following SQL script:

```
3 • DROP VIEW IF EXISTS Vista_Lecciones_Por_Curso;
4
5 • CREATE VIEW Vista_Lecciones_Por_Curso AS
6 SELECT
7     c.id_curso AS ID_Curso,
8     c.nombre_curso AS Nombre_Curso,
9     l.id_leccion AS ID_Leccion,
10    l.titulo_leccion AS Titulo_Leccion,
11    l.orden AS Orden_Leccion
12 FROM
13     Cursos c
14 JOIN Lecciones l ON c.id_curso = l.id_curso;
15
16 -- Consulta de la vista
17 -- SELECT * FROM Vista_Lecciones_Por_Curso;
18
19 /*
20 Descripción:
21 Esta vista muestra las lecciones asociadas a cada curso, incluyendo el ID y nombre del curso, y el ID, título y orden de la lección.
22 Objetivo:
23 Facilitar la consulta de las lecciones que componen cada curso, combinando datos de las tablas Cursos y Lecciones.
24
25 Tablas involucradas:
26 - Cursos
27 - Lecciones
28 */
```

At the bottom, the 'Output' pane shows the 'Action Output' table with the following data:

#	Time	Action	Message	Duration / Fetch
1	14:22:24	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	14:22:24	DROP VIEW IF EXISTS Vista_Lecciones_Por_Curso	0 row(s) affected, 1 warning(s): 1051 Unknown table 'corcheaacademydb_dev.vista_lecciones_por_curso'	0.000 sec
3	14:22:24	CREATE VIEW Vista_Lecciones_Por_Curso AS SELECT c.id_curso AS ID_Curso, c.nombre_curso AS Nombre_Curso, l.id_leccion AS ID_Leccion...	0 row(s) affected	0.016 sec

## 10.9. Creación Vista 5

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane shows the database structure, including the 'corcheaacademydb\_dev' database and its tables. The main pane shows the SQL script for creating the view 'Vista\_Comentarios\_Foro'. The script includes a description, objective, and the list of tables involved. The output pane at the bottom shows the execution of the script, including a warning about an unknown table.

```
11 u.id_usuario AS ID_Usuario,
12 u.nombre AS Nombre_Usuario,
13 u.apellido AS Apellido_Usuario,
14 cf.comentario AS Comentario,
15 cf.fecha_comentario AS Fecha_Comentario
16 FROM
17 Foros f
18 JOIN Comentarios_Foro cf ON f.id_foro = cf.id_foro
19 JOIN Usuarios u ON cf.id_usuario = u.id_usuario
20 JOIN Cursos c ON f.id_curso = c.id_curso;
21
22 -- Consulta de la vista
23 -- SELECT * FROM Vista_Comentarios_Foro;
24
25 /*
26 Descripción:
27 Esta vista muestra los comentarios realizados en los foros, incluyendo el ID del foro, el ID y nombre del curso al que pertenece el foro, el ID del comentario, el ID, nombre, y apellido del usuario que realizó el comentario.
28 Objetivo:
29 Facilitar la consulta de los comentarios en los foros, mostrando información consolidada de las tablas Foros, Comentarios_Foro, Usuarios y Cursos.
30
31 Tablas involucradas:
32 - Foros
33 - Comentarios_Foro
34 - Usuarios
35 - Cursos
36 */
```

Output

#	Time	Action	Message	Duration / Fetch
1	14:23:22	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	14:23:22	DROP VIEW IF EXISTS Vista_Comentarios_Foro	0 row(s) affected, 1 warning(s): 1051 Unknown table 'corcheaacademydb_dev.vista_comentarios_foro'	0.016 sec
3	14:23:22	CREATE VIEW Vista_Comentarios_Foro AS SELECT f.id_foro AS ID_Foro, c.id_curso AS ID_Curso, c.nombre_curso AS Nombre_Curso, cf....	0 row(s) affected	0.015 sec

