



Trabajo Final

Alexandra Arisneidy Sánchez Santana.

Diseñar y desarrollar una base de datos en MySQL que refleje la relación 1:M entre tablas, aplicando claves primarias, foráneas, consultas básicas, joins, estadísticas y manipulación de datos (insertar, actualizar y eliminar).

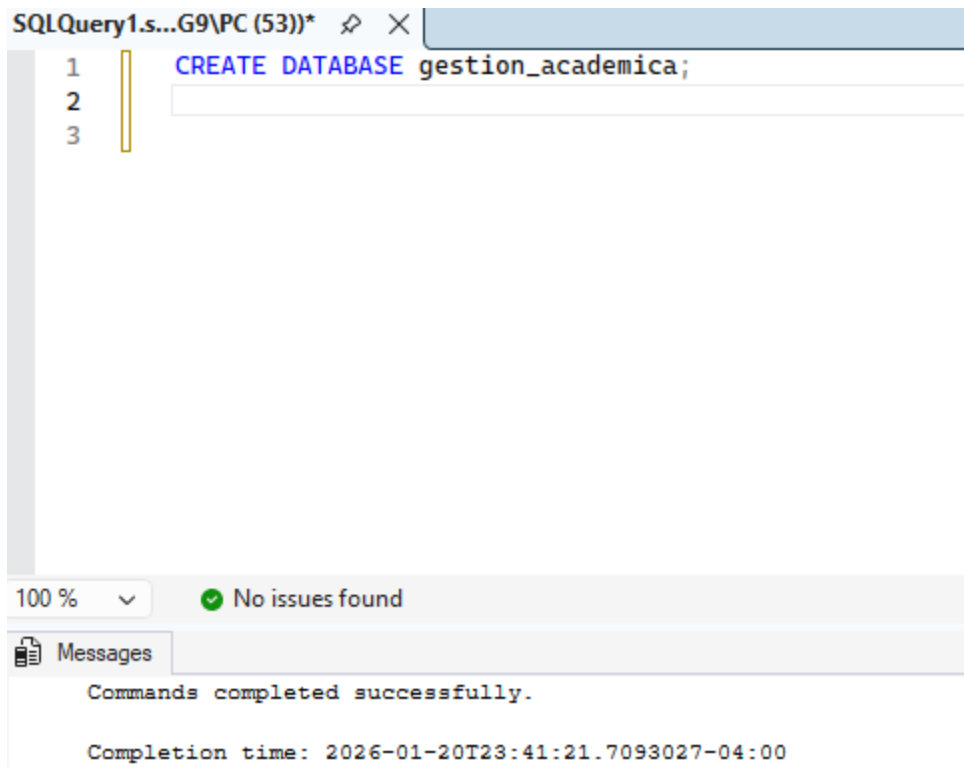
Estructura del proyecto

Tendremos un sistema de gestión académica simple con estas entidades:

- **Departamento**
- **Estudiante**
- **Profesor**
- **Curso**
- **Clase**
- **Inscripcion**
- **Calificacion**

Crear una carpeta llamada trabajo final BD en su repositorio de base de datos.

Paso crear Base de datos



The screenshot shows the SQL Query window in SQL Server Enterprise Manager. The title bar reads 'SQLQuery1.s...G9\PC (53))*'. The query editor contains the command 'CREATE DATABASE gestion_academica;'. The status bar at the bottom indicates '100 %' zoom, 'No issues found', and a message pane showing 'Commands completed successfully.' with a completion time of '2026-01-20T23:41:21.7093027-04:00'.

```
SQLQuery1.s...G9\PC (53))*  
1 CREATE DATABASE gestion_academica;  
2  
3  
100 % No issues found  
Messages  
Commands completed successfully.  
Completion time: 2026-01-20T23:41:21.7093027-04:00
```

Códigos creación de tablas

Código crear tabla departamento:

```
USE gestion_academica;  
GO  
CREATE TABLE departamento(  
    id_departamento INT IDENTITY(1,1) PRIMARY KEY,  
    nombre VARCHAR(100) NOT NULL  
);
```

Código crear tabla estudiante:

```
USE gestion_academica;  
GO  
CREATE TABLE estudiante (  
    id_estudiante INT IDENTITY(1,1) PRIMARY KEY,  
    nombre VARCHAR(100),  
    apellido VARCHAR(100),  
    id_departamento INT,  
    FOREIGN KEY (id_departamento) REFERENCES departamento(id_departamento)  
);
```

