**MySQL Join Statements**

Stephan Peters

Colorado State University Global

23WC-ITS410-2: Database Management

Jamia Mills

4 February 2024

## MySQL Join Statements

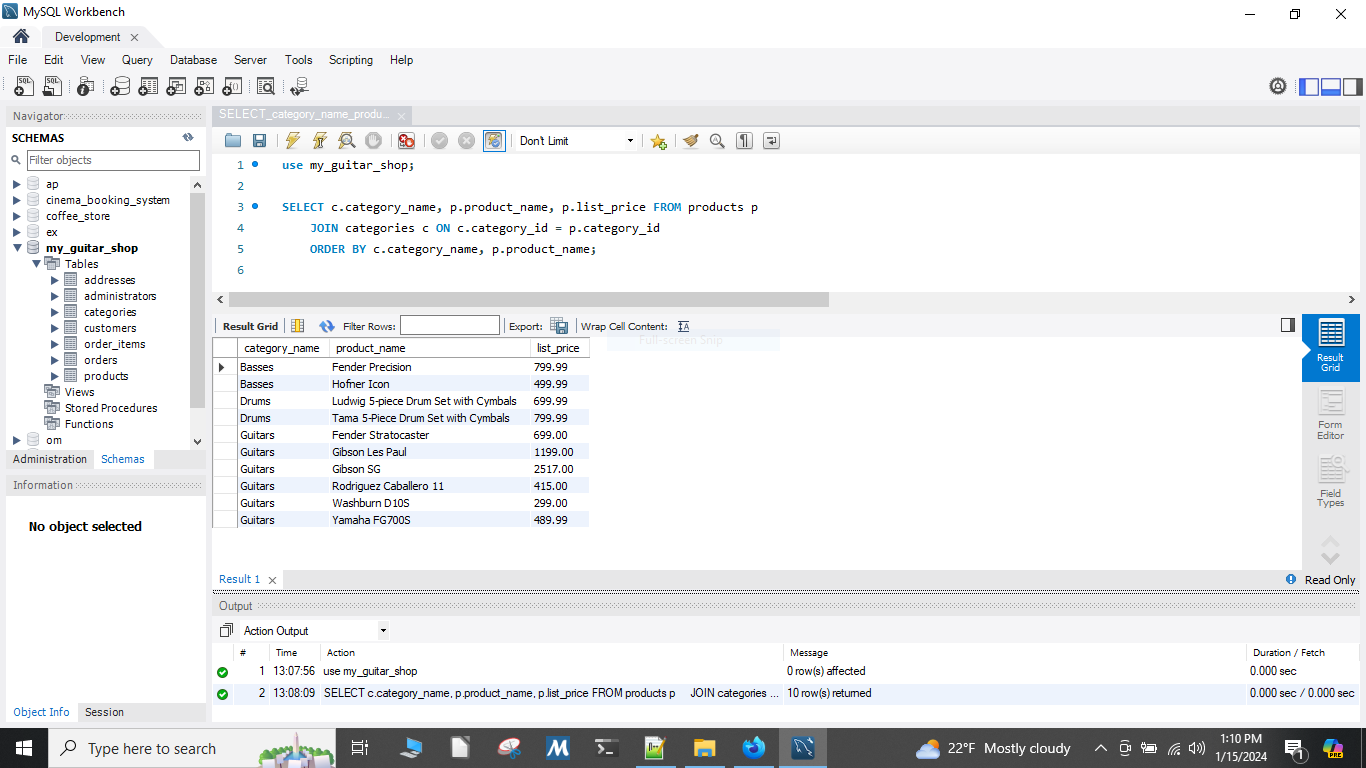
Full resolution images of the figures shown may be viewed with CSU Global Login via Google Drive here: <https://drive.google.com/drive/folders/1kGarJlj9qYitDCd8MCFmfbIUfJrOqpx9?usp=sharing>

## Join Categories and Products

This query returns category\_name, product\_name and list\_price from the categories and products table in my\_guitar\_shop, sorted by category\_name then product\_name.