Schema: corcheaacademydb\_dev

## 10.10. Creación de SP 1

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane shows the database structure, including 'corcheaacademydb' and its 'Stored Procedures' folder. The main editor window shows the following SQL script:

```
1 USE CorcheaAcademyDB_dev;
2
3 DELIMITER $$
4
5 DROP PROCEDURE IF EXISTS SP_Agregar_Comentario_Foro$$
6
7 CREATE PROCEDURE SP_Agregar_Comentario_Foro(
8     IN id_foro_input INT,
9     IN id_usuario_input INT,
10    IN comentario_input TEXT
11)
12 BEGIN
13     -- Validar si el foro existe
14     IF EXISTS (SELECT 1 FROM Foros WHERE id_foro = id_foro_input) THEN
15         -- Validar si el usuario existe
16         IF EXISTS (SELECT 1 FROM Usuarios WHERE id_usuario = id_usuario_input) THEN
17             -- Insertar el comentario en el foro
18             INSERT INTO Comentarios_Foro (id_foro, id_usuario, comentario, fecha_comentario)
19             VALUES (id_foro_input, id_usuario_input, comentario_input, NOW());
20         ELSE
21             SIGNAL SQLSTATE '45000'
22             SET MESSAGE_TEXT = 'El usuario especificado no existe.';
23         END IF;
24     ELSE
25         SIGNAL SQLSTATE '45000'
26         SET MESSAGE_TEXT = 'El foro especificado no existe.';
27     END IF;
28 END IF;
```

Below the script, the 'Output' pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	14:32:48	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	14:32:48	DROP PROCEDURE IF EXISTS SP_Agregar_Comentario_Foro	0 row(s) affected, 1 warning(s): 1305 PROCEDURE corcheaacademydb.SP_Agregar_Comentario_Foro does not exist	0.000 sec
3	14:32:48	CREATE PROCEDURE SP_Agregar_Comentario_Foro( IN id_foro_input INT, IN id_usuario_input INT, IN comentario_input TEXT) BEGIN ...	0 row(s) affected	0.016 sec

The 'Administration' tab is selected, and the 'Schemas' pane shows the database structure. The 'Information' pane at the bottom indicates 'No object selected'.

## 10.11. Creación de Sp2

The screenshot displays the MySQL Workbench interface. On the left, the 'SCHEMAS' pane shows a tree view with databases including 'corcheaacademydb' and 'corcheaacademydb\_dev'. The main editor window shows a SQL script for creating a stored procedure. The script includes comments in Spanish for validation and error handling. The 'Output' pane at the bottom shows the execution results of the script.

```
1 USE CorcheaAcademyDB_dev;
2
3 DELIMITER $$
4
5 DROP PROCEDURE IF EXISTS SP_Actualizar_Nivel_Curso$$
6
7 CREATE PROCEDURE SP_Actualizar_Nivel_Curso(
8     IN id_curso_input INT,
9     IN id_nivel_nuevo INT
10 )
11 BEGIN
12     -- Validar si el curso existe
13     IF EXISTS (SELECT 1 FROM Cursos WHERE id_curso = id_curso_input) THEN
14         -- Validar si el nivel nuevo existe
15         IF EXISTS (SELECT 1 FROM Niveles WHERE id_nivel = id_nivel_nuevo) THEN
16             -- Actualizar el nivel del curso
17             UPDATE Cursos
18             SET id_nivel = id_nivel_nuevo
19             WHERE id_curso = id_curso_input;
20         ELSE
21             SIGNAL SQLSTATE '45000'
22             SET MESSAGE_TEXT = 'El nivel especificado no existe.';
23         END IF;
24     ELSE
25         SIGNAL SQLSTATE '45000'
26         SET MESSAGE_TEXT = 'El curso especificado no existe.';
27     END IF;
28 END IF;
```

**Output**

#	Time	Action	Message	Duration / Fetch
✓ 1	14:35:16	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
⚠ 2	14:35:16	DROP PROCEDURE IF EXISTS SP_Actualizar_Nivel_Curso	0 row(s) affected, 1 warning(s): 1305 PROCEDURE corcheaacademydb_dev.SP_Actualizar_Nivel_Curso does not exist	0.015 sec
✓ 3	14:35:16	CREATE PROCEDURE SP_Actualizar_Nivel_Curso( IN id_curso_input INT, IN id_nivel_nuevo INT ) BEGIN -- Validar si el curso existe IF...	0 row(s) affected	0.000 sec

No object selected

## 10.12. Creación de Sp 3

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane shows the database structure, including the 'corcheaacademydb' database and its 'dev' schema. The main pane shows the SQL script for creating the stored procedure 'SP\_Registrar\_Inscripcion'.

