



TECNOLÓGICO
NACIONAL DE MÉXICO



INSTITUTO TECNOLÓGICO DE MEXICALI
INGENIERÍA EN SISTEMAS COMPUTACIONALES
“FUNDAMENTOS EN BASES DE DATOS”
TAREA 1 UNIDAD 3

ALUMNO: QUIÑONEZ MORFIN ELIAN

MATRICULA: 23490368

SEMESTRE: 4

MAESTRO: JOSÉ RAMON BOGARIN VALENZUELA



TECNOLOGICO
NACIONAL DE MÉXICO



Código Maestro

-- 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

);



TECNOLÓGICO
NACIONAL DE MÉXICO



-- 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');
```

-- Consultas CLE



-- Estudiantes matriculados en "Bases de Datos"

```
SELECT e.nombre  
FROM estudiantes e  
JOIN matriculas m ON e.id = m.id_estudiante  
JOIN cursos c ON c.id = m.id_curso  
WHERE c.nombre_curso = 'Bases de Datos';
```

-- Cursos con cantidad de estudiantes matriculados

```
SELECT c.nombre_curso, COUNT(m.id_estudiante) AS  
total_estudiantes  
FROM cursos c  
LEFT JOIN matriculas m ON c.id = m.id_curso  
GROUP BY c.nombre_curso;
```

-- Estudiantes mayores de 25 años

```
SELECT nombre, fecha_nacimiento,  
DATE_PART('year', AGE(fecha_nacimiento)) AS edad  
FROM estudiantes  
WHERE DATE_PART('year', AGE(fecha_nacimiento)) > 25;
```



TECNOLÓGICO
NACIONAL DE MÉXICO



-- Edad promedio de los estudiantes

```
SELECT ROUND(AVG(DATE_PART('year',  
AGE(fecha_nacimiento)))) AS edad_promedio  
FROM estudiantes;
```

-- Estudiantes ordenados por fecha de nacimiento

```
SELECT nombre, fecha_nacimiento  
FROM estudiantes  
ORDER BY fecha_nacimiento ASC;
```



TECNOLÓGICO
NACIONAL DE MÉXICO



Querys Mios

--Paso 1: Verificación y Ajustes de Estructura (LDD)

--1. 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.

ALTER TABLE estudiantes

ADD COLUMN telefono VARCHAR(15);

	id	nombre	email	fecha_nacimiento	telefono
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>

--2. Modifica la tabla cursos para que el nombre del curso no pueda repetirse

ALTER TABLE cursos

ADD CONSTRAINT nombre_curso_unico UNIQUE

(nombre_curso);

[23505] ERROR: llave duplicada viola restricción de unicidad «nombre_curso_unico»
Detail: Ya existe la llave (nombre_curso)=(Bases de Datos).



--Parte 2: Carga y Ajuste de Datos (LMD)

--1. Actualiza el email de Luis Gomez

UPDATE estudiantes

SET email='luisgomez@universidad.edu'

WHERE nombre='Luis Gómez';

id	nombre	email	fecha_nacimiento	telefono
1	Ana Torres	ana@example.com	1998-03-12	<null>
2	Carla Ruiz	carla@example.com	1995-11-05	<null>
3	Luis Gómez	luisgomez@universidad.edu	2000-07-22	<null>

--2. Registra Carla Ruiz en Base de Datos en 2025-04-01

INSERT INTO matriculas (id_estudiante, id_curso,

fecha_matricula) VALUES

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

id	id_estudiante	id_curso	fecha_matricula
1	2	2	2025-01-12
2	3	3	2025-02-05
3	4	1	2025-02-10
4	5	3	2025-04-01

--3. Eliminar matricula 'Ana Torres' del curso 'Base de datos'

DELETE FROM matriculas

WHERE id_estudiante=1 AND id_curso=1;

nombre
Luis Gómez
Carla Ruiz



--Parte 3: Consultas Avanzadas (CLE)

--1. Muestra un listado con el nombre de cada estudiante,
--el nombre del curso al que está matriculado y la fecha de
matrícula.

```
SELECT
    e.nombre AS estudiante,
    c.nombre_curso AS curso,
    m.fecha_matricula AS 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;
```

	estudiante	curso	fecha_matricula
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



--2. Muestra cuántos cursos ha tomado cada estudiante.

```
SELECT
    e.nombre AS estudiante,
    COUNT(m.id_curso) AS total_cursos
FROM
    estudiantes e
INNER JOIN
    matriculas m ON e.id = m.id_estudiante
GROUP BY
    e.nombre;
```

	estudiante ▾	total_cursos ▾
1	Carla Ruiz	2
2	Luis Gómez	1
3	Ana Torres	1

--3. Calcula la edad actual de cada estudiante y ordénalos de mayor a menor edad.

```
SELECT
    nombre AS estudiante,
    DATE_PART('year', AGE(fecha_nacimiento)) AS edad
FROM
    estudiantes
ORDER BY
    edad DESC;
```

	estudiante ▾	edad ▾
1	Carla Ruiz	29
2	Ana Torres	27
3	Luis Gómez	24



--4. Muestra qué curso tiene más estudiantes matriculados.

```
SELECT
  c.nombre_curso AS curso,
  COUNT(m.id_estudiante) AS total_estudiantes
FROM
  cursos c
  LEFT JOIN
  matriculas m ON c.id = m.id_curso
GROUP BY
  c.nombre_curso
ORDER BY
  total_estudiantes DESC
LIMIT 1;
```

	curso	total_estudiantes
1	Bases de Datos	2



--5. Calcula el porcentaje de estudiantes matriculados
--respecto al total de estudiantes para cada curso

```
SELECT
  c.nombre_curso AS curso,
  (COUNT(m.id_estudiante) * 100.0) / (SELECT COUNT(*) FROM
  estudiantes) AS porcentaje_matriculados
FROM
  cursos c
  LEFT JOIN
  matriculas m ON c.id = m.id_curso
GROUP BY
  c.nombre_curso;
```

	curso	porcentaje_matriculados
1	Base de Datos	0
2	Bases de Datos	66.66666666666667
3	Programación Web	66.66666666666667
4	Estructura de Datos	0