Database Management Class Analysis

Thang Sian Kop

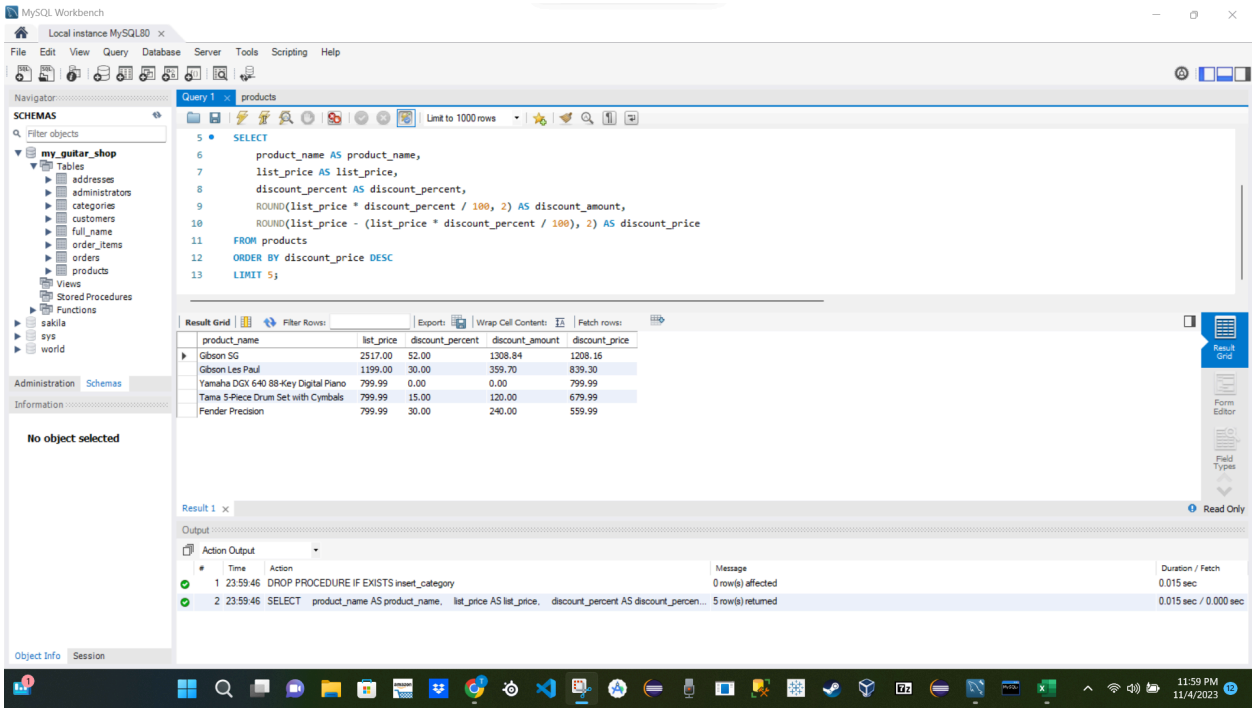
Database Management Class Analysis

In this database class, I have learned several fundamental lessons that have enhanced my understanding of database management and reporting. I will outline the key takeaways from each module and discuss how these lessons can be applied to improve database management and reporting.

Introduction

This module introduced me to the world of relational databases and how to work with MySQL. I learned how to install MySQL and MySQL Workbench, create databases, and load data. These basic skills lay the foundation for effective database management. Managing databases effectively requires setting up and maintaining the database environment, and this module provided the necessary skills to get started.

```
SELECT
    product_name AS product_name,
    list_price AS list_price,
    discount_percent AS discount_percent,
    ROUND(list_price * discount_percent / 100, 2) AS discount_amount,
    ROUND(list_price - (list_price * discount_percent / 100), 2) AS
discount_price
FROM products
ORDER BY discount_price DESC
LIMIT 5;
```

