Week 1: Foundations of Data Retrieval

- **Starting MySQL and Creating Databases**
  - **Statrting MySQL:** mysql -u root;
  - **Showing Databases:** show databases;
  - **Create Databases:** create database [database_name];
  - **Using Database:** use database [database_name];
- **Setup Script**
  - **Part 1:** Run this script in your SQL client to create the `Customers` table.
  - **Code:**
  - CREATE TABLE Customers (
    customer_id INT PRIMARY KEY,
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    email VARCHAR(100) UNIQUE,
    city VARCHAR(50),
    join_date DATE
    );

- **Populate Data**
  - **Part 2:** Run this script to insert sample data into your new `Customers` table.
  - **Code:**
  - INSERT INTO Customers (customer_id, first_name, last_name, email, city, join_date) VALUES
    (1, 'John', 'Smith', 'john.smith@example.com', 'New York', '2024-01-15'),
    (2, 'Jane', 'Doe', 'jane.doe@example.com', 'Los Angeles', '2024-02-20'),
    (3, 'Peter', 'Jones', 'peter.jones@example.com', 'Chicago', '2024-03-10'),
    (4, 'Susan', 'Miller', 'susan.miller@example.com', 'New York', '2024-04-05'),
    (5, 'David', 'Scott', 'david.scott@example.com', 'Houston', '2024-05-21'),
    (6, 'Mary', 'Adams', 'mary.adams@example.com', 'Phoenix', '2024-06-11'),
    (7, 'Michael', 'Stewart', 'michael.stewart@example.com', 'Philadelphia', '2024-07-29'),
    (8, 'Sarah', 'Lee', 'sarah.lee@example.com', 'San Diego', '2024-08-01');

- **In-Class Example:** We'll work with the `Products (product_id, product_name, product_price, category_id)` table.
  - **Goal:** Learn to select, filter, and sort data.
  - `SELECT product_name, product_price FROM Products;` -- Get specific columns.
  - `SELECT * FROM Products WHERE category_id = 1;` -- Filter by category.
  - `SELECT * FROM Products WHERE product_price > 100.00 AND product_price < 500.00;` -- Filter with ranges.
  - `SELECT * FROM Products WHERE product_name LIKE '%Keyboard%';` -- Find products using wildcards.

- `SELECT * FROM Products ORDER BY price DESC;` -- Find the most expensive products.
- `SELECT * FROM Products ORDER BY price DESC LIMIT 10;` -- Find the 10 most expensive.