



**TECNOLÓGICO
NACIONAL DE MÉXICO**



Clara Andrea Martinez Valdez

23490379

Fundamentos de Base de Datos

Jose Ramon Bogarin Valenzuela

Examen Final Unidad 4

Viernes 23 de mayo de 2025, a las 20:00 hrs

Problema Técnico: Gestionando la Información de una Universidad Imagina que estás trabajando en el equipo de desarrollo de software para una universidad. La universidad necesita un sistema para gestionar la información de sus estudiantes, los cursos que ofrecen, las inscripciones de los estudiantes a los cursos, los profesores que imparten los cursos y los departamentos a los que pertenecen los profesores. Tu tarea es diseñar y trabajar con la base de datos que almacenará toda esta información.

Objetivo General: Diseñar una base de datos relacional y realizar diversas operaciones para gestionar la información de la universidad. Esto incluye la creación y modificación de la estructura de las tablas, la manipulación de los datos y la realización de consultas complejas para obtener información específica.

Esquema de la Base de Datos: Aquí se presenta un esquema de la base de datos con 10 tablas y sus relaciones:

Modelo de Datos Lógico (MDL):

Tabla: Estudiantes

- IDEstudiante (Clave Principal)
- Nombre
- Apellido
- FechaNacimiento
- Direccion
- Ciudad
- Email

```
CREATE TABLE Estudiantes (  
  id_estudiante SERIAL PRIMARY KEY,  
  nombre VARCHAR(50) NOT NULL,  
  apellido VARCHAR(50) NOT NULL,  
  FechaNacimiento DATE,  
  direccion VARCHAR(80) NOT NULL,  
  ciudad VARCHAR(50) NOT NULL,  
  email VARCHAR(25) UNIQUE NOT NULL  
);
```

Tabla: Cursos

- IDCurso (Clave Principal)
- NombreCurso
- Descripcion
- Creditos
- Semestre
- IDDepartamento (Clave Foránea, referencia a la tabla Departamentos)

```
CREATE TABLE Cursos (  
  id_curso SERIAL PRIMARY KEY,  
  NombreCurso VARCHAR(50),  
  descripcion TEXT,  
  creditos INT,  
  semestre INT,  
  id_departamento INT REFERENCES Departamentos (id_departamento)  
);
```

Tabla: Inscripciones

- IDInscripcion (Clave Principal)
- IDEstudiante (Clave Foránea, referencia a la tabla Estudiantes)
- IDCurso (Clave Foránea, referencia a la tabla Cursos)
- FechaInscripcion
- Calificacion

```
CREATE TABLE Inscripciones (  
  id_inscripcion SERIAL PRIMARY KEY,  
  id_estudiante INT REFERENCES Estudiantes(id_estudiante),  
  id_curso INT REFERENCES Cursos(id_curso),  
  FechaInscripcion DATE NOT NULL,  
  Calificacion DECIMAL  
);
```

Tabla: Profesores

- IDProfesor (Clave Principal)
- Nombre
- Apellido
- Titulo
- IDDepartamento (Clave Foránea, referencia a la tabla Departamentos)

```
CREATE TABLE Profesores (  
  id_profesor SERIAL PRIMARY KEY,  
  Nombre VARCHAR(50) NOT NULL,  
  Apellido VARCHAR(50) NOT NULL,  
  Titulo VARCHAR(100),  
  id_departamento INT REFERENCES Departamentos(id_departamento)  
);
```

Tabla: Departamentos

- IDDepartamento (Clave Principal)
- NombreDepartamento
- Edificio

```
CREATE TABLE Departamentos (  
  id_departamento SERIAL PRIMARY KEY,  
  NombreDepartamento VARCHAR(100) NOT NULL,  
  Edificio VARCHAR(100)  
);
```

Tabla: Aulas

- IDAula (Clave Principal)
- NombreAula
- Capacidad
- Ubicacion

```
CREATE TABLE Aulas (  
  id_aula SERIAL PRIMARY KEY,  
  NombreAula VARCHAR(50) NOT NULL,  
  Capacidad INT,  
  Ubicacion VARCHAR(80)  
);
```

Tabla: Horarios

- IDHorario (Clave Principal)
- IDCurso (Clave Foránea, referencia a la tabla Cursos)
- IDAula (Clave Foránea, referencia a la tabla Aulas)
- FechaInicio
- FechaFin
- HoraInicio
- HoraFin

```
CREATE TABLE Horarios (  
  id_horario SERIAL PRIMARY KEY,  
  id_curso INT REFERENCES Cursos(id_curso),  
  id_aula INT REFERENCES Aulas(id_aula),  
  FechaInicio DATE,  
  FechaFin DATE,  
  HoraInicio TIME,  
  HoraFin TIME  
);
```

Tabla Intermedia: CursosProfesores (Relación Muchos a Muchos entre Cursos y Profesores)

- IDCursoProfesor (Clave Principal)
- IDCurso (Clave Foránea, referencia a la tabla Cursos)
- IDProfesor (Clave Foránea, referencia a la tabla Profesores)

```
CREATE TABLE CursosProfesores (  
  id_cursoprofesor SERIAL PRIMARY KEY,  
  id_curso INT REFERENCES Cursos(id_curso),  
  id_profesor INT REFERENCES Profesores (id_profesor)  
);
```

Tabla: ProgramasEstudio

- IDPrograma (Clave Principal)
- NombrePrograma
- DescripcionPrograma

```
CREATE TABLE ProgramasEstudio (  
  id_ProgramaEstudio SERIAL PRIMARY KEY,  
  NombrePrograma VARCHAR(100) NOT NULL,  
  DescripcionPrograma TEXT  
);
```

Tabla Intermedia: ProgramasCursos (Relación Muchos a Muchos entre ProgramasEstudio y Cursos)

- IDProgramaCurso (Clave Principal)
- IDPrograma (Clave Foránea, referencia a la tabla ProgramasEstudio)
- IDCurso (Clave Foránea, referencia a la tabla Cursos)

```
CREATE TABLE ProgramasCursos (  
  id_ProgramaCurso SERIAL PRIMARY KEY,  
  id_ProgramaEstudio INT REFERENCES ProgramasEstudio(id_ProgramaEstudio),  
  id_curso INT REFERENCES Cursos(id_curso)  
);
```

Modificando las Tablas (DDL): Realizar modificaciones a la estructura de las tablas, como agregar, modificar o eliminar campos, según sea necesario.

