

## 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"

✓ Showing rows 0 - 1 (2 total, Query took 0.0004 seconds.)

```

SELECT course_name, CONCAT(s.first_name, ' ', s.last_name) AS full_name FROM students s JOIN enrollments e ON s.id = e.student_id JOIN courses c ON e.course_id = c.id
WHERE c.course_name = 'Physics 101';

```

☐ Profiling [\[ Edit inline \]](#) [\[ Edit \]](#) [\[ Explain SQL \]](#) [\[ Create PHP code \]](#) [\[ Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

course_name	full_name
Physics 101	John Doe
Physics 101	Jane Doe

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

✓ Showing rows 0 - 2 (3 total, Query took 0.0005 seconds.)

```

SELECT c.course_name, CONCAT(p.first_name, ' ', p.last_name) AS professor_name FROM courses c JOIN professors p ON c.professor_id = p.id;

```

☐ Profiling [\[ Edit inline \]](#) [\[ Edit \]](#) [\[ Explain SQL \]](#) [\[ Create PHP code \]](#) [\[ Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

course_name	professor_name
Physics 101	Emily Chen
Math 202	David Lee
CS 303	Sarah Taylor

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

## Retrieve courses with students enrolled.

✓ Showing rows 0 - 4 (5 total, Query took 0.0004 seconds.)

```
-- Retrieve courses with students enrolled SELECT c.course_name FROM courses c JOIN enrollments e ON c.id = e.course_id;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

☐ Show all | Number of rows: 25 ▼ | Filter rows: Search this table | Sort by key: None ▼

Extra options

course_name
Physics 101
Physics 101
Math 202
Math 202
CS 303

☐ Show all | Number of rows: 25 ▼ | Filter rows: Search this table | Sort by key: None ▼

## Update Data

✓ 1 row affected. (Query took 0.0005 seconds.)

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

## Delete Data

✓ 2 rows deleted. (Query took 0.0052 seconds.)

```
DELETE FROM order_items WHERE order_id = 2;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ 1 row deleted. (Query took 0.0004 seconds.)

```
DELETE FROM orders WHERE id = 2;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

## 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

✔ Showing rows 0 - 4 (5 total, Query took 0.0004 seconds.)

`SELECT product_name, stock_quantity FROM products;`

❑ Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	product_name	stock_quantity
<input type="checkbox"/> Edit Copy Delete	Laptop	10
<input type="checkbox"/> Edit Copy Delete	Smartphone	20
<input type="checkbox"/> Edit Copy Delete	Headphones	15
<input type="checkbox"/> Edit Copy Delete	Smartwatch	30
<input type="checkbox"/> Edit Copy Delete	Tablet	25

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

✔ Showing rows 0 - 1 (2 total, Query took 0.0004 seconds.)

`SELECT p.product_name, oi.quantity FROM order_items oi JOIN products p ON oi.product_id = p.id WHERE oi.order_id = 1;`

❑ Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

product_name	quantity
Laptop	1
Smartphone	2

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

✔ Showing rows 0 - 3 (4 total, Query took 0.0006 seconds.)

`SELECT o.id AS order_id, oi.product_id, oi.quantity FROM orders o JOIN order_items oi ON o.id = oi.order_id WHERE o.customer_id = 1;`

❑ Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

order_id	product_id	quantity
1	1	1
1	2	2
4	1	1
4	3	1

Update Data

✔ 2 rows affected. (Query took 0.0012 seconds.)

```
UPDATE products SET stock_quantity = stock_quantity - 1 WHERE id IN (SELECT product_id FROM order_items WHERE order_id = 1);
```

[\[ Edit inline \]](#) [\[ Edit \]](#) [\[ Create PHP code \]](#)

✔ 1 row affected. (Query took 0.0007 seconds.)

```
UPDATE products SET stock_quantity = stock_quantity - 2 WHERE id IN (SELECT product_id FROM order_items WHERE order_id = 1 AND product_id = 2);
```

[\[ Edit inline \]](#) [\[ Edit \]](#) [\[ Create PHP code \]](#)

## Delete Data

✔ 2 rows deleted. (Query took 0.0052 seconds.)

```
DELETE FROM order_items WHERE order_id = 2;
```

[\[ Edit inline \]](#) [\[ Edit \]](#) [\[ Create PHP code \]](#)

✔ 1 row deleted. (Query took 0.0004 seconds.)

```
DELETE FROM orders WHERE id = 2;
```

[\[ Edit inline \]](#) [\[ Edit \]](#) [\[ Create PHP code \]](#)