

INSTITUTO TECNOLOGICO DE MEXICALI

Carrera:  
ING. en Sistemas.

Materia:  
Fundamento de base de datos.

Alumno:  
Marin Salazar Juan Sebastian 22490423.

Correo Institucional:  
a22490423@itmexicali.edu.mx

Profesor:  
Jose Ramon Bogarin Valenzuela

Fecha de entrega:  
10 de Abril del 2025

Mexicali, Baja California a 10 de Abril del 2025.

## Querys del profe

-- Crear tablas

```
CREATE TABLE estudiantes (  
  id SERIAL PRIMARY KEY,  
  nombre VARCHAR(100),  
  email VARCHAR(100),  
  fecha_nacimiento DATE  
);
```

```
CREATE TABLE cursos (  
  id SERIAL PRIMARY KEY,  
  nombre_curso VARCHAR(100),  
  duracion_meses INT  
);
```

```
CREATE TABLE matriculas (  
  id SERIAL PRIMARY KEY,  
  id_estudiante INT REFERENCES estudiantes(id),  
  id_curso INT REFERENCES cursos(id),  
  fecha_matricula DATE  
);
```

-- Insertar datos en estudiantes

```
INSERT INTO estudiantes (nombre, email, fecha_nacimiento) VALUES  
( 'Ana Torres', 'ana@example.com', '1998-03-12'),  
( 'Luis Gómez', 'luis@example.com', '2000-07-22'),  
( 'Carla Ruiz', 'carla@example.com', '1995-11-05');
```

-- Insertar datos en cursos

```
INSERT INTO cursos (nombre_curso, duracion_meses) VALUES  
( 'Bases de Datos', 4),  
( 'Programación Web', 6);
```

-- Insertar datos en matriculas

```
INSERT INTO matriculas (id_estudiante, id_curso, fecha_matricula) VALUES  
(1, 1, '2025-01-10'),  
(2, 1, '2025-01-12'),  
(3, 2, '2025-02-05'),  
(1, 2, '2025-02-10');
```

## **Problema a resolver: "Analítica y Gestión Académica"**

### **Contexto**

Una institución educativa quiere aprovechar su sistema de base de datos para obtener información útil sobre sus estudiantes, los cursos ofrecidos y las matrículas realizadas. Como analista de datos, se te solicita realizar una serie de tareas para mejorar la toma de decisiones académicas.

### **Parte 1: Verificación y Ajustes de Estructura (LDD)**

- **Verifica si la base de datos contiene una columna para almacenar el número de teléfono de los estudiantes. Si no existe, agrégala a la tabla estudiantes.**

R= ALTER TABLE Estudiantes ADD Telefono VARCHAR(14);

- **Modifica la tabla cursos para que el nombre del curso no pueda repetirse.**

R= ALTER TABLE Cursos ADD UNIQUE(nombre\_curso);

### **Parte 2: Carga y Ajuste de Datos (LMD)**

- **Actualiza el email de "Luis Gómez" a luisgomez@universidad.edu.**

R=  
UPDATE Estudiantes  
SET email = 'luisgomez@universidad.edu'  
WHERE nombre = 'Luis Gómez';

- **Registra una nueva matrícula para "Carla Ruiz" en el curso "Bases de Datos", con fecha 2025-04-01.**

R=  
INSERT INTO matriculas (id\_estudiante, id\_curso, fecha\_matricula) VALUES  
(3, 1, '2025-04-01');

- **Elimina la matrícula de "Ana Torres" del curso "Bases de Datos".**

R= DELETE FROM matriculas M WHERE M.id = 1;

### **Parte 3: Consultas Avanzadas (CLE)**

- **Muestra un listado con el nombre de cada estudiante, el nombre del curso al que está matriculado y la fecha de matrícula.**

R=

```
SELECT E.nombre, C.nombre_curso, M.fecha_matricula
FROM estudiantes E INNER JOIN matriculas M ON E.id = M.id_estudiante
INNER JOIN cursos C ON M.id_curso = C.id;
```

- **Muestra cuántos cursos ha tomado cada estudiante.**

R=

```
SELECT E.nombre, COUNT(M.id_estudiante) AS Numero_cursos
FROM estudiantes E INNER JOIN matriculas M ON E.id = M.id_estudiante
GROUP BY E.nombre;
```

- **Calcula la edad actual de cada estudiante y ordénalos de mayor a menor edad.**

R=

```
SELECT E.nombre, E.fecha_nacimiento ,
DATE_PART('year', AGE(CURRENT_DATE, E.fecha_nacimiento)) AS Edad
FROM estudiantes E INNER JOIN matriculas M ON E.id = M.id_estudiante
ORDER BY E.fecha_nacimiento ASC;
```