- Agregar Tablas:

Tabla: Campus

- IDCampus (Clave Principal)
- NombreCampus
- DireccionCampus

Tabla: Carreras

- IDCarrera (Clave Principal)
- NombreCarrera
- TituloOtorgado

- Agregar una relación de muchos a muchos entre Estudiantes y Carreras

```
CREATE TABLE Campus (
  id_campus SERIAL PRIMARY KEY,
  NombreCampus VARCHAR(100) NOT NULL,
  DireccionCampus VARCHAR(200)
);

CREATE TABLE Carreras (
  id_carrera SERIAL PRIMARY KEY,
  NombreCarrera VARCHAR(100) NOT NULL,
  TituloOtorgado VARCHAR(100)
);

CREATE TABLE Estudiantes_Carreras (
  id_estudiante_carreras SERIAL PRIMARY KEY,
  id_estudiante INT NOT NULL,
  id_carrera INT NOT NULL,
  FechaInscripcion DATE DEFAULT CURRENT_DATE,
  FOREIGN KEY (id_estudiante) REFERENCES Estudiantes(id_estudiante),
  FOREIGN KEY (id_carrera) REFERENCES Carreras(id_carrera)
);
```

- **Modificar Tablas:**

- En la tabla **Estudiantes**, agregar una clave foránea IDCarrera que haga referencia a la tabla Carreras.

```
ALTER TABLE Estudiantes
ADD COLUMN id_carrera INT;

ALTER TABLE Estudiantes
ADD CONSTRAINT FK_estudiante_carrera
FOREIGN KEY (id_carrera) REFERENCES Carreras(id_carrera)
ON DELETE SET NULL;
```

| | | | |
|-------------------------------------|----------------------------------|------------------------------------|-------------------------|
| id_estudiante [PK] integer | nombre character varying (50) | apellido character varying (50) | fechanacimiento date |
| direccion character varying (80) | ciudad character varying (50) | email character varying (25) | id_carrera integer |

- En la tabla **Cursos**, agregar una columna IDCampus como clave foránea, referenciando la tabla Campus.

```
ALTER TABLE Cursos
ADD COLUMN id_campus INT,

ADD CONSTRAINT fk_curso_campus
FOREIGN KEY (id_campus) REFERENCES Campus(id_campus)
ON DELETE SET NULL;
```

| | | | | | | |
|--------------------------|---------------------------------------|---------------------|---------------------|---------------------|----------------------------|----------------------|
| id_curso [PK] integer | nombrecurso character varying (50) | descripcion text | creditos integer | semestre integer | id_departamento integer | id_campus integer |
|--------------------------|---------------------------------------|---------------------|---------------------|---------------------|----------------------------|----------------------|

- Modificar la tabla **Profesores** para incluir un campo Email

```
ALTER TABLE Profesores
ADD COLUMN Email VARCHAR(50) UNIQUE;
```

| | | | | | |
|-----------------------------|----------------------------------|------------------------------------|-----------------------------------|----------------------------|---------------------------------|
| id_profesor [PK] integer | nombre character varying (50) | apellido character varying (50) | titulo character varying (100) | id_departamento integer | email character varying (50) |
|-----------------------------|----------------------------------|------------------------------------|-----------------------------------|----------------------------|---------------------------------|

- **Eliminar Tablas/Campos**

- Eliminar la columna **Ciudad** de la tabla **Estudiantes**.

```
ALTER TABLE Estudiantes
DROP COLUMN ciudad;
```

| | | | | | | |
|-------------------------------|----------------------------------|------------------------------------|-------------------------|-------------------------------------|---------------------------------|-----------------------|
| id_estudiante [PK] integer | nombre character varying (50) | apellido character varying (50) | fechanacimiento date | direccion character varying (80) | email character varying (25) | id_carrera integer |
|-------------------------------|----------------------------------|------------------------------------|-------------------------|-------------------------------------|---------------------------------|-----------------------|

- Eliminar la tabla **Aulas**

```
DROP TABLE Aulas CASCADE ;
```

Insertando Datos: Insertar datos de ejemplo en las tablas para representar la información de la universidad.

Tabla Campus

```
INSERT INTO Campus (NombreCampus, DireccionCampus) VALUES
('Campus Central', 'Av. Universidad 1000, Lima'),
('Campus Norte', 'Jr. Libertad 321, Trujillo'),
('Campus Sur', 'Av. Independencia 456, Arequipa'),
('Campus Este', 'Calle Sol 789, Cusco'),
('Campus Oeste', 'Av. América 234, Piura'),
('Campus San Miguel', 'Jr. Grau 111, Lima'),
('Campus Los Olivos', 'Av. Túpac Amaru 567, Lima'),
('Campus Santa Anita', 'Av. Metropolitana 654, Lima'),
('Campus Callao', 'Calle Comercio 789, Callao'),
('Campus Villa El Salvador', 'Av. 200 Millas 101, Lima');
```

| | id_campus [PK] integer | nombrecampus character varying (100) | direccioncampus character varying (200) |
|----|---------------------------|---|--|
| 1 | 1 | Campus Central | Av. Universidad 1000, Lima |
| 2 | 2 | Campus Norte | Jr. Libertad 321, Trujillo |
| 3 | 3 | Campus Sur | Av. Independencia 456, Arequipa |
| 4 | 4 | Campus Este | Calle Sol 789, Cusco |
| 5 | 5 | Campus Oeste | Av. América 234, Piura |
| 6 | 6 | Campus San Miguel | Jr. Grau 111, Lima |
| 7 | 7 | Campus Los Olivos | Av. Túpac Amaru 567, Lima |
| 8 | 8 | Campus Santa Anita | Av. Metropolitana 654, Lima |
| 9 | 9 | Campus Callao | Calle Comercio 789, Callao |
| 10 | 10 | Campus Villa El Salvador | Av. 200 Millas 101, Lima |