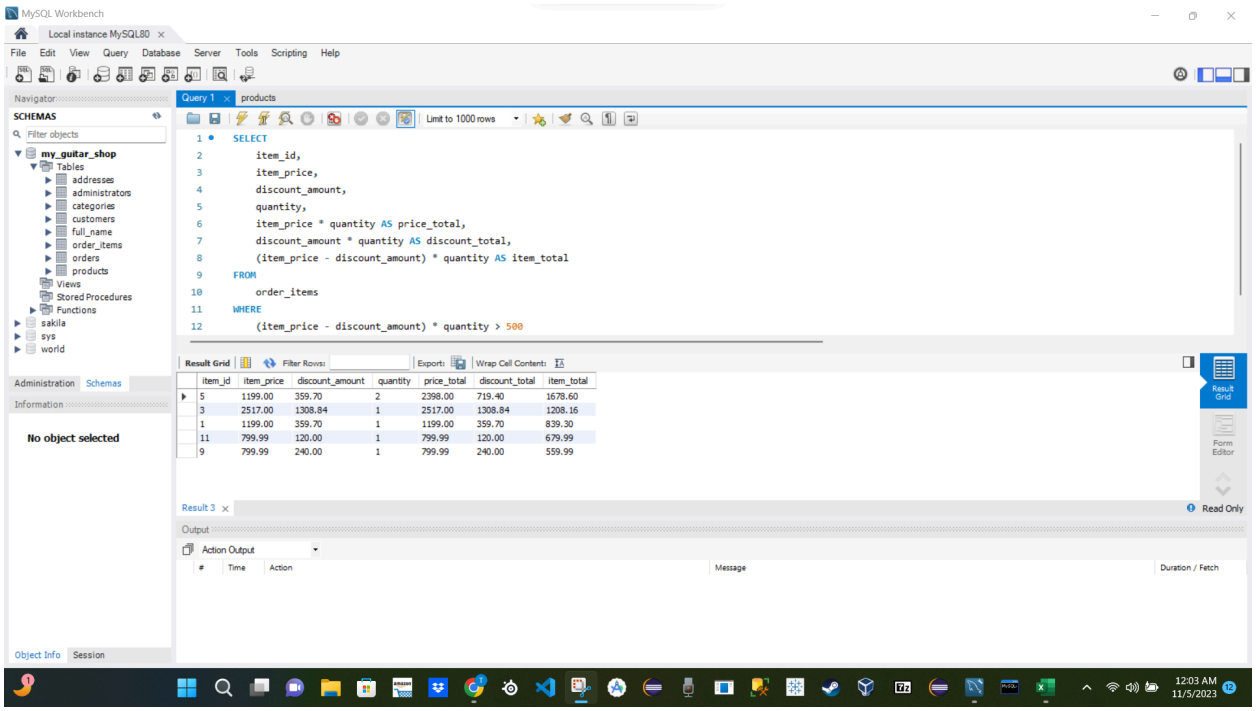


MySQL Data Types and Retrieve Data

In this module, I delved into MySQL data types and how to retrieve data from tables. I learned about numeric data types, date and time types, and string types. Understanding data types is crucial for designing a database schema that can handle the specific data requirements of an organization. The ability to retrieve data is essential for generating reports and extracting valuable insights from the database.

```
SELECT
    item_id,
    item_price,
    discount_amount,
    quantity,
    item_price * quantity AS price_total,
    discount_amount * quantity AS discount_total,
    (item_price - discount_amount) * quantity AS item_total
FROM
    order_items
```