Código crear tabla profesor:

```
USE gestion_academica;
GO
CREATE TABLE profesor (
    id_profesor INT IDENTITY(1,1) PRIMARY KEY,
    nombre VARCHAR(100),
    apellido VARCHAR(100),
    id_departamento INT,
    FOREIGN KEY (id_departamento) REFERENCES departamento(id_departamento)
);
```

Código crear tabla curso:

```
USE gestion_academica;
GO
CREATE TABLE curso (
    id_curso INT IDENTITY(1,1) PRIMARY KEY,
    nombre VARCHAR(100),
    id_departamento INT,
    FOREIGN KEY (id_departamento) REFERENCES departamento(id_departamento)
);
```

Código crear tabla clase:

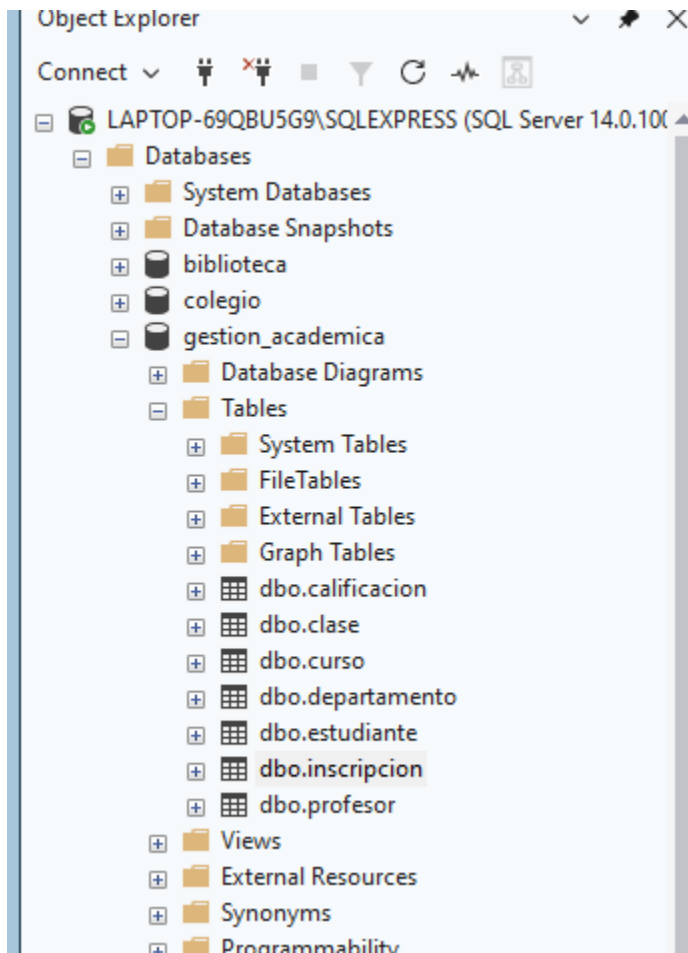
```
USE gestion_academica;
GO
CREATE TABLE clase (
    id_clase INT IDENTITY(1,1) PRIMARY KEY,
    id_curso INT,
    id_profesor INT,
    horario VARCHAR(50),
    FOREIGN KEY (id_curso) REFERENCES curso(id_curso),
    FOREIGN KEY (id_profesor) REFERENCES profesor(id_profesor)
);
```

Código crear tabla inscripción:

```
USE gestion_academica;
GO
CREATE TABLE inscripcion (
    id_inscripcion INT IDENTITY(1,1) PRIMARY KEY,
    id_estudiante INT,
    id_clase INT,
    fecha DATE,
    FOREIGN KEY (id_estudiante) REFERENCES estudiante(id_estudiante),
    FOREIGN KEY (id_clase) REFERENCES clase(id_clase)
);
```

Código crear tabla calificación:

```
USE gestion_academica;
GO
CREATE TABLE calificacion (
    id_calificacion INT IDENTITY(1,1) PRIMARY KEY,
    id_inscripcion INT,
    nota DECIMAL(5,2),
    FOREIGN KEY (id_inscripcion) REFERENCES inscripcion(id_inscripcion)
);
```



Inserción de datos (INSERT)