Tabla Carrera

```
INSERT INTO Carreras (NombreCarrera, TituloOtorgado) VALUES
('Ingeniería de Sistemas', 'Ingeniero de Sistemas'),
('Matemáticas Aplicadas', 'Licenciado en Matemáticas'),
('Física', 'Licenciado en Física'),
('Biología Molecular', 'Licenciado en Biología'),
('Química Industrial', 'Ingeniero Químico'),
('Literatura Hispanoamericana', 'Licenciado en Literatura'),
('Filosofía y Ética', 'Licenciado en Filosofía'),
('Economía y Finanzas', 'Economista'),
('Psicología Clínica', 'Psicólogo'),
('Historia del Perú', 'Licenciado en Historia');
```

| | id_carrera [PK] integer | nombrecarrera character varying (100) | titulootorgado character varying (100) |
|----|----------------------------|--|---|
| 1 | 1 | Ingeniería de Sistemas | Ingeniero de Sistemas |
| 2 | 2 | Matemáticas Aplicadas | Licenciado en Matemáticas |
| 3 | 3 | Física | Licenciado en Física |
| 4 | 4 | Biología Molecular | Licenciado en Biología |
| 5 | 5 | Química Industrial | Ingeniero Químico |
| 6 | 6 | Literatura Hispanoamericana | Licenciado en Literatura |
| 7 | 7 | Filosofía y Ética | Licenciado en Filosofía |
| 8 | 8 | Economía y Finanzas | Economista |
| 9 | 9 | Psicología Clínica | Psicólogo |
| 10 | 10 | Historia del Perú | Licenciado en Historia |

Tabla Cursos

```
INSERT INTO Cursos (NombreCurso, descripcion, credits, semestre, id_departamento, id_campus) VALUES
('Programación I', 'Introducción a la programación', 4, 1, 1, 1),
('Álgebra Lineal', 'Matrices y vectores', 3, 1, 2, 2),
('Física I', 'Cinemática y dinámica', 4, 1, 3, 3),
('Biología General', 'Estructura celular y genética', 3, 1, 4, 4),
('Química Orgánica', 'Estudio de compuestos orgánicos', 3, 2, 5, 5),
('Literatura Española', 'Autores clásicos', 2, 2, 6, 6),
('Lógica', 'Fundamentos del pensamiento crítico', 2, 2, 7, 7),
('Macroeconomía', 'Economía a nivel nacional', 3, 2, 8, 8),
('Psicología del Desarrollo', 'Etapas del desarrollo humano', 3, 2, 9, 9),
('Historia Universal', 'De la antigüedad a la modernidad', 3, 2, 10, 10);
```

| | id_curso [PK] integer | nombrecurso character varying (50) | descripcion text | credits integer | semestre integer | id_departamento integer | id_campus integer |
|----|--------------------------|---------------------------------------|-------------------------------------|--------------------|---------------------|----------------------------|----------------------|
| 1 | 1 | Programación I | Introducción a la programación | 4 | 1 | 1 | 1 |
| 2 | 2 | Álgebra Lineal | Matrices y vectores | 3 | 1 | 2 | 2 |
| 3 | 3 | Física I | Cinemática y dinámica | 4 | 1 | 3 | 3 |
| 4 | 4 | Biología General | Estructura celular y genética | 3 | 1 | 4 | 4 |
| 5 | 5 | Química Orgánica | Estudio de compuestos orgánicos | 3 | 2 | 5 | 5 |
| 6 | 6 | Literatura Española | Autores clásicos | 2 | 2 | 6 | 6 |
| 7 | 7 | Lógica | Fundamentos del pensamiento crítico | 2 | 2 | 7 | 7 |
| 8 | 8 | Macroeconomía | Economía a nivel nacional | 3 | 2 | 8 | 8 |
| 9 | 9 | Psicología del Desarrollo | Etapas del desarrollo humano | 3 | 2 | 9 | 9 |
| 10 | 10 | Historia Universal | De la antigüedad a la modernidad | 3 | 2 | 10 | 10 |

Tabla Departamentos

```
INSERT INTO Departamentos (NombreDepartamento, Edificio) VALUES
('Ingeniería Informática', 'Edificio A'),
('Matemáticas', 'Edificio B'),
('Física', 'Edificio C'),
('Biología', 'Edificio D'),
('Química', 'Edificio E'),
('Literatura', 'Edificio F'),
('Filosofía', 'Edificio G'),
('Economía', 'Edificio H'),
('Psicología', 'Edificio I'),
('Historia', 'Edificio J');
```

| | id_departamento [PK] integer | nombredepartamento character varying (100) | edificio character varying (100) |
|----|---------------------------------|---|-------------------------------------|
| 1 | 1 | Ingeniería Informática | Edificio A |
| 2 | 2 | Matemáticas | Edificio B |
| 3 | 3 | Física | Edificio C |
| 4 | 4 | Biología | Edificio D |
| 5 | 5 | Química | Edificio E |
| 6 | 6 | Literatura | Edificio F |
| 7 | 7 | Filosofía | Edificio G |
| 8 | 8 | Economía | Edificio H |
| 9 | 9 | Psicología | Edificio I |
| 10 | 10 | Historia | Edificio J |