- **Muestra qué curso tiene más estudiantes matriculados.**

R=

```
SELECT C.id, C.nombre_curso , COUNT(M.id_estudiante) AS total_estudiantes
FROM estudiantes E INNER JOIN matriculas M ON E.id = M.id_estudiante
INNER JOIN cursos C ON M.id_curso = C.id GROUP BY C.id, C.nombre_curso
ORDER BY total_estudiantes DESC
LIMIT 1;
```

- **Calcula el porcentaje de estudiantes matriculados respecto al total de estudiantes para cada curso.**

R=

```
SELECT C.nombre_curso AS curso, COUNT(DISTINCT E.id) AS inscritos,
ROUND(COUNT(DISTINCT E.id) * 100.0 / (SELECT COUNT(*) FROM
estudiantes), 2) AS porcentaje
FROM estudiantes E INNER JOIN
matriculas M ON E.id = M.id_estudiante
INNER JOIN cursos C ON M.id_curso = C.id
GROUP BY C.nombre_curso;
```

# Resultados

QueryQuery History

1ALTER TABLE Estudiantes

2ADD Telefono VARCHAR(14);

3

Data OutputMessagesNotifications

ALTER TABLE

Query returned successfully in 118 msec.

QueryQuery History

1SELECT \* FROM estudiantes;

Data OutputMessagesNotifications

Showing rows: 1 to 3

	id [PK] integer	nombre character varying (100)	email character varying (100)	fecha_nacimiento date	telefono character varying (14)
1	1	Ana Torres	ana@example.com	1998-03-12	[null]
2	2	Luis Gómez	luis@example.com	2000-07-22	[null]
3	3	Carla Ruiz	carla@example.com	1995-11-05	[null]

QueryQuery History

1ALTER TABLE Cursos

2ADD UNIQUE(nombre\_curso);

3

Data OutputMessagesNotifications

ALTER TABLE

Query returned successfully in 142 msec.

QueryQuery History

1UPDATE Estudiantes

2SET email = 'luisgomez@universidad.edu'

3WHERE nombre = 'Luis Gómez';

Data OutputMessagesNotifications

UPDATE 1

Query returned successfully in 96 msec.

QueryQuery History

1SELECT \* FROM estudiantes;

Data OutputMessagesNotifications

Showing rows: 1 to 3

	id [PK] integer	nombre character varying (100)	email character varying (100)	fecha_nacimiento date	telefono character varying (14)
1	1	Ana Torres	ana@example.com	1998-03-12	[null]
2	3	Carla Ruiz	carla@example.com	1995-11-05	[null]
3	2	Luis Gómez	luisgomez@universidad.edu	2000-07-22	[null]

Query

Query History

1

▼

INSERT INTO matriculas (id\_estudiante, id\_curso, fecha\_matricula) VALUES

2

(3, 1, '2025-04-01');

Data Output

Messages

Notifications

INSERT 0 1

Query returned successfully in 82 msec.

Query

Query History

1

▼

SELECT \* FROM estudiantes E

2

INNER JOIN matriculas M ON E.id = M.id\_estudiante

3

INNER JOIN cursos C ON M.id\_curso = C.id;

Data Output

Messages

Notifications

Showing rows: 1 to 5													Page No: 1 of 1	
	id	nombre	email	fecha_nacimiento	telefono	id	id_estudiante	id_curso	fecha_matricula	id	nombre_curso	duracion_m		
	integer	character varying	character varying (100)	date	character varying (10)	integer	integer	integer	date	integer	character varying (100)	integer		
1	1	Ana Torres	ana@example.com	1998-03-12	[null]	1	1	1	2025-01-10	1	Bases de Datos	4		
2	2	Luis Gómez	luisgomez@universidad.edu	2000-07-22	[null]	2	2	1	2025-01-12	1	Bases de Datos	4		
3	3	Carla Ruiz	carla@example.com	1995-11-05	[null]	3	3	2	2025-02-05	2	Programación Web	6		
4	1	Ana Torres	ana@example.com	1998-03-12	[null]	4	1	2	2025-02-10	2	Programación Web	6		
5	3	Carla Ruiz	carla@example.com	1995-11-05	[null]	5	3	1	2025-04-01	1	Bases de Datos	4		

5

6

DELETE FROM matriculas M WHERE M.id = 1;

Data Output

Messages

Notifications

DELETE 1

Query returned successfully in 142 msec.