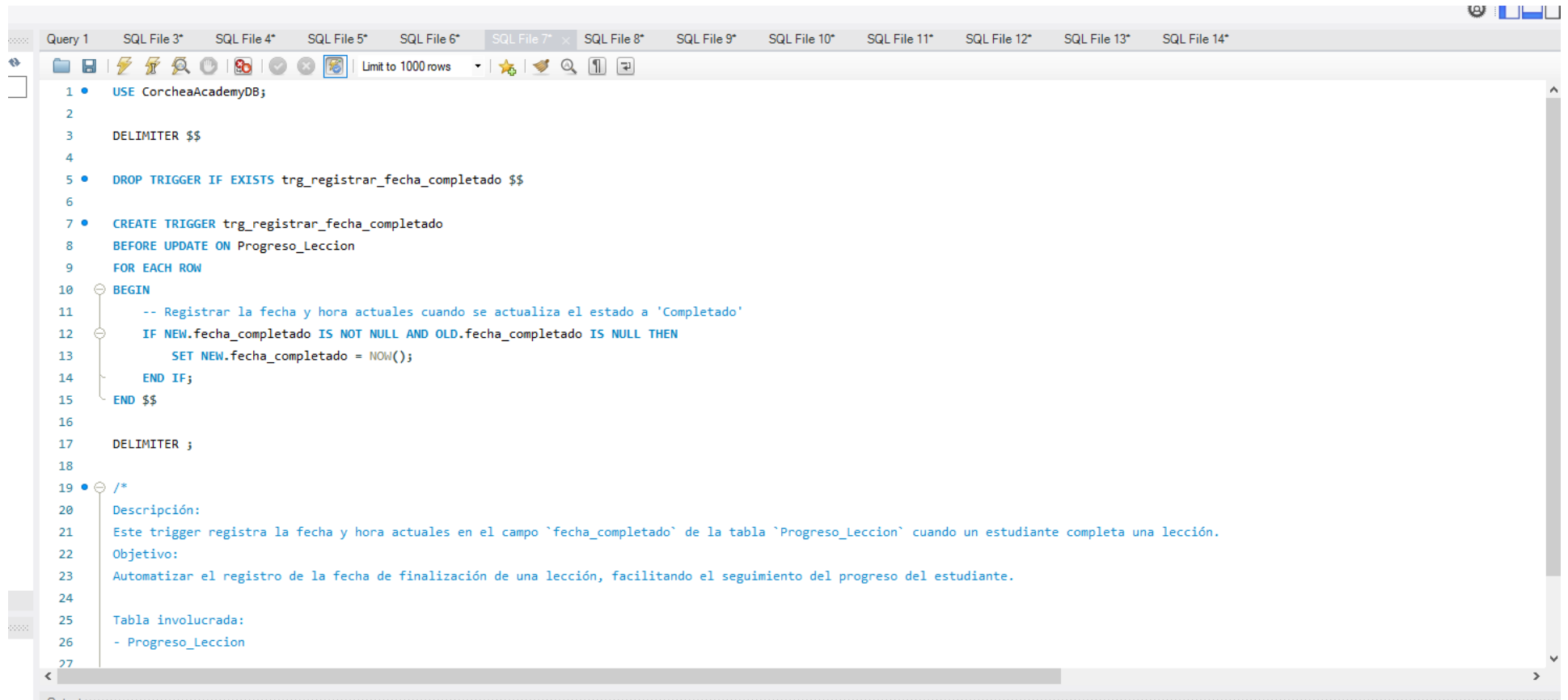
```
1 USE CorcheaAcademyDB_dev;
2
3 DELIMITER $$
4
5 DROP PROCEDURE IF EXISTS SP_Registrar_Inscripcion$$
6
7 CREATE PROCEDURE SP_Registrar_Inscripcion(
8     IN id_usuario_input INT,
9     IN id_curso_input INT
10 )
11 BEGIN
12     -- Validar si el usuario existe y tiene el rol de estudiante
13     IF EXISTS (
14         SELECT 1 FROM Usuarios
15         WHERE id_usuario = id_usuario_input AND id_rol = 3
16     ) THEN
17         -- Validar si el curso existe
18         IF EXISTS (
19             SELECT 1 FROM Cursos WHERE id_curso = id_curso_input
20         ) THEN
21             -- Insertar la inscripción
22             INSERT INTO Inscripciones (id_usuario, id_curso, fecha_inscripcion)
23             VALUES (id_usuario_input, id_curso_input, CURDATE());
24         ELSE
25             SIGNAL SQLSTATE '45000'
26             SET MESSAGE_TEXT = 'El curso especificado no existe.';
27     END IF;
28 END IF;
```