Tabla Estudiantes

```
INSERT INTO Estudiantes (nombre, apellido, FechaNacimiento, direccion, email) VALUES
('Juan', 'Pérez', '2000-04-10', 'Calle Falsa 123', 'juanp@email.com'),
('María', 'García', '1999-09-15', 'Av. Central 456', 'mariag@email.com'),
('Luis', 'Torres', '2001-12-01', 'Jr. Lima 789', 'luist@email.com'),
('Ana', 'Ramírez', '2002-01-20', 'Calle Sur 321', 'anar@email.com'),
('Pedro', 'Sánchez', '2000-06-11', 'Av. Norte 654', 'pedros@email.com'),
('Lucía', 'Flores', '1998-03-22', 'Pasaje Uno 111', 'luciaf@email.com'),
('Carlos', 'Lopez', '2003-05-30', 'Jr. Este 222', 'carlosl@email.com'),
('Diana', 'Reyes', '2000-07-25', 'Av. Oeste 333', 'dianar@email.com'),
('Jorge', 'Castro', '1999-11-19', 'Calle Real 999', 'jorgec@email.com'),
('Andrea', 'Mendoza', '2001-08-14', 'Av. Central 007', 'andream@email.com');
```

| | id_estudiante [PK] integer | nombre character varying (50) | apellido character varying (50) | fechanacimiento date | direccion character varying (80) | email character varying (25) | id_carrera integer |
|----|-------------------------------|----------------------------------|------------------------------------|-------------------------|-------------------------------------|---------------------------------|-----------------------|
| 1 | 1 | Juan | Pérez | 2000-04-10 | Calle Falsa 123 | juanp@email.com | [null] |
| 2 | 2 | María | García | 1999-09-15 | Av. Central 456 | mariag@email.com | [null] |
| 3 | 3 | Luis | Torres | 2001-12-01 | Jr. Lima 789 | luist@email.com | [null] |
| 4 | 4 | Ana | Ramírez | 2002-01-20 | Calle Sur 321 | anar@email.com | [null] |
| 5 | 5 | Pedro | Sánchez | 2000-06-11 | Av. Norte 654 | pedros@email.com | [null] |
| 6 | 6 | Lucía | Flores | 1998-03-22 | Pasaje Uno 111 | luciaf@email.com | [null] |
| 7 | 7 | Carlos | Lopez | 2003-05-30 | Jr. Este 222 | carlosl@email.com | [null] |
| 8 | 8 | Diana | Reyes | 2000-07-25 | Av. Oeste 333 | dianar@email.com | [null] |
| 9 | 9 | Jorge | Castro | 1999-11-19 | Calle Real 999 | jorgec@email.com | [null] |
| 10 | 10 | Andrea | Mendoza | 2001-08-14 | Av. Central 007 | andream@email.com | [null] |

Tabla Profesores

```
INSERT INTO Profesores (Nombre, Apellido, Titulo, id_departamento, email) VALUES
('Ricardo', 'Fernández', 'PhD en Informática', 1, 'Ricardo@email.com'),
('Marta', 'Jiménez', 'MSc en Matemáticas', 2, 'Marta@email.com'),
('Javier', 'Ruiz', 'PhD en Física', 3, 'Javier@email.com'),
('Elena', 'Navarro', 'MSc en Biología', 4, 'Elena@email.com'),
('Alfredo', 'Castañeda', 'PhD en Química', 5, 'Alfredo@email.com'),
('Carmen', 'Villanueva', 'Lic. en Letras', 6, 'Carmen@email.com'),
('Sofía', 'Herrera', 'PhD en Filosofía', 7, 'Sofia@email.com'),
('Daniel', 'Salas', 'MSc en Economía', 8, 'Daniel@email.com'),
('Paula', 'Delgado', 'PhD en Psicología', 9, 'Paula@email.com'),
('Oscar', 'Ortiz', 'PhD en Historia', 10, 'Oscar@email.com');
```

| | id_profesor [PK] integer | nombre character varying (50) | apellido character varying (50) | titulo character varying (100) | id_departamento integer | email character varying (50) |
|----|-----------------------------|----------------------------------|------------------------------------|-----------------------------------|----------------------------|---------------------------------|
| 1 | 11 | Ricardo | Fernández | PhD en Informática | 1 | Ricardo@email.com |
| 2 | 12 | Marta | Jiménez | MSc en Matemáticas | 2 | Marta@email.com |
| 3 | 13 | Javier | Ruiz | PhD en Física | 3 | Javier@email.com |
| 4 | 14 | Elena | Navarro | MSc en Biología | 4 | Elena@email.com |
| 5 | 15 | Alfredo | Castañeda | PhD en Química | 5 | Alfredo@email.com |
| 6 | 16 | Carmen | Villanueva | Lic. en Letras | 6 | Carmen@email.com |
| 7 | 17 | Sofía | Herrera | PhD en Filosofía | 7 | Sofia@email.com |
| 8 | 18 | Daniel | Salas | MSc en Economía | 8 | Daniel@email.com |
| 9 | 19 | Paula | Delgado | PhD en Psicología | 9 | Paula@email.com |
| 10 | 20 | Oscar | Ortiz | PhD en Historia | 10 | Oscar@email.com |

Tabla Inscripciones

```
INSERT INTO Inscripciones (id_estudiante, id_curso, FechaInscripcion, Calificacion) VALUES
(1, 1, '2025-01-15', 15.5),
(2, 2, '2025-01-16', 17.2),
(3, 3, '2025-01-17', 14.0),
(4, 4, '2025-01-18', 18.0),
(5, 5, '2025-01-19', 16.8),
(6, 6, '2025-01-20', 19.0),
(7, 7, '2025-01-21', 13.5),
(8, 8, '2025-01-22', 17.5),
(9, 9, '2025-01-23', 18.3),
(10, 10, '2025-01-24', 14.9);
```

| | id_inscripcion [PK] integer | id_estudiante integer | id_curso integer | fechainscripcion date | calificacion numeric |
|----|--------------------------------|--------------------------|---------------------|--------------------------|-------------------------|
| 1 | 1 | 1 | 1 | 2025-01-15 | 15.5 |
| 2 | 2 | 2 | 2 | 2025-01-16 | 17.2 |
| 3 | 3 | 3 | 3 | 2025-01-17 | 14.0 |
| 4 | 4 | 4 | 4 | 2025-01-18 | 18.0 |
| 5 | 5 | 5 | 5 | 2025-01-19 | 16.8 |
| 6 | 6 | 6 | 6 | 2025-01-20 | 19.0 |
| 7 | 7 | 7 | 7 | 2025-01-21 | 13.5 |
| 8 | 8 | 8 | 8 | 2025-01-22 | 17.5 |
| 9 | 9 | 9 | 9 | 2025-01-23 | 18.3 |
| 10 | 10 | 10 | 10 | 2025-01-24 | 14.9 |