Showing rows: 1 to 4													Page No: 1 of 1	
	id	nombre	email	fecha_nacimiento	telefono	id	id_estudiante	id_curso	fecha_matricula	id	nombre_curso	duracion_m		
	integer	character varying	character varying (100)	date	character varying (10)	integer	integer	integer	date	integer	character varying (100)	integer		
1	2	Luis Gómez	luisgomez@universidad.edu	2000-07-22	[null]	2	2	1	2025-01-12	1	Bases de Datos	4		
2	3	Carla Ruiz	carla@example.com	1995-11-05	[null]	3	3	2	2025-02-05	2	Programación Web	6		
3	1	Ana Torres	ana@example.com	1998-03-12	[null]	4	1	2	2025-02-10	2	Programación Web	6		
4	3	Carla Ruiz	carla@example.com	1995-11-05	[null]	5	3	1	2025-04-01	1	Bases de Datos	4		

```

5
6 ✓ SELECT E.nombre, C.nombre_curso, M.fecha_matricula
7 FROM estudiantes E INNER JOIN matriculas M
8 ON E.id = M.id_estudiante
9 INNER JOIN cursos C ON M.id_curso = C.id;

```

Data Output Messages Notifications

	nombre character varying (100)	nombre_curso character varying (100)	fecha_matricula date
1	Luis Gómez	Bases de Datos	2025-01-12
2	Carla Ruiz	Programación Web	2025-02-05
3	Ana Torres	Programación Web	2025-02-10
4	Carla Ruiz	Bases de Datos	2025-04-01

```

5
6 ✓ SELECT E.nombre, C.nombre_curso, M.fecha_matricula
7 FROM estudiantes E INNER JOIN matriculas M
8 ON E.id = M.id_estudiante
9 INNER JOIN cursos C ON M.id_curso = C.id;

```

Data Output Messages Notifications

	nombre character varying (100)	nombre_curso character varying (100)	fecha_matricula date
1	Luis Gómez	Bases de Datos	2025-01-12
2	Carla Ruiz	Programación Web	2025-02-05
3	Ana Torres	Programación Web	2025-02-10
4	Carla Ruiz	Bases de Datos	2025-04-01

```

5
6 ✓ SELECT E.nombre, COUNT(M.id_estudiante) AS Numero_cursos
7 FROM estudiantes E
8 INNER JOIN matriculas M ON E.id = M.id_estudiante
9 GROUP BY E.nombre;

```

Data Output Messages Notifications

	nombre character varying (100)	numero_cursos bigint
1	Carla Ruiz	2
2	Luis Gómez	1
3	Ana Torres	1

```

6  SELECT E.nombre, E.fecha_nacimiento ,
7  DATE_PART('year', AGE(CURRENT_DATE, E.fecha_nacimiento)) AS Edad
8  FROM estudiantes E
9  INNER JOIN matriculas M ON E.id = M.id_estudiante
10 ORDER BY E.fecha_nacimiento ASC;

```

Data Output Messages Notifications

Showing

	nombre character varying (100)	fecha_nacimiento date	edad double precision
1	Carla Ruiz	1995-11-05	29
2	Carla Ruiz	1995-11-05	29
3	Ana Torres	1998-03-12	27
4	Luis Gómez	2000-07-22	24

```

13 SELECT C.id, C.nombre_curso , COUNT(M.id_estudiante) AS total_estudiantes
14 FROM estudiantes E
15 INNER JOIN matriculas M ON E.id = M.id_estudiante
16 INNER JOIN cursos C ON M.id_curso = C.id
17 GROUP BY C.id, C.nombre_curso
18 ORDER BY total_estudiantes DESC
19 LIMIT 1;
20

```

Data Output Messages Notifications

Showing rows: 1 to 1

	id [PK] integer	nombre_curso character varying (100)	totalEstudiantes bigint
1	1	Bases de Datos	2

```

29 SELECT C.nombre_curso AS curso, COUNT(DISTINCT E.id) AS inscritos,
30 ROUND(COUNT(DISTINCT E.id) * 100.0 / (SELECT COUNT(*) FROM estudiantes), 2) AS porcentaje
31 FROM estudiantes E
32 INNER JOIN matriculas M ON E.id = M.id_estudiante
33 INNER JOIN cursos C ON M.id_curso = C.id
34 GROUP BY C.nombre_curso;
35
36
37

```

Data Output Messages Notifications

Showing rows: 1 to 2 Page No: 1

	curso character varying (100)	inscritos bigint	porcentaje numeric
1	Bases de Datos	2	66.67
2	Programación Web	2	66.67