Figure 1

SELECT c.category\_name, p.product\_name, p.list\_price FROM products p JOIN categories c ON c.category\_id = p.category\_id ORDER BY c.category\_name, p.product\_name;

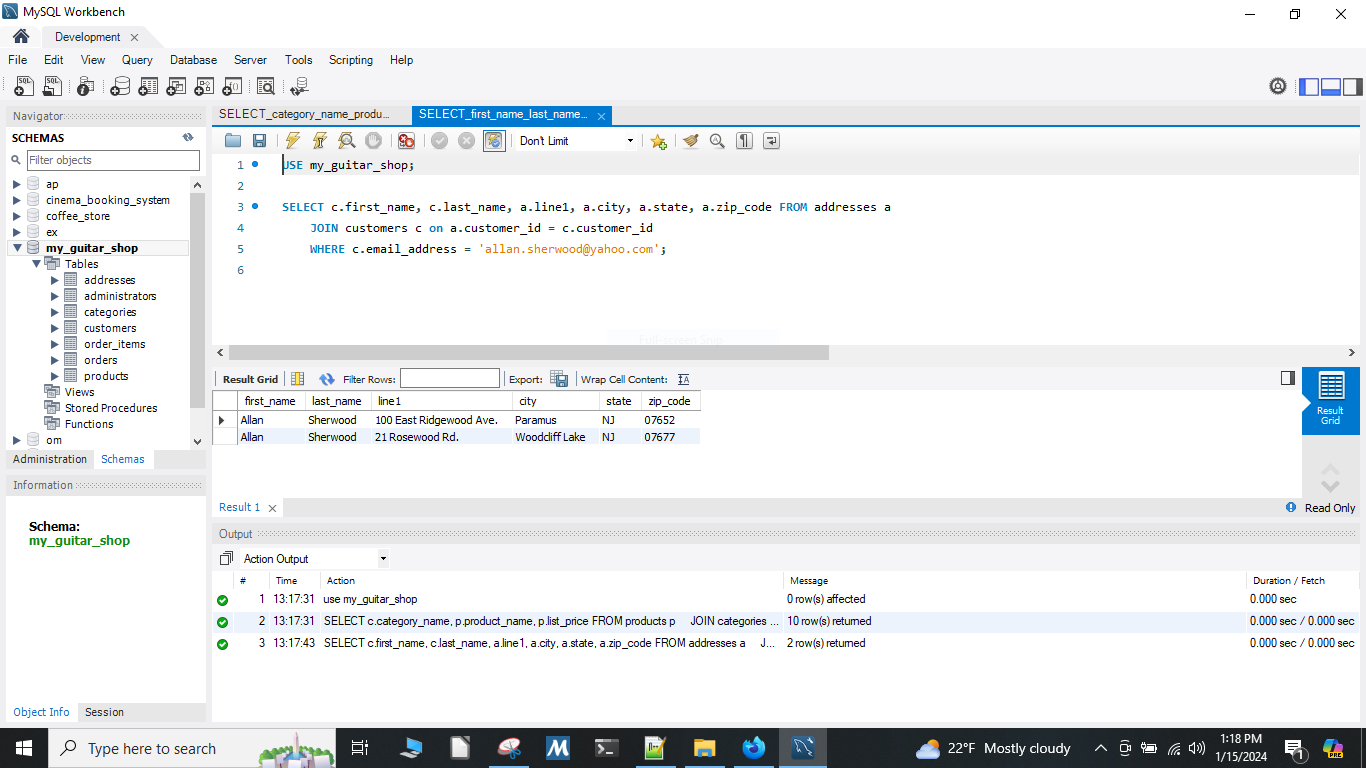
****

## Joining Tables to Retrieve Data

This query returns the customer’s name, shipping address and billing address (if different from shipping address) for a single customer email address.

