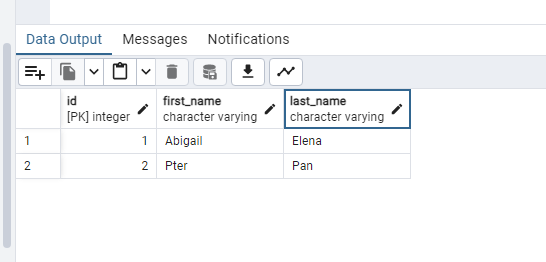
**LAB 12 - POSTGRESQL**

**ABIGAIL ELENA**

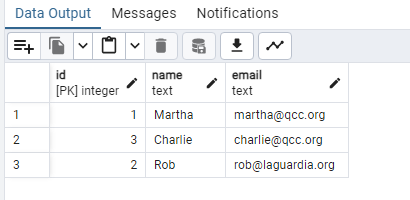
**TUESDAY, APRIL 29, 2025(INTRO)**

**CLASS EXAMPLE 12A**



**THURSDAY MAY 1, 2025**

**CLASS EXERCISE 12B**



**LAB EXERCISE 12B**

CREATE TABLE books (

id SERIAL PRIMARY KEY,

title VARCHAR(100) NOT NULL,

author VARCHAR(100) NOT NULL,

year\_published INT

);

INSERT INTO books (title, author, year\_published)

VALUES

('The Great Gatsby', 'F. Scott Fitzgerald', 1925),

('Animal Farm', 'George Orwell', 1945),

('To Kill a Mockingbird', 'Harper Lee', 1960),

('Brave New World', 'Aldous Huxley', 1932);

SELECT \* FROM books

WHERE author = 'George Orwell';

UPDATE books

SET year\_published = 1931

WHERE title = 'Brave New World';

DELETE FROM books

WHERE title = 'To Kill a Mockingbird';

SELECT \* FROM books;

BONUS CHALLENGE

ALTER TABLE books

ADD COLUMN genre VARCHAR(50);

UPDATE books

SET genre = 'Classic'

WHERE title = 'The Great Gatsby';

UPDATE books

SET genre = 'Dystopian'

WHERE title = 'Animal Farm';

UPDATE books

SET genre = 'Science Fiction'

WHERE title = 'Brave New World';

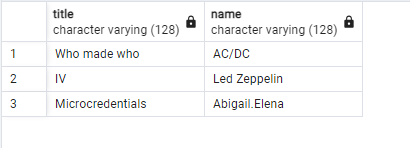
SELECT genre, COUNT(\*) AS book\_count

FROM books

GROUP BY genre;

**MAY 2 2025**

**CLASS EXAMPLE 12C**



**LAB EXERCISE 12C**

-- Create the students table

CREATE TABLE students (

id SERIAL PRIMARY KEY,

name VARCHAR(100),

major VARCHAR(100)

);

-- Create the courses table

CREATE TABLE courses (

id SERIAL PRIMARY KEY,

title VARCHAR(100),

department VARCHAR(100)

);

-- Create the enrollments table

CREATE TABLE enrollments (

id SERIAL PRIMARY KEY,

grade CHAR(2),

student\_id INT REFERENCES students(id),

courses\_id INT REFERENCES courses(id)

);

INSERT INTO students (name, major)

VALUES

('Alice Brown', 'Computer Science'),

('Peter Pan', 'Mathematics'),

('Annie Chen', 'Physics'),

('Abigail Elena', 'Music');

INSERT INTO courses (title, department)

VALUES

('Database Systems', 'Computer Science'),

('Linear Algebra', 'Mathematics'),

('Quantum Mechanics', 'Physics'),

('Python Programming', 'Computer Science'),

('Calculus III', 'Mathematics');

-- Replace with actual student IDs from SELECT \* FROM students if needed

INSERT INTO enrollments (grade, student\_id, courses\_id)

VALUES

('B+', 1, 1), -- Alice Brown -> Database Systems

('C', 2, 2), -- Peter Pan -> Linear Algebra

('B', 3, 3), -- Annie Chen -> Quantum Mechanics

('A', 4, 4); -- Abigail Elena -> Python Programming

SELECT name, major

FROM students;

SELECT s.name AS student\_name, c.title AS course\_title

FROM enrollments e

JOIN students s ON e.student\_id = s.id

JOIN courses c ON e.courses\_id = c.id;

SELECT s.name AS student\_name, c.title AS course\_title, e.grade

FROM enrollments e

JOIN students s ON e.student\_id = s.id

JOIN courses c ON e.courses\_id = c.id;