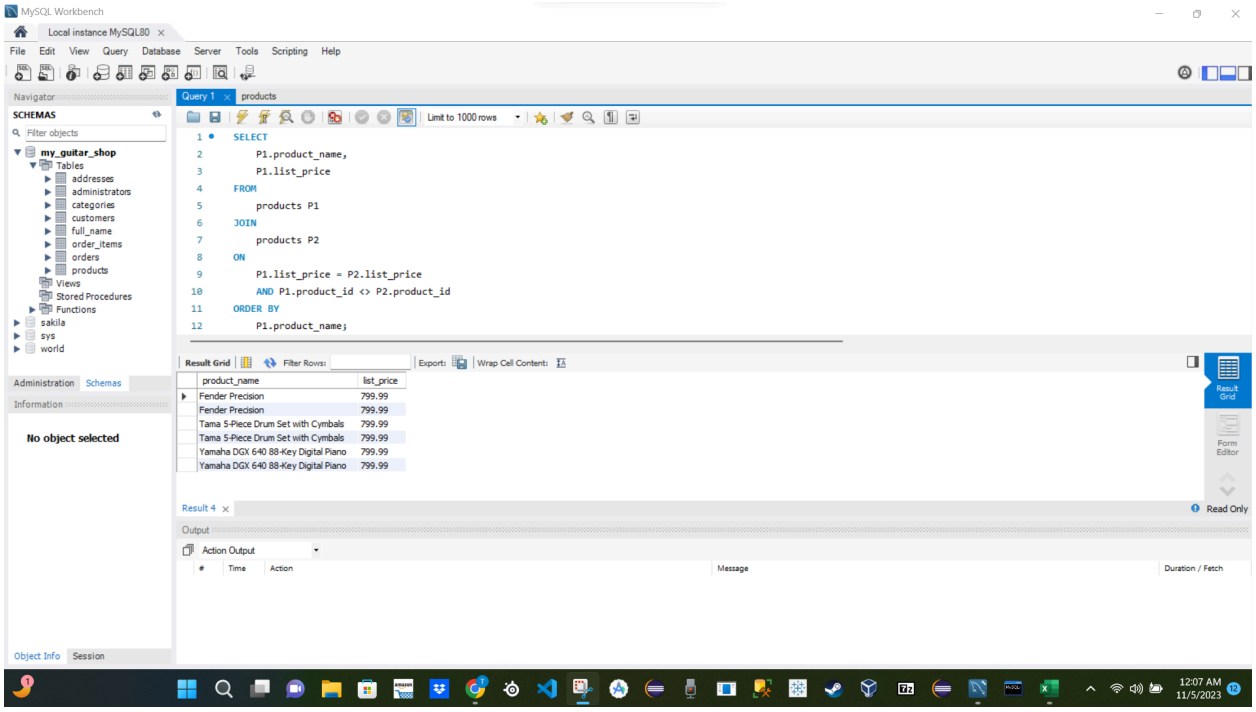
```
WHERE
    (item_price - discount_amount) * quantity > 500
ORDER BY
    item_total DESC;
```



Retrieve Data from Multiple Tables

Module 3 focused on retrieving data from multiple tables through various types of joins. I learned about table and column aliases, inner joins, left, right, cross, and self joins. This knowledge is vital for more complex reporting tasks where data from multiple tables needs to be combined. Effective database management involves optimizing these queries for better performance, especially in scenarios where large datasets are involved.

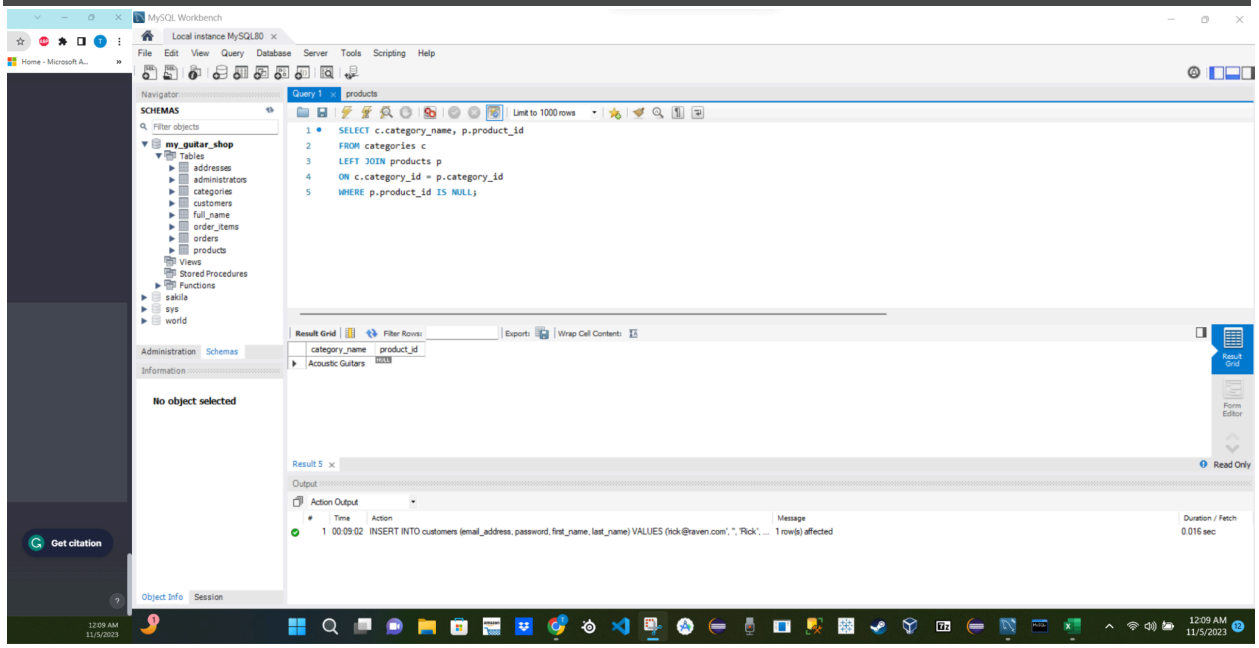
```
SELECT
    P1.product_name,
    P1.list_price
FROM
    products P1
JOIN
    products P2
ON
    P1.list_price = P2.list_price
    AND P1.product_id <> P2.product_id
ORDER BY
    P1.product_name;
```