Figure 2

SELECT c.first\_name, c.last\_name, a.line1, a.city, a.state, a.zip\_code FROM addresses a JOIN customers c on a.customer\_id = c.customer\_id WHERE c.email\_address = 'allan.sherwood@yahoo.com';

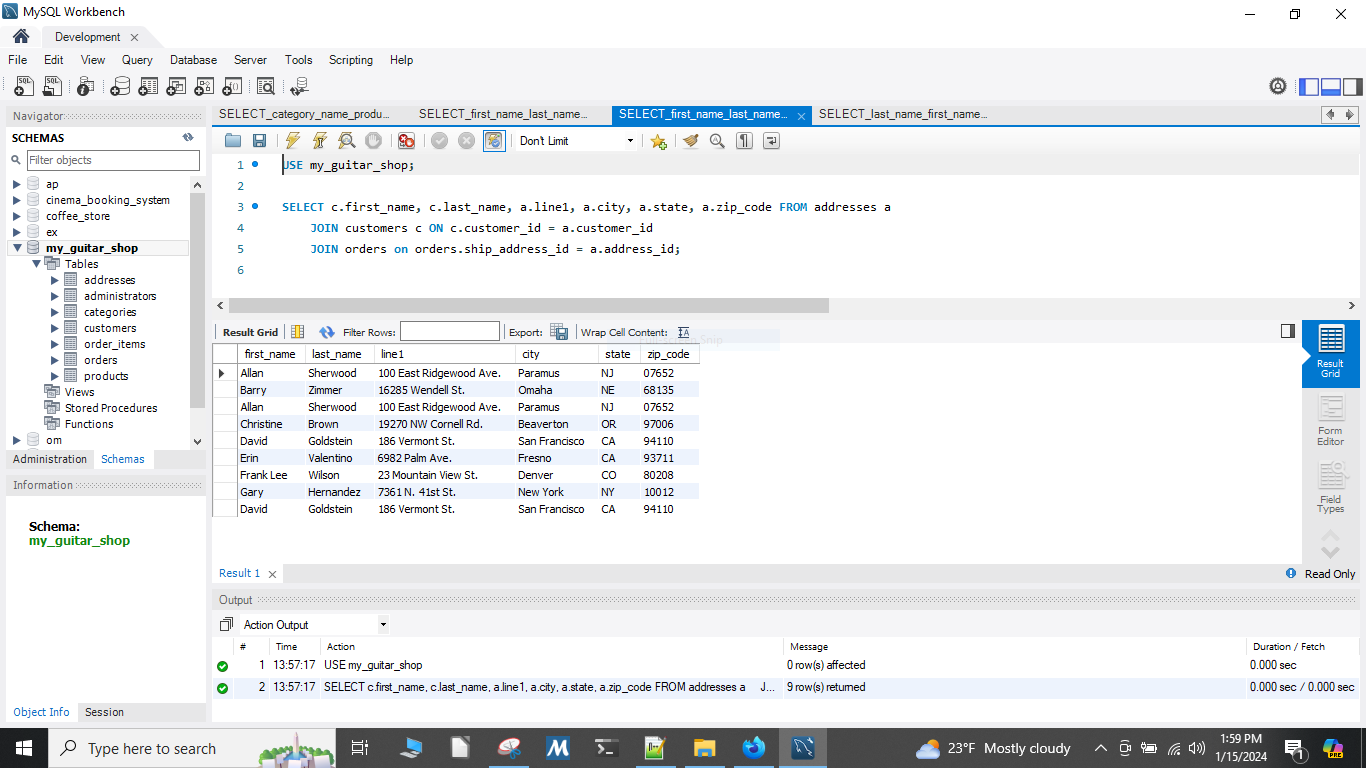
****

## Joining Tables to Retrieve Data from a Non-Normal Form Database

This query returns the customer’s name and shipping address but not the billing address from the customers and addresses tables. It uses a join to the orders table to force shipping address only.