Tabla: Departamento

```
USE gestion_academica;
GO
INSERT INTO departamento (nombre) VALUES
('Sistemas'),
```

```
('Administración'),  
( 'Contabilidad'),  
( 'Mercadeo'),  
( 'Recursos Humanos');
```

Tabla: Estudiante

```
USE gestion_academica;  
GO  
INSERT INTO estudiante (nombre, apellido, id_departamento) VALUES  
( 'Juan', 'Pérez', 1),  
( 'Ana', 'Gómez', 1),  
( 'Luis', 'Martínez', 2),  
( 'María', 'Rodríguez', 3),  
( 'Pedro', 'López', 1),  
( 'Laura', 'Fernández', 4),  
( 'Carlos', 'Díaz', 5),  
( 'Sofía', 'Castillo', 2);
```

Tabla: Profesor

```
USE gestion_academica;  
GO  
INSERT INTO profesor (nombre, apellido, id_departamento) VALUES  
( 'Carlos', 'Ramírez', 1),  
( 'Elena', 'Morales', 2),  
( 'José', 'Santos', 3),  
( 'Patricia', 'Vargas', 4),  
( 'Ricardo', 'Peña', 1);
```

Tabla: Curso

```
USE gestion_academica;  
GO  
INSERT INTO curso (nombre, id_departamento) VALUES  
( 'Base de Datos', 1),  
( 'Programación I', 1),  
( 'Contabilidad General', 3),  
( 'Administración de Empresas', 2),  
( 'Marketing Digital', 4),  
( 'Gestión del Talento Humano', 5);
```

Tabla: Clase

```
USE gestion_academica;  
GO  
INSERT INTO clase (id_curso, id_profesor, horario) VALUES  
(1, 1, 'Lunes 8:00-10:00 AM'),  
(2, 5, 'Martes 10:00-12:00 PM'),
```

```
(3, 3, 'Miércoles 6:00-8:00 PM'),
(4, 2, 'Jueves 8:00-10:00 AM'),
(5, 4, 'Viernes 4:00-6:00 PM'),
(6, 2, 'Sábado 9:00-11:00 AM');
```

Tabla: Inscripción

```
USE gestion_academica;
GO
INSERT INTO inscripcion (id_estudiante, id_clase, fecha) VALUES
(1, 1, '2025-01-10'),
(2, 1, '2025-01-10'),
(3, 4, '2025-01-11'),
(4, 3, '2025-01-11'),
(5, 2, '2025-01-12'),
(6, 5, '2025-01-12'),
(7, 6, '2025-01-13'),
(8, 4, '2025-01-13'),
(1, 2, '2025-01-14'),
(2, 3, '2025-01-14');
```

Tabla: Calificación

```
USE gestion_academica;
GO
INSERT INTO calificacion (id_inscripcion, nota) VALUES
(1, 95.50),
(2, 88.75),
(3, 90.00),
(4, 85.25),
(5, 92.00),
(6, 89.50),
(7, 78.00),
(8, 91.75),
(9, 100.00),
(10, 87.25);
```

Consultas básicas

```
SELECT * FROM estudiante;
```

	id_estudiante	nombre	apellido	id_departamento
1	1	Juan	Pérez	1
2	2	Ana	Gómez	1
3	3	Luis	Martínez	2
4	4	María	Rodríguez	3
5	5	Pedro	López	1
6	6	Laura	Fernández	4
7	7	Carlos	Díaz	5
8	8	Sofía	Castillo	2

Query executed successfully.

SELECT * FROM curso;

	id_curso	nombre	id_departamento
1	1	Base de Datos	1
2	2	Programación I	1
3	3	Contabilidad General	3
4	4	Administración de Empresas	2
5	5	Marketing Digital	4
6	6	Gestión del Talento Humano	5

Query executed successfully.

SELECT * FROM calificacion;

	id_calificacion	id_inscripcion	nota
1	2	1	95.50
2	3	2	88.75
3	4	3	90.00
4	5	4	85.25
5	6	5	92.00
6	7	6	89.50
7	8	7	78.00
8	9	8	91.75
9	10	9	100.00
10	11	10	87.25

Query executed successfully.