The 'Output' pane at the bottom shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	14:38:28	USE CorcheaAcademyDB_dev	0 row(s) affected	0.000 sec
2	14:38:28	DROP PROCEDURE IF EXISTS SP_Registrar_Inscripcion	0 row(s) affected, 1 warning(s): 1305 PROCEDURE corcheaacademydb_dev.SP_Registrar_Inscripcion does not exist	0.000 sec
3	14:38:28	CREATE PROCEDURE SP_Registrar_Inscripcion( IN id_usuario_input INT, IN id_curso_input INT ) BEGIN -- Validar si el usuario existe y tie...	0 row(s) affected	0.016 sec

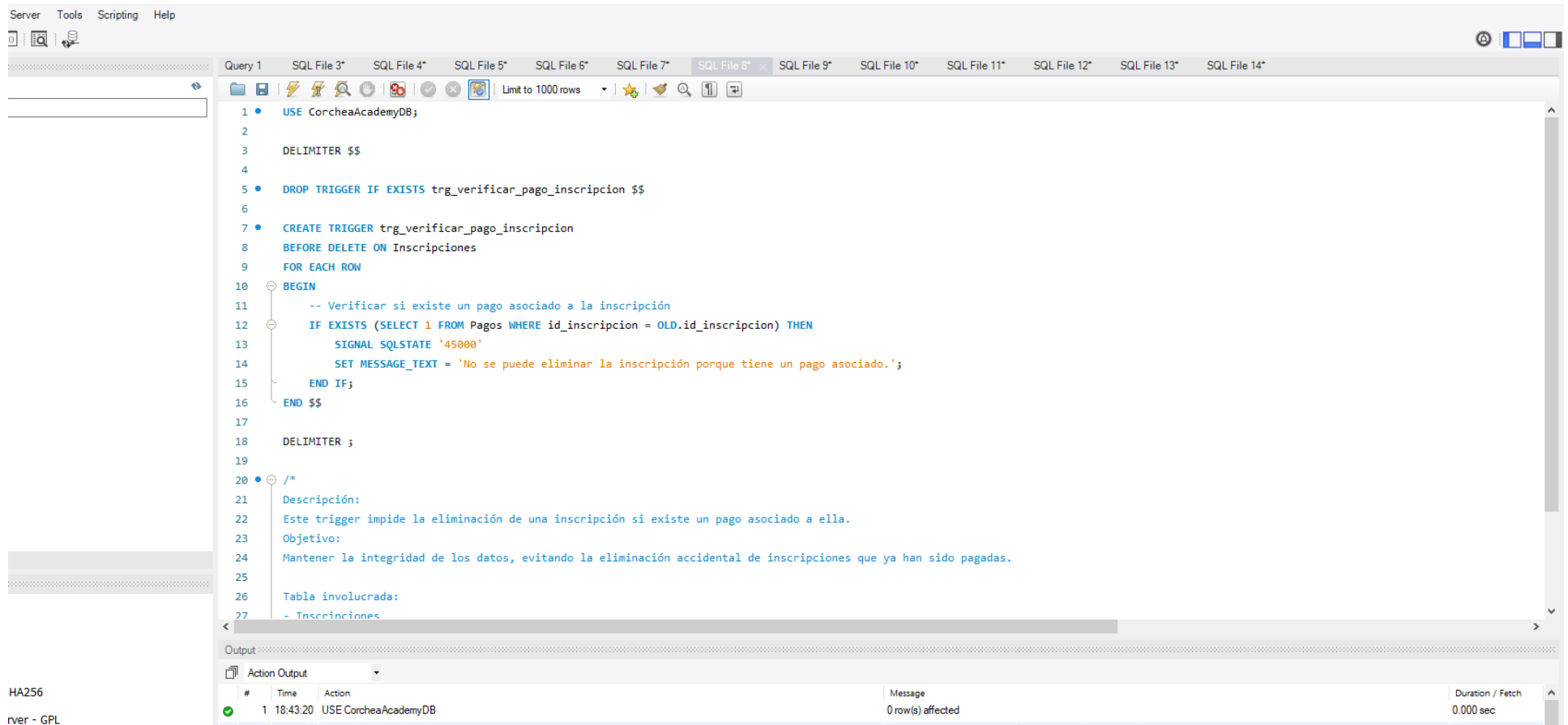


## 10.13 Creación de Trigger 1



```
1 • USE CorcheaAcademyDB;
2
3 DELIMITER $$
4
5 • DROP TRIGGER IF EXISTS trg_registrar_fecha_completado $$
6
7 • CREATE TRIGGER trg_registrar_fecha_completado
8 BEFORE UPDATE ON Progreso_Leccion
9 FOR EACH ROW
10 BEGIN
11     -- Registrar la fecha y hora actuales cuando se actualiza el estado a 'Completado'
12     IF NEW.fecha_completado IS NOT NULL AND OLD.fecha_completado IS NULL THEN
13         SET NEW.fecha_completado = NOW();
14     END IF;
15 END $$
16
17 DELIMITER ;
18
19 • /*
20 Descripción:
21 Este trigger registra la fecha y hora actuales en el campo `fecha_completado` de la tabla `Progreso_Leccion` cuando un estudiante completa una lección.