Tabla Horarios

```
INSERT INTO Horarios (id_curso, id_aula, FechaInicio, FechaFin, HoraInicio, HoraFin) VALUES
(1, 1, '2025-03-01', '2025-06-30', '08:00', '10:00'),
(2, 2, '2025-03-01', '2025-06-30', '10:00', '12:00'),
(3, 3, '2025-03-01', '2025-06-30', '12:00', '14:00'),
(4, 4, '2025-03-01', '2025-06-30', '08:00', '10:00'),
(5, 5, '2025-03-01', '2025-06-30', '10:00', '12:00'),
(6, 6, '2025-03-01', '2025-06-30', '12:00', '14:00'),
(7, 7, '2025-03-01', '2025-06-30', '14:00', '16:00'),
(8, 8, '2025-03-01', '2025-06-30', '08:00', '10:00'),
(9, 9, '2025-03-01', '2025-06-30', '10:00', '12:00'),
(10, 10, '2025-03-01', '2025-06-30', '12:00', '14:00');
```

| | id_horario [PK] integer | id_curso integer | id_aula integer | fechainicio date | fechafin date | horainicio time without time zone | horafin time without time zone |
|----|----------------------------|---------------------|--------------------|---------------------|------------------|--------------------------------------|-----------------------------------|
| 1 | 11 | 1 | 1 | 2025-03-01 | 2025-06-30 | 08:00:00 | 10:00:00 |
| 2 | 12 | 2 | 2 | 2025-03-01 | 2025-06-30 | 10:00:00 | 12:00:00 |
| 3 | 13 | 3 | 3 | 2025-03-01 | 2025-06-30 | 12:00:00 | 14:00:00 |
| 4 | 14 | 4 | 4 | 2025-03-01 | 2025-06-30 | 08:00:00 | 10:00:00 |
| 5 | 15 | 5 | 5 | 2025-03-01 | 2025-06-30 | 10:00:00 | 12:00:00 |
| 6 | 16 | 6 | 6 | 2025-03-01 | 2025-06-30 | 12:00:00 | 14:00:00 |
| 7 | 17 | 7 | 7 | 2025-03-01 | 2025-06-30 | 14:00:00 | 16:00:00 |
| 8 | 18 | 8 | 8 | 2025-03-01 | 2025-06-30 | 08:00:00 | 10:00:00 |
| 9 | 19 | 9 | 9 | 2025-03-01 | 2025-06-30 | 10:00:00 | 12:00:00 |
| 10 | 20 | 10 | 10 | 2025-03-01 | 2025-06-30 | 12:00:00 | 14:00:00 |

Tabla CursosProfesores

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

| | id_cursoprofesor [PK] integer | id_curso integer | id_profesor integer |
|----|----------------------------------|---------------------|------------------------|
| 1 | 11 | 1 | 11 |
| 2 | 12 | 2 | 12 |
| 3 | 13 | 3 | 13 |
| 4 | 14 | 4 | 14 |
| 5 | 15 | 5 | 15 |
| 6 | 16 | 6 | 16 |
| 7 | 17 | 7 | 17 |
| 8 | 18 | 8 | 18 |
| 9 | 19 | 9 | 19 |
| 10 | 20 | 10 | 20 |

Tabla ProgramasEstudio

```
INSERT INTO ProgramasEstudio (NombrePrograma, DescripcionPrograma) VALUES
('Ingeniería Informática', 'Programa de formación en informática y programación'),
('Matemáticas Aplicadas', 'Enfoque en cálculo y análisis de datos'),
('Ciencias Físicas', 'Carrera en investigación y física aplicada'),
('Biociencias', 'Estudios en biología celular y genética'),
('Química Pura', 'Investigación y laboratorios químicos'),
('Letras y Humanidades', 'Estudio literario y crítico de textos'),
('Filosofía Clásica', 'Análisis filosófico desde la antigüedad'),
('Economía Global', 'Entorno económico internacional y finanzas'),
('Psicología Clínica', 'Psicología del comportamiento humano'),
('Historia Contemporánea', 'Estudios históricos modernos y contemporáneos');
```

| | id_programaestudio [PK] integer | nombreprograma character varying (100) | descripcionprograma text |
|----|------------------------------------|---|---|
| 1 | 1 | Ingeniería Informática | Programa de formación en informática y programación |
| 2 | 2 | Matemáticas Aplicadas | Enfoque en cálculo y análisis de datos |
| 3 | 3 | Ciencias Físicas | Carrera en investigación y física aplicada |
| 4 | 4 | Biociencias | Estudios en biología celular y genética |
| 5 | 5 | Química Pura | Investigación y laboratorios químicos |
| 6 | 6 | Letras y Humanidades | Estudio literario y crítico de textos |
| 7 | 7 | Filosofía Clásica | Análisis filosófico desde la antigüedad |
| 8 | 8 | Economía Global | Entorno económico internacional y finanzas |
| 9 | 9 | Psicología Clínica | Psicología del comportamiento humano |
| 10 | 10 | Historia Contemporánea | Estudios históricos modernos y contemporáneos |