Consultas con JOIN

Estudiantes con su departamento

```
SELECT e.nombre, e.apellido, d.nombre AS departamento
FROM estudiante e
INNER JOIN departamento d ON e.id_departamento = d.id_departamento;
```

	nombre	apellido	departamento
1	Juan	Pérez	Sistemas
2	Ana	Gómez	Sistemas
3	Luis	Martínez	Administración
4	María	Rodríguez	Contabilidad
5	Pedro	López	Sistemas
6	Laura	Fernández	Mercadeo
7	Carlos	Díaz	Recursos Humanos
8	Sofía	Castillo	Administración

Estudiantes con cursos inscritos

```
SELECT e.nombre, c.nombre AS curso
FROM estudiante e
INNER JOIN inscripcion i ON e.id_estudiante = i.id_estudiante
INNER JOIN clase cl ON i.id_clase = cl.id_clase
INNER JOIN curso c ON cl.id_curso = c.id_curso;
```

	nombre	curso
1	Juan	Base de Datos
2	Ana	Base de Datos
3	Luis	Administración de Empresas
4	María	Contabilidad General
5	Pedro	Programación I
6	Laura	Marketing Digital
7	Carlos	Gestión del Talento Humano
8	Sofía	Administración de Empresas
9	Juan	Programación I
10	Ana	Contabilidad General

Estadísticas (Funciones Agregadas)

Promedio de calificaciones

```
SELECT AVG(nota) AS promedio_general FROM calificacion;
```

Results		Messages
	promedio_general	
1	89.800000	

Cantidad de estudiantes por departamento

```
SELECT d.nombre, COUNT(e.id_estudiante) AS total_estudiantes  
FROM departamento d  
LEFT JOIN estudiante e ON d.id_departamento = e.id_departamento  
GROUP BY d.nombre;
```

Results			Messages
	nombre	total_estudiantes	
1	Administración	2	
2	Contabilidad	1	
3	Mercadeo	1	
4	Recursos Humanos	1	
5	Sistemas	3	

Manipulación de Datos

UPDATE

SQLQuery1.s...G9\PC (53))* ✕

```
1  USE gestion_academica;
2  GO
3  SELECT * FROM estudiante;
```

100 % ✕ 1 ⚠ 0 ↑ ↓

Results Messages

	id_estudiante	nombre	apellido	id_departamento
1	1	Juan	Pérez	1
2	2	Ana	Gómez	1

```
USE gestion_academica;
GO
UPDATE estudiante
SET apellido = 'Martínez'
WHERE id_estudiante = 1;
```

SQLQuery1.s...G9\PC (53))* ✕

```
1  USE gestion_academica;
2  GO
3  UPDATE estudiante
4  SET apellido = 'Martínez'
5  WHERE id_estudiante = 1;
6
```

100 % ✕ 3 ⚠ 0 ↑ ↓

Messages

(1 row affected)

Completion time: 2026-01-21T01:13:38.5171417-04:00

SQLQuery1.s...G9\PC (53))*

```

1  USE gestion_academica;
2  GO
3  SELECT * FROM estudiante;
4

```

100 % 1 0

Results Messages

	id_estudiante	nombre	apellido	id_departamento
1	1	Juan	Martínez	1
2	2	Ana	Gómez	1

DELETE

SQLQuery1.s...G9\PC (53))*

```

1  USE gestion_academica;
2  GO
3  SELECT * FROM calificacion;
4

```

100 % 1 0

Results Messages

	id_calificacion	id_inscripcion	nota
1	2	1	95.50
2	3	2	88.75
3	4	3	90.00
4	5	4	85.25
5	6	5	92.00
6	7	6	89.50
7	8	7	78.00
8	9	8	91.75
9	10	9	100.00
10	11	10	87.25

```
DELETE FROM calificacion WHERE id_calificacion = 2;
```

Messages

(1 row affected)

Completion time: 2026-01-21T01:16:43.9167314-04:00

SQLQuery1.s...G9\PC (53))* ✕

```
1 USE gestion_academica;
2 GO
3 SELECT * FROM calificacion;
4
```

100 % 1 0

Results Messages

	id_calificacion	id_inscripcion	nota
1	3	2	88.75
2	4	3	90.00
3	5	4	85.25
4	6	5	92.00
5	7	6	89.50
6	8	7	78.00
7	9	8	91.75
8	10	9	100.00
9	11	10	87.25

Relación 1:M (Resumen)

- Departamento → Estudiante
- Departamento → Profesor
- Departamento → Curso
- Curso → Clase
- Clase → Inscripción
- Inscripción → Calificación