22 Objetivo:
23 Automatizar el registro de la fecha de finalización de una lección, facilitando el seguimiento del progreso del estudiante.
24
25 Tabla involucrada:
26 - Progreso_Leccion
27
```

## 10.14 Creación de Trigger 2



```
1 • USE CorcheaAcademyDB;
2
3 DELIMITER $$
4
5 • DROP TRIGGER IF EXISTS trg_verificar_pago_inscripcion $$
6
7 • CREATE TRIGGER trg_verificar_pago_inscripcion
8 BEFORE DELETE ON Inscripciones
9 FOR EACH ROW
10 BEGIN
11     -- Verificar si existe un pago asociado a la inscripción
12     IF EXISTS (SELECT 1 FROM Pagos WHERE id_inscripcion = OLD.id_inscripcion) THEN
13         SIGNAL SQLSTATE '45000'
14         SET MESSAGE_TEXT = 'No se puede eliminar la inscripción porque tiene un pago asociado.';
15     END IF;
16 END $$
17
18 DELIMITER ;
19
20 • /*
21 Descripción:
22 Este trigger impide la eliminación de una inscripción si existe un pago asociado a ella.
23 Objetivo:
24 Mantener la integridad de los datos, evitando la eliminación accidental de inscripciones que ya han sido pagadas.
25
26 Tabla involucrada:
27 - Inscripciones
```

Output

#	Time	Action	Message	Duration / Fetch
1	18:43:20	USE CorcheaAcademyDB	0 row(s) affected	0.000 sec

HA256

rver - GPL