Insert, Update, and Delete Tables

This module taught me how to create tables, insert new rows, and update existing rows in a database. These are fundamental operations for managing a database effectively. Knowing how to insert, update, and delete data is essential for maintaining data integrity and keeping the database up to date.

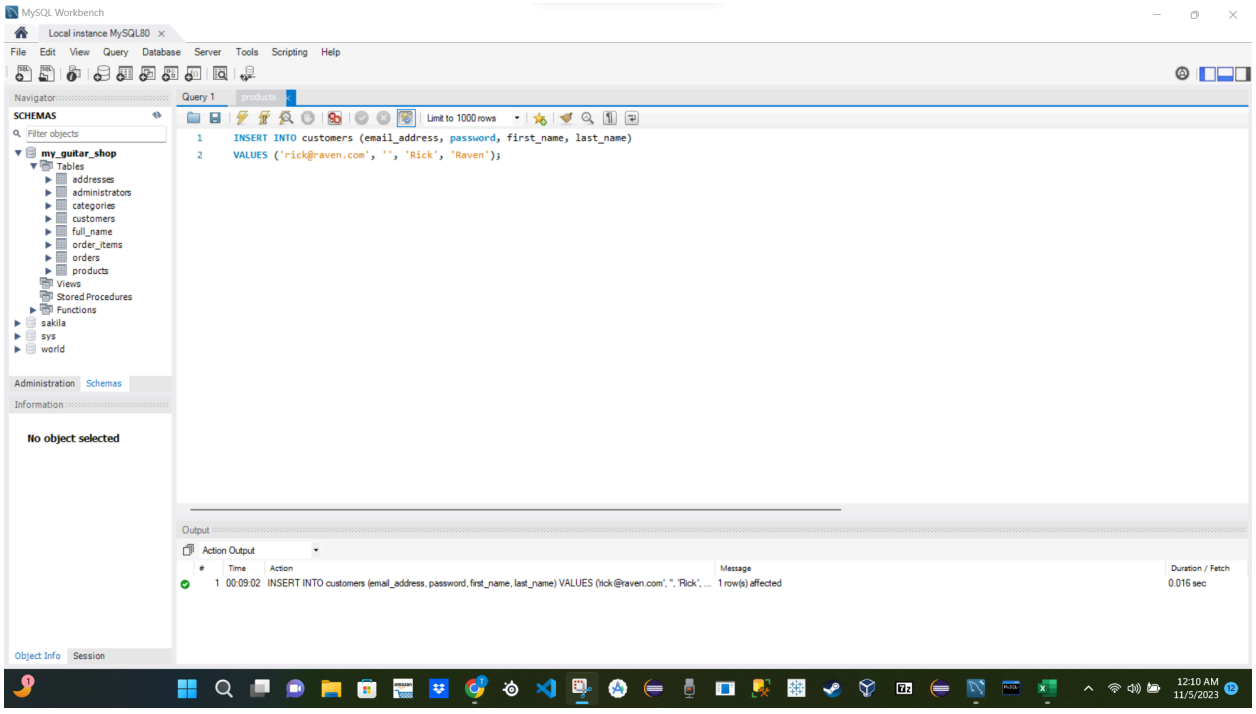
```
SELECT c.category_name, p.product_id
FROM categories c
LEFT JOIN products p
ON c.category_id = p.category_id
WHERE p.product_id IS NULL;
```



Grouping and Summarizing Data

Module 5 introduced me to the Group By and Having clauses, as well as union, intersect, and subqueries. These techniques are vital for summarizing and aggregating data in a database, which is crucial for reporting and analytics. Effective database management involves writing efficient queries to obtain meaningful insights from the data.