Tabla ProgramasCursos

```
INSERT INTO ProgramasCursos (id_ProgramaEstudio, id_curso) VALUES
(1, 1),
(2, 2),
(3, 3),
(4, 4),
(5, 5),
(6, 6),
(7, 7),
(8, 8),
(9, 9),
(10, 10);
```

| | id_programacurso [PK] integer | id_programaestudio integer | id_curso integer |
|----|----------------------------------|-------------------------------|---------------------|
| 1 | 1 | 1 | 1 |
| 2 | 2 | 2 | 2 |
| 3 | 3 | 3 | 3 |
| 4 | 4 | 4 | 4 |
| 5 | 5 | 5 | 5 |
| 6 | 6 | 6 | 6 |
| 7 | 7 | 7 | 7 |
| 8 | 8 | 8 | 8 |
| 9 | 9 | 9 | 9 |
| 10 | 10 | 10 | 10 |

Tabla EstudianteCarrera

```
INSERT INTO Estudiantes_Carreras (id_estudiante, id_carrera, FechaInscripcion) VALUES
(1, 1, '2024-03-15'),
(2, 2, '2024-03-16'),
(3, 3, '2024-03-17'),
(4, 4, '2024-03-18'),
(5, 5, '2024-03-19'),
(6, 6, '2024-03-20'),
(7, 7, '2024-03-21'),
(8, 8, '2024-03-22'),
(9, 9, '2024-03-23'),
(10, 10, '2024-03-24');
```

| | id_estudiante_carreras [PK] integer | id_estudiante integer | id_carrera integer | fechainscripcion date |
|----|--|--------------------------|-----------------------|--------------------------|
| 1 | 1 | 1 | 1 | 2024-03-15 |
| 2 | 2 | 2 | 2 | 2024-03-16 |
| 3 | 3 | 3 | 3 | 2024-03-17 |
| 4 | 4 | 4 | 4 | 2024-03-18 |
| 5 | 5 | 5 | 5 | 2024-03-19 |
| 6 | 6 | 6 | 6 | 2024-03-20 |
| 7 | 7 | 7 | 7 | 2024-03-21 |
| 8 | 8 | 8 | 8 | 2024-03-22 |
| 9 | 9 | 9 | 9 | 2024-03-23 |
| 10 | 10 | 10 | 10 | 2024-03-24 |

- **Actualizando Datos:** Actualizar la información existente en las tablas para reflejar cambios o correcciones.
- **Eliminando Datos:** Eliminar registros de las tablas que ya no sean relevantes.
- **Realizando Consultas (Búsquedas):** Formular y ejecutar consultas para obtener información específica de la base de datos.

Consultas Específicas:

Los estudiantes deben formular consultas para responder a las siguientes solicitudes:

1. Selección Básica: Muestra todos los nombres y apellidos de los estudiantes.

```
SELECT nombre, apellido
FROM Estudiantes;
```

| | nombre character varying (50) | apellido character varying (50) |
|----|----------------------------------|------------------------------------|
| 1 | Juan | Pérez |
| 2 | María | García |
| 3 | Luis | Torres |
| 4 | Ana | Ramírez |
| 5 | Pedro | Sánchez |
| 6 | Lucía | Flores |
| 7 | Carlos | Lopez |
| 8 | Diana | Reyes |
| 9 | Jorge | Castro |
| 10 | Andrea | Mendoza |

2. Cláusula WHERE: Encuentra todos los cursos que tienen 3 créditos.

```
SELECT *
FROM Cursos
WHERE credits = 3;
```

| | id_curso [PK] integer | nombrecurso character varying (50) | descripcion text | credits integer | semestre integer | id_departamento integer | id_campus integer |
|---|--------------------------|---------------------------------------|------------------------------------|--------------------|---------------------|----------------------------|----------------------|
| 1 | 2 | Álgebra Lineal | Matrices y vectores | 3 | 1 | 2 | 2 |
| 2 | 4 | Biología General | Estructura celular y genética | 3 | 1 | 4 | 4 |
| 3 | 5 | Química Orgánica | Estudio de compuestos orgánicos... | 3 | 2 | 5 | 5 |
| 4 | 8 | Macroeconomía | Economía a nivel nacional | 3 | 2 | 8 | 8 |
| 5 | 9 | Psicología del Desarrollo | Etapas del desarrollo humano | 3 | 2 | 9 | 9 |
| 6 | 10 | Historia Universal | De la antigüedad a la modernidad | 3 | 2 | 10 | 10 |

3. INNER JOIN: Obtén una lista que muestre el nombre del estudiante y el nombre del curso en el que está inscrito.

```
SELECT *
FROM Cursos
WHERE credits = 3;

SELECT
    e.nombre AS nombreEstudiante,
    e.apellido AS apellidoEstudiante,
    c.NombreCurso
FROM
    Inscripciones i
INNER JOIN Estudiantes e ON i.id_estudiante = e.id_estudiante
INNER JOIN Cursos c ON i.id_curso = c.id_curso;
```

| | nombrestudiante character varying (50) | apellidoeestudiante character varying (50) | nombrecurso character varying (50) |
|----|---|---|---------------------------------------|
| 1 | Juan | Pérez | Programación I |
| 2 | María | García | Álgebra Lineal |
| 3 | Luis | Torres | Física I |
| 4 | Ana | Ramírez | Biología General |
| 5 | Pedro | Sánchez | Química Orgánica |
| 6 | Lucía | Flores | Literatura Española |
| 7 | Carlos | Lopez | Lógica |
| 8 | Diana | Reyes | Macroeconomía |
| 9 | Jorge | Castro | Psicología del Desarrollo |
| 10 | Andrea | Mendoza | Historia Universal |