Figure 3

SELECT c.first\_name, c.last\_name, a.line1, a.city, a.state, a.zip\_code FROM addresses a JOIN customers c ON c.customer\_id = a.customer\_id INNER JOIN orders on orders.ship\_address\_id = a.address\_id;

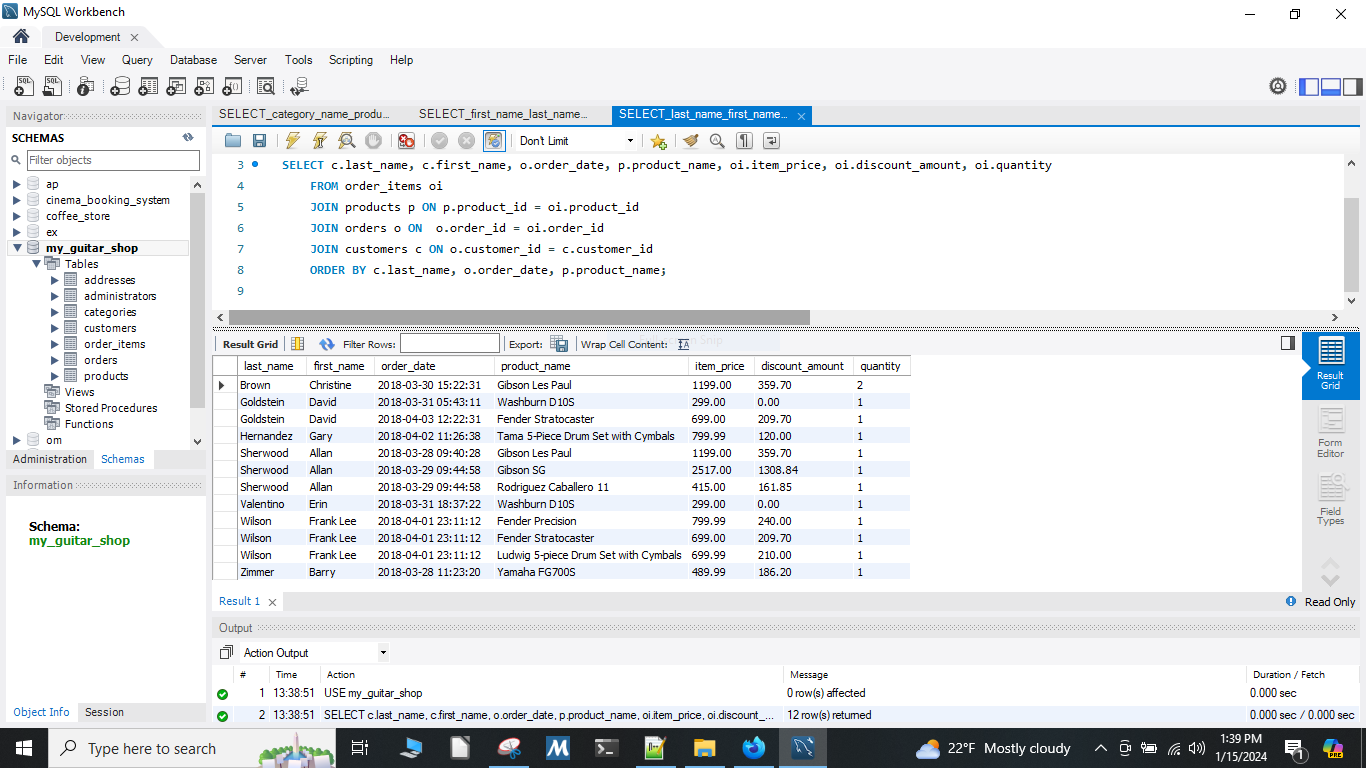
****

## Joining Multiple Tables

This query returns the customer’s last\_name, first\_name, the order’s order\_date, the product’s product\_name, and the order item’s item\_price, discount\_amount and quantity. It uses multiple joins to retrieve this information.

Figure 4

SELECT c.last\_name, c.first\_name, o.order\_date, p.product\_name, oi.item\_price, oi.discount\_amount, oi.quantity FROM order\_items oi JOIN products p ON p.product\_id = oi.product\_id JOIN orders o ON o.order\_id = oi.order\_id JOIN customers c ON o.customer\_id = c.customer\_id ORDER BY c.last\_name, o.order\_date, p.product\_name;

****