# 1. University Course Enrollment System

# Sql queries

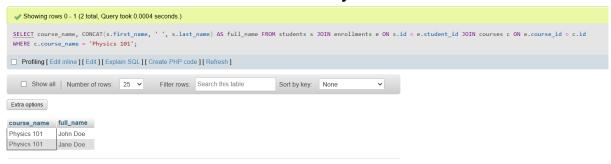
#### **Create Tables**

```
CREATE TABLE students (
id INT PRIMARY KEY AUTO_INCREMENT,
first_name VARCHAR(50),
last_name VARCHAR(50),
email VARCHAR(100),
enrollment date DATE
);
CREATE TABLE professors (
id INT PRIMARY KEY AUTO_INCREMENT,
first_name VARCHAR(50),
last_name VARCHAR(50),
department VARCHAR(100)
);
CREATE TABLE courses (
id INT PRIMARY KEY AUTO_INCREMENT,
course_name VARCHAR(100),
course_description TEXT,
professor_id INT,
FOREIGN KEY (professor_id) REFERENCES professors(id)
);
CREATE TABLE enrollments (
 student_id INT,
course_id INT,
 enrollment date DATE,
PRIMARY KEY (student_id, course_id),
FOREIGN KEY (student_id) REFERENCES students(id),
FOREIGN KEY (course_id) REFERENCES courses(id)
);
Insert Data
INSERT INTO students (first_name, last_name, email, enrollment_date)
VALUES
('John', 'Doe', 'john.doe@example.com', '2022-01-01'),
('Jane', 'Doe', 'jane.doe@example.com', '2022-01-05'),
('Bob', 'Smith', 'bob.smith@example.com', '2022-01-10'),
('Alice', 'Johnson', 'alice.johnson@example.com', '2022-01-15'),
('Mike', 'Williams', 'mike.williams@example.com', '2022-01-20');
```

```
INSERT INTO professors (first_name, last_name, department)
VALUES
('Emily', 'Chen', 'Physics'),
('David', 'Lee', 'Mathematics'),
('Sarah', 'Taylor', 'Computer Science'),
('Kevin', 'White', 'Biology');
INSERT INTO courses (course_name, course_description, professor_id)
VALUES
('Physics 101', 'Introduction to Physics', 1),
('Math 202', 'Calculus II', 2),
('CS 303', 'Data Structures', 3);
INSERT INTO enrollments (student_id, course_id, enrollment_date)
VALUES
(1, 1, '2022-01-05'),
(1, 2, '2022-01-10'),
(2, 1, '2022-01-15'),
(3, 3, '2022-02-01'),
(4, 2, '2022-02-05');
```

#### **Retrieve Data**

# Retrieve full names of students enrolled in "Physics 101"



#### Retrieve list of courses with professor's full name.



#### Retrieve courses with students enrolled.



# **Update Data**

```
## 1 row affected. (Query took 0.0005 seconds.)

UPDATE students SET email = 'john.doe2@example.com' WHERE id = 1;
```

#### **Delete Data**

# 2. Online Store inventory and Order System

# **Creating tables:**

```
CREATE TABLE products (
id INT AUTO_INCREMENT PRIMARY KEY,
product_name VARCHAR(255) NOT NULL,
price DECIMAL(10, 2) NOT NULL,
stock_quantity INT NOT NULL
);

CREATE TABLE customers (
id INT AUTO_INCREMENT PRIMARY KEY,
first_name VARCHAR(255) NOT NULL,
```

```
last_name VARCHAR(255) NOT NULL,
  email VARCHAR(255) UNIQUE NOT NULL
);
CREATE TABLE orders (
  id INT AUTO_INCREMENT PRIMARY KEY,
  customer_id INT,
  order_date DATE NOT NULL,
  FOREIGN KEY (customer_id) REFERENCES customers(id)
);
CREATE TABLE order_items (
  order_id INT,
  product_id INT,
  quantity INT NOT NULL,
  PRIMARY KEY (order_id, product_id),
  FOREIGN KEY (order_id) REFERENCES orders(id),
  FOREIGN KEY (product_id) REFERENCES products(id)
);
Inserting values:
-- Inserting Products
INSERT INTO products (product_name, price, stock_quantity) VALUES
('Laptop', 999.99, 10),
('Smartphone', 499.99, 20),
('Headphones', 199.99, 15),
('Smartwatch', 249.99, 30),
('Tablet', 299.99, 25);
-- Inserting Customers
INSERT INTO customers (first_name, last_name, email) VALUES
('John', 'Doe', 'john.doe@example.com'),
('Jane', 'Smith', 'jane.smith@example.com'),
('Alice', 'Johnson', 'alice.johnson@example.com'),
('Bob', 'Brown', 'bob.brown@example.com');
-- Inserting Orders
INSERT INTO orders (customer_id, order_date) VALUES
(1, '2024-10-01'),
(2, '2024-10-02'),
(3, '2024-10-03'),
(1, '2024-10-04'),
(4, '2024-10-05');
-- Inserting Order Items
INSERT INTO order_items (order_id, product_id, quantity) VALUES
(1, 1, 1), -- Order 1: 1 Laptop
(1, 2, 2), -- Order 1: 2 Smartphones
(2, 3, 1), -- Order 2: 1 Headphones
(2, 4, 1), -- Order 2: 1 Smartwatch
(3, 2, 1), -- Order 3: 1 Smartphone
(3, 5, 2), -- Order 3: 2 Tablets
(4, 1, 1), -- Order 4: 1 Laptop
```

```
(4, 3, 1), -- Order 4: 1 Headphones
(5, 4, 2), -- Order 5: 2 Smartwatches
(5, 5, 1); -- Order 5: 1 Tablet
```

# Retrieve the names and stock quantities of all products



# Retrieve the product names and quantities for one of the orders placed (e.g., Order ID 1)



# Retrieve all orders placed by a specific customer (e.g., Customer ID 1)



# **Update Data**

#### **Delete Data**