4. LEFT JOIN: Muestra todos los estudiantes y, si están inscritos en algún curso, el nombre del curso. Si un estudiante no está inscrito en ningún curso, el campo del nombre del curso debe mostrar un valor que lo indique (ej: NULL).

```
SELECT
    e.nombre AS NombreEstudiante,
    e.apellido AS ApellidoEstudiante,
    c.NombreCurso
FROM
    Estudiantes e
LEFT JOIN Inscripciones i ON e.id_estudiante = i.id_estudiante
LEFT JOIN Cursos c ON i.id_curso = c.id_curso;
```

| | nombrestudiante character varying (50) | apellidoeestudiante character varying (50) | nombrecurso character varying (50) |
|----|---|---|---------------------------------------|
| 1 | Juan | Pérez | Programación I |
| 2 | María | García | Álgebra Lineal |
| 3 | Luis | Torres | Física I |
| 4 | Ana | Ramírez | Biología General |
| 5 | Pedro | Sánchez | Química Orgánica |
| 6 | Lucía | Flores | Literatura Española |
| 7 | Carlos | Lopez | Lógica |
| 8 | Diana | Reyes | Macroeconomía |
| 9 | Jorge | Castro | Psicología del Desarrollo |
| 10 | Andrea | Mendoza | Historia Universal |

5. RIGHT JOIN: Lista todos los cursos y, si tienen estudiantes inscritos, el nombre de los estudiantes. Muestra todos los cursos, incluso si no tienen estudiantes inscritos actualmente.

```
SELECT c.NombreCurso,
       e.nombre AS NombreEstudiante,
       e.apellido AS ApellidoEstudiante
FROM Estudiantes e
RIGHT JOIN Inscripciones i ON e.id_estudiante = i.id_estudiante
RIGHT JOIN Cursos c ON i.id_curso = c.id_curso;
```

| | nombrecurso character varying (50) 🔒 | nombreestudiante character varying (50) 🔒 | apellidoestudiante character varying (50) 🔒 |
|----|---|--|--|
| 1 | Programación I | Juan | Pérez |
| 2 | Álgebra Lineal | María | García |
| 3 | Física I | Luis | Torres |
| 4 | Biología General | Ana | Ramírez |
| 5 | Química Orgánica | Pedro | Sánchez |
| 6 | Literatura Española | Lucía | Flores |
| 7 | Lógica | Carlos | Lopez |
| 8 | Macroeconomía | Diana | Reyes |
| 9 | Psicología del Desarrollo | Jorge | Castro |
| 10 | Historia Universal | Andrea | Mendoza |

6. GROUP BY y COUNT: Calcula cuántos estudiantes están inscritos en cada curso. Muestra el nombre del curso y la cantidad de estudiantes.

```
SELECT c.NombreCurso,
       COUNT(i.id_estudiante) AS CantidadEstudiantes
FROM Cursos c
LEFT JOIN Inscripciones i ON c.id_curso = i.id_curso
GROUP BY c.NombreCurso;
```

| | nombrecurso character varying (50) 🔒 | cantidadestudiantes bigint 🔒 |
|----|---|---------------------------------|
| 1 | Biología General | 1 |
| 2 | Física I | 1 |
| 3 | Álgebra Lineal | 1 |
| 4 | Literatura Española | 1 |
| 5 | Psicología del Desarrollo | 1 |
| 6 | Programación I | 1 |
| 7 | Macroeconomía | 1 |
| 8 | Historia Universal | 1 |
| 9 | Química Orgánica | 1 |
| 10 | Lógica | 1 |

7. BETWEEN: Encuentra todos los estudiantes que nacieron entre el 1 de enero de 1995 y el 31 de diciembre de 1998.

```
SELECT nombre, apellido, FechaNacimiento
FROM Estudiantes
WHERE FechaNacimiento BETWEEN '1998-01-01' AND '2003-12-31';
```

| | nombre character varying (50) | apellido character varying (50) | fechanacimiento date |
|----|----------------------------------|------------------------------------|-------------------------|
| 1 | Juan | Pérez | 2000-04-10 |
| 2 | María | García | 1999-09-15 |
| 3 | Luis | Torres | 2001-12-01 |
| 4 | Ana | Ramírez | 2002-01-20 |
| 5 | Pedro | Sánchez | 2000-06-11 |
| 6 | Lucía | Flores | 1998-03-22 |
| 7 | Carlos | Lopez | 2003-05-30 |
| 8 | Diana | Reyes | 2000-07-25 |
| 9 | Jorge | Castro | 1999-11-19 |
| 10 | Andrea | Mendoza | 2001-08-14 |

Mi consulta es entre las fechas 1998-2003

8. ORDER BY: Muestra todos los cursos ordenados alfabéticamente por su nombre.

```
SELECT *
FROM Cursos
ORDER BY NombreCurso ASC;
```

| | id_curso [PK] integer | nombrecurso character varying (50) | descripcion text | creditos integer | semestre integer | id_departamento integer | id_campus integer |
|----|--------------------------|---------------------------------------|-------------------------------------|---------------------|---------------------|----------------------------|----------------------|
| 1 | 2 | Álgebra Lineal | Matrices y vectores | 3 | 1 | 2 | 2 |
| 2 | 4 | Biología General | Estructura celular y genética | 3 | 1 | 4 | 4 |
| 3 | 3 | Física I | Cinemática y dinámica | 4 | 1 | 3 | 3 |
| 4 | 10 | Historia Universal | De la antigüedad a la modernidad | 3 | 2 | 10 | 10 |
| 5 | 6 | Literatura Española | Autores clásicos | 2 | 2 | 6 | 6 |
| 6 | 7 | Lógica | Fundamentos del pensamiento crítico | 2 | 2 | 7 | 7 |
| 7 | 8 | Macroeconomía | Economía a nivel nacional | 3 | 2 | 8 | 8 |
| 8 | 1 | Programación I | Introducción a la programación | 4 | 1 | 1 | 1 |
| 9 | 9 | Psicología del Desarrollo | Etapas del desarrollo humano | 3 | 2 | 9 | 9 |
| 10 | 5 | Química Orgánica | Estudio de compuestos orgánicos | 3 | 2 | 5 | 5 |