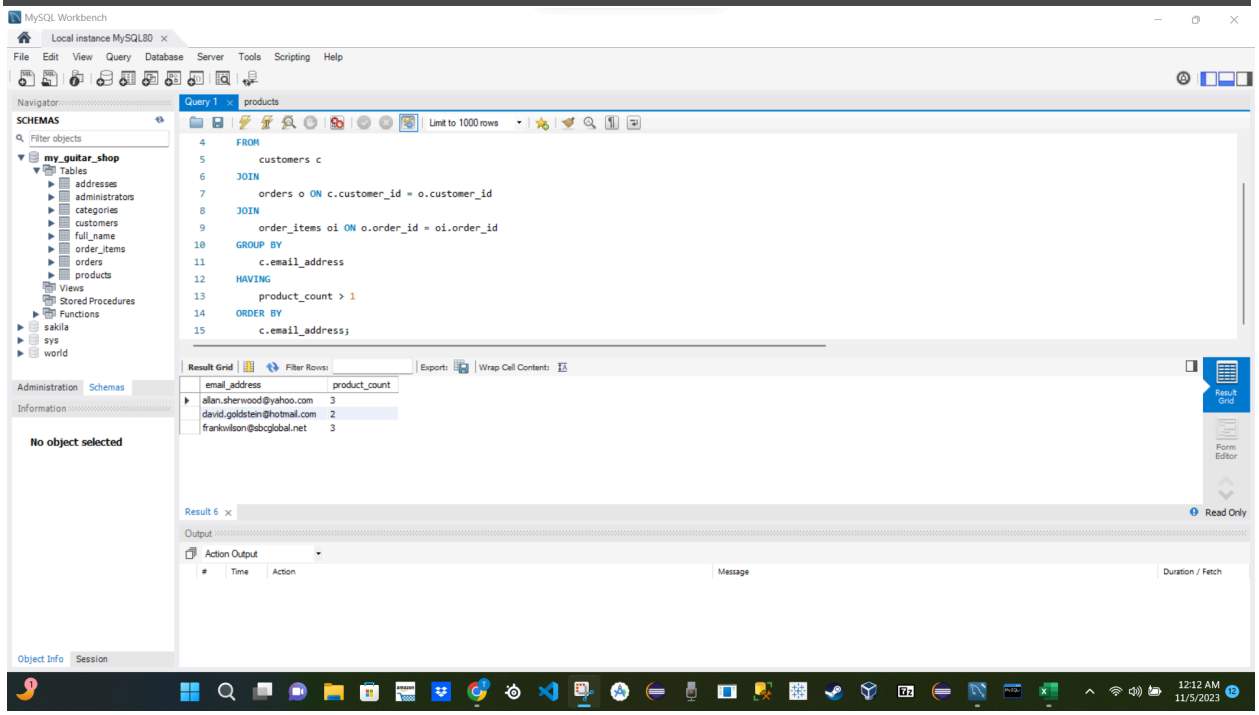
```
INSERT INTO customers (email_address, password, first_name,
last_name)
VALUES ('rick@raven.com', '', 'Rick', 'Raven');
```



Stored Procedures and Functions

In this module, I learned about stored procedures and MySQL functions. Stored procedures are valuable for automating tasks and providing an extra layer of security, while MySQL functions allow for custom data processing. Effective database management often includes the use of stored procedures to streamline common tasks and functions to perform complex data transformations.

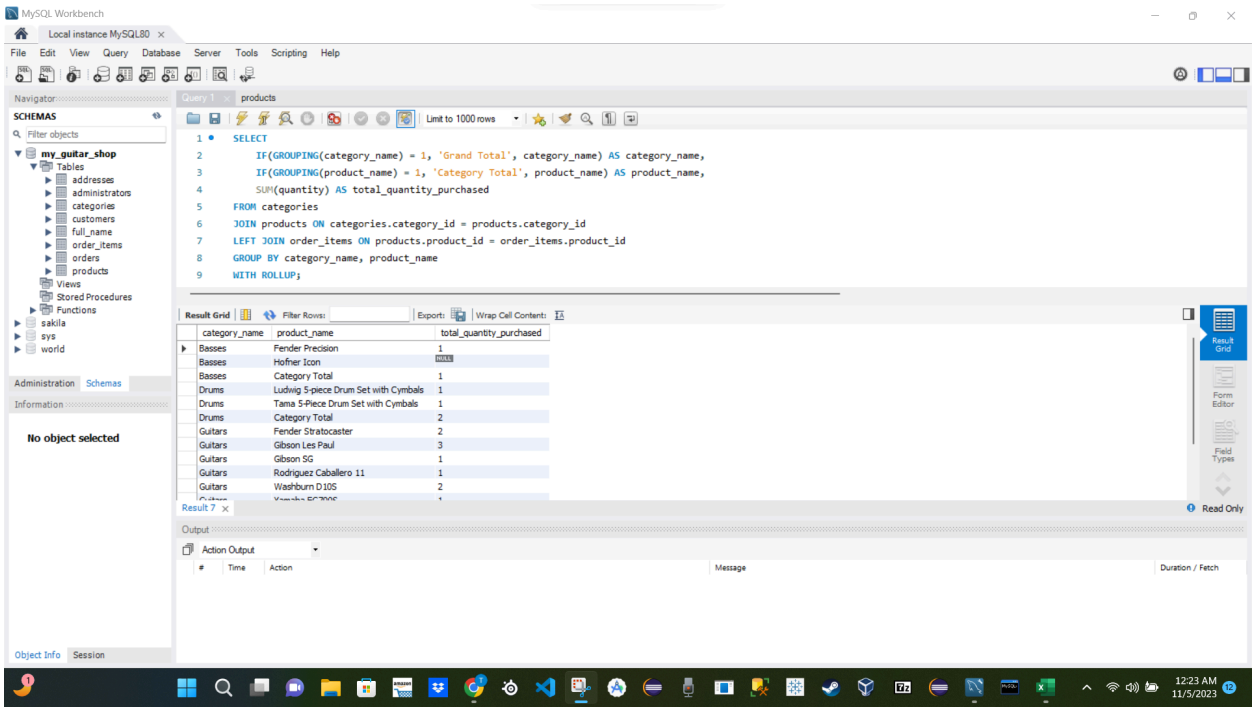
```
SELECT
    c.email_address AS email_address,
    COUNT(DISTINCT oi.product_id) AS product_count
FROM
    customers c
JOIN
    orders o ON c.customer_id = o.customer_id
JOIN
    order_items oi ON o.order_id = oi.order_id
GROUP BY c.email_address
HAVING product_count > 1
ORDER BY c.email_address;
```



Database Tuning and Security

Module 7 covered MySQL indexing, performance tuning, and best practices for database security. Tuning a database is crucial for optimizing its performance, especially when dealing with large datasets. Security is paramount in database management, as it protects sensitive information from unauthorized access. Applying these principles can lead to more efficient and secure database management.

```
SELECT
    IF(GROUPING(category_name) = 1, 'Grand Total', category_name) AS
category_name,
    IF(GROUPING(product_name) = 1, 'Category Total', product_name) AS
product_name,
    SUM(quantity) AS total_quantity_purchased
FROM categories
JOIN products ON categories.category_id = products.category_id
LEFT JOIN order_items ON products.product_id = order_items.product_id
GROUP BY category_name, product_name
WITH ROLLUP;
```



Backup and Recovery

This module discussed the importance of database backups and recovery procedures. Regular backups are a critical part of effective database management, ensuring that data can be restored in case of data loss or system failures. Understanding backup methods and strategies is essential for ensuring data availability and continuity.