9. CTE: Crea una tabla de expresión común que liste el número de inscripciones por estudiante. Luego, consulta esta CTE para encontrar los 3 estudiantes con más inscripciones, mostrando el nombre del estudiante y el número de inscripciones.

```
WITH InscripcionesPorEstudiante AS (
    SELECT
        Estudiantes.id_estudiante,
        Estudiantes.nombre,
        Estudiantes.apellido,
        COUNT(Inscripciones.id_curso) AS TotalInscripciones
    FROM Estudiantes
    LEFT JOIN Inscripciones ON Estudiantes.id_estudiante = Inscripciones.id_estudiante
    GROUP BY Estudiantes.id_estudiante, Estudiantes.nombre, Estudiantes.apellido
)
SELECT
    nombre,
    apellido,
    TotalInscripciones
FROM InscripcionesPorEstudiante
ORDER BY TotalInscripciones DESC
LIMIT 3;
```

10.Consulta Compleja 1: Para cada departamento, muestra el nombre del departamento y el nombre del curso con la mayor cantidad de estudiantes inscritos.

```
WITH ConteoInscripciones AS (
    SELECT
        d.NombreDepartamento,
        c.id_curso,
        c.NombreCurso,
        COUNT(i.id_estudiante) AS TotalInscritos
    FROM Cursos c
    JOIN Departamentos d ON c.id_departamento = d.id_departamento
    LEFT JOIN Inscripciones i ON c.id_curso = i.id_curso
    GROUP BY d.NombreDepartamento, c.id_curso, c.NombreCurso
),
MaximosPorDepartamento AS (
    SELECT
        NombreDepartamento,
        MAX(TotalInscritos) AS MaxInscritos
    FROM ConteoInscripciones
    GROUP BY NombreDepartamento
)
SELECT
    C.NombreDepartamento,
    C.NombreCurso,
    C.TotalInscritos
FROM ConteoInscripciones C
JOIN MaximosPorDepartamento mpd
ON C.NombreDepartamento = mpd.NombreDepartamento
AND C.TotalInscritos = mpd.MaxInscritos;
```

| | nombreddepartamento character varying (100) 🔒 | nombrecurso character varying (50) 🔒 | totalinscritos bigint 🔒 |
|----|--|---|----------------------------|
| 1 | Física | Física I | 1 |
| 2 | Química | Química Orgánica | 1 |
| 3 | Psicología | Psicología del Desarrollo | 1 |
| 4 | Biología | Biología General | 1 |
| 5 | Literatura | Literatura Española | 1 |
| 6 | Matemáticas | Álgebra Lineal | 1 |
| 7 | Historia | Historia Universal | 1 |
| 8 | Filosofía | Lógica | 1 |
| 9 | Ingeniería Informática | Programación I | 1 |
| 10 | Economía | Macroeconomía | 1 |

11. Consulta Compleja 2: Encuentra a los profesores que imparten más de dos cursos, mostrando su nombre, apellido y la cantidad de cursos que imparten.

```

137 SELECT p.Nombre AS NombreProf,
138        p.Apellido AS ApellidoProf,
139        COUNT(cp.id_curso) AS TotalCursos
140 FROM Profesores p
141 JOIN CursosProfesores cp ON p.id_profesor = cp.id_profesor
142 GROUP BY p.id_profesor, p.Nombre, p.Apellido
143 HAVING COUNT(cp.id_curso) > 2;

```

Data Output Mensajes Notificaciones

| nombreprof | apellidoprof | totalcursos |
|------------------------|------------------------|-------------|
| character varying (50) | character varying (50) | bigint |

12.Consulta Compleja 3: Lista los nombres de los programas de estudio y, para cada programa, el nombre del curso con el promedio de calificación más alto.

```

WITH PromediosCursos AS (
    SELECT c.id_curso,
           c.NombreCurso,
           AVG(i.Calificacion) AS PromedioCalificacion
    FROM Cursos c
    JOIN Inscripciones i ON c.id_curso = i.id_curso
    GROUP BY c.id_curso, c.NombreCurso
),
CursosConPrograma AS (
    SELECT p.id_ProgramaEstudio,
           p.NombrePrograma,
           pc.id_curso,
           c.NombreCurso,
           c.PromedioCalificacion
    FROM ProgramasEstudio p
    JOIN ProgramasCursos pc ON p.id_ProgramaEstudio = pc.id_ProgramaEstudio
    JOIN PromediosCursos c ON pc.id_curso = c.id_curso
),
MaxPromediosPorPrograma AS (
    SELECT id_ProgramaEstudio,
           MAX(PromedioCalificacion) AS MaxPromedio
    FROM CursosConPrograma
    GROUP BY id_ProgramaEstudio
)

SELECT cp.NombrePrograma AS "Programa",
       cp.NombreCurso AS "Curso Destacado",
       cp.PromedioCalificacion AS "Promedio"
FROM CursosConPrograma cp
JOIN MaxPromediosPorPrograma m ON cp.id_ProgramaEstudio = m.id_ProgramaEstudio
AND cp.PromedioCalificacion = m.MaxPromedio
ORDER BY cp.NombrePrograma;

```

| | Programa character varying (100) 🔒 | Curso Destacado character varying (50) 🔒 | Promedio numeric 🔒 |
|----|---------------------------------------|---|-----------------------|
| 1 | Biociencias | Biología General | 18.0000000000000000 |
| 2 | Ciencias Físicas | Física I | 14.0000000000000000 |
| 3 | Economía Global | Macroeconomía | 17.5000000000000000 |
| 4 | Filosofía Clásica | Lógica | 13.5000000000000000 |
| 5 | Historia Contemporánea | Historia Universal | 14.9000000000000000 |
| 6 | Ingeniería Informática | Programación I | 15.5000000000000000 |
| 7 | Letras y Humanidades | Literatura Española | 19.0000000000000000 |
| 8 | Matemáticas Aplicadas | Álgebra Lineal | 17.2000000000000000 |
| 9 | Psicología Clínica | Psicología del Desarrollo | 18.3000000000000000 |
| 10 | Química Pura | Química Orgánica | 16.8000000000000000 |

Los datos, tablas y consultas están lo mejor especificados posibles, dando como resultado la administración de una institución académica universitaria.