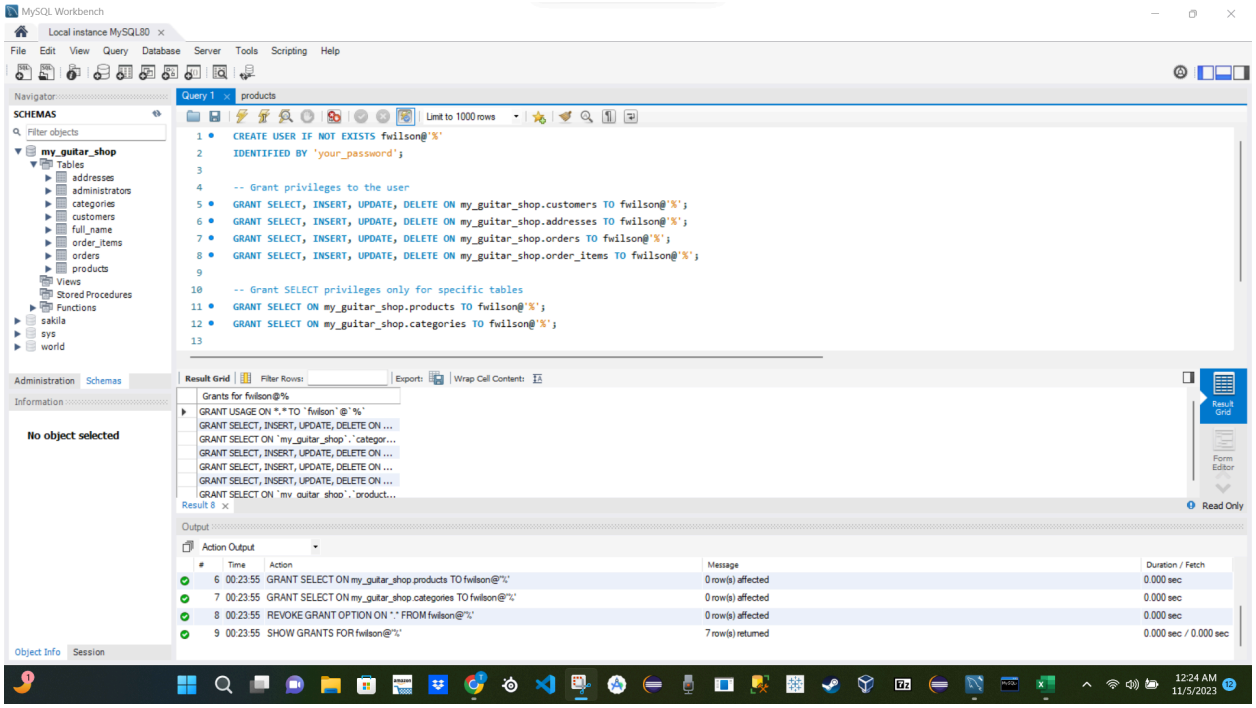
```
CREATE USER IF NOT EXISTS fwilson@'%'
IDENTIFIED BY 'your_password';

-- Grant privileges to the user
GRANT SELECT, INSERT, UPDATE, DELETE ON my_guitar_shop.customers TO
fwilson@'%';
GRANT SELECT, INSERT, UPDATE, DELETE ON my_guitar_shop.addresses TO
fwilson@'%';
GRANT SELECT, INSERT, UPDATE, DELETE ON my_guitar_shop.orders TO
fwilson@'%';
GRANT SELECT, INSERT, UPDATE, DELETE ON my_guitar_shop.order_items TO
fwilson@'%';

-- Grant SELECT privileges only for specific tables
GRANT SELECT ON my_guitar_shop.products TO fwilson@'%';
GRANT SELECT ON my_guitar_shop.categories TO fwilson@'%';

-- Revoke the right to grant privileges to other users
REVOKE GRANT OPTION ON *.* FROM fwilson@'%';

-- Show the privileges for the user
SHOW GRANTS FOR fwilson@'%';
```



The class has provided me with a solid foundation in database management and reporting. I have learned how to create and maintain databases, retrieve and manipulate data, and ensure data integrity and security. These lessons are invaluable for effective database management, which is essential for organizations to make informed decisions and drive their operations efficiently. The skills I have acquired in this class will serve as a solid basis for my future work in database management and reporting.