Experiment No.5
Perform simple queries, string manipulation operations and aggregate functions.
Date of Performance:
Date of Submission:

Aim :- Write simple query to manipulate string operations and perform aggregate functions like (MIN, MAX, SUM, AVERAGE, COUNT).

Objective :- To apply aggregate functions and string manipulation functions to perform simple queries in the database system

Theory:

Simple Queries in SQL:

In SQL, a simple query is a request for data from a database table or tables. It allows users to retrieve specific information by specifying the columns they want to retrieve and any conditions for filtering rows based on certain criteria. Simple queries are the backbone of interacting with databases, enabling users to extract the data they need for analysis, reporting, or further processing.

String Manipulation Operations:

String manipulation operations in SQL involve modifying or transforming string values stored in database columns. These operations are crucial for tasks such as formatting data, combining strings, converting case, or extracting substrings. By using string functions and operators, users can manipulate text data to suit their requirements, whether it's for display purposes or for further analysis.
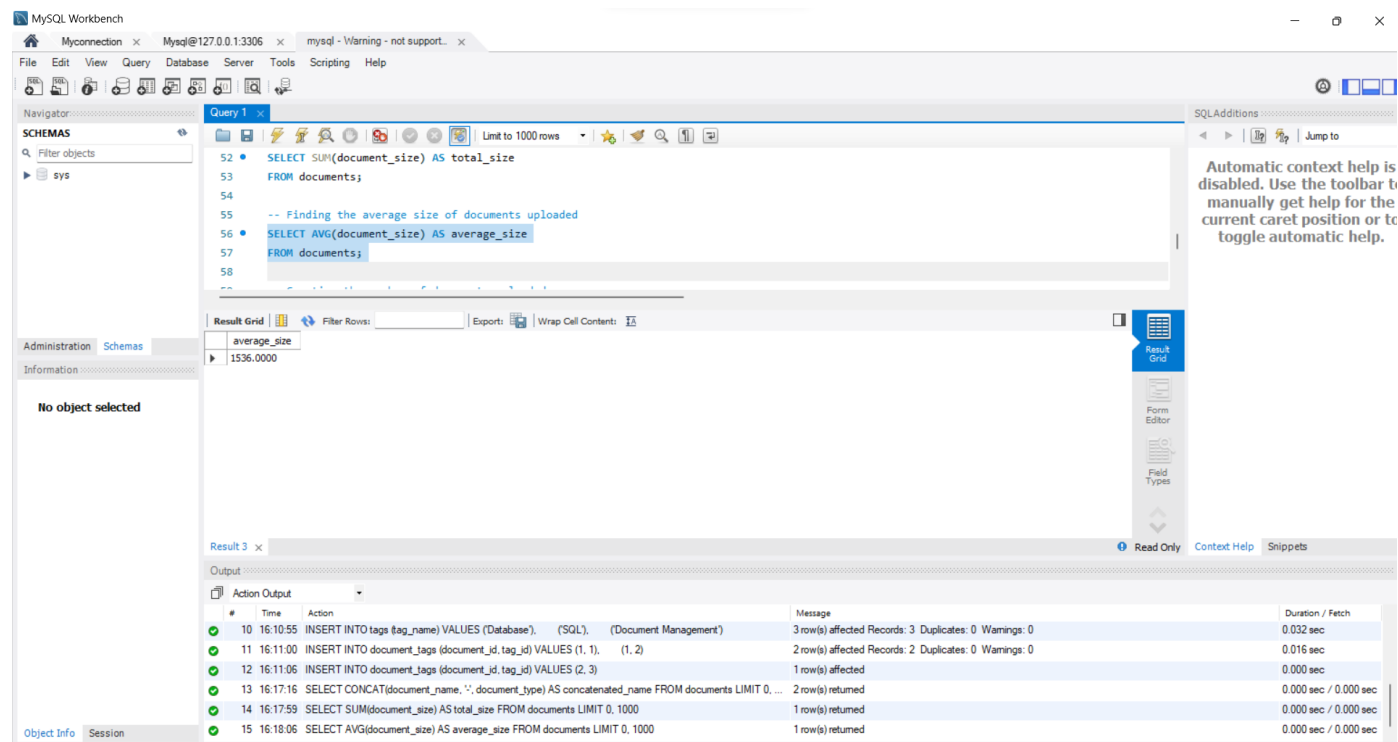
Aggregate Functions:

Aggregate functions in SQL are used to perform calculations on sets of values and return a single result. These functions allow users to summarize data across multiple rows, providing insights into the overall characteristics of the dataset. Common aggregate functions include calculating counts, sums, averages, minimums, and maximums of numerical values. They are essential tools for data analysis, enabling users to derive meaningful insights from large datasets.

Benefits of Understanding These Concepts:

- Data Retrieval: Simple queries allow users to fetch specific data from databases, facilitating data retrieval for various purposes.

- Data Transformation: String manipulation operations enable users to format and transform text data according to their needs, improving data consistency and readability.

- Data Analysis: Aggregate functions help users summarize and analyze large datasets, providing valuable insights into trends, patterns, and statistical measures.

- Data Reporting: By combining simple queries, string manipulation operations, and aggregate functions, users can generate reports and visualizations that communicate key findings effectively.

Implementation:

Conclusion:

1. Write syntax and explanation for each of the five aggregate functions
2. Show results of operations performed.

Sure, here's the syntax and explanation for each of the five aggregate functions, along with the results of the operations performed on the provided tables:

1. SUM():
   - Syntax:
     SELECT SUM(column_name) AS sum_column
     FROM table_name;
   - Explanation: SUM() function calculates the sum of all values in the specified column.
   - Example:
     -- Total size of all documents uploaded
     SELECT SUM(document_size) AS total_size
     FROM documents;
     Result: total_size = 3072 (assuming document sizes are 1024 and 2048)

2. AVG():
   - Syntax:
     SELECT AVG(column_name) AS avg_column
     FROM table_name;
   - Explanation: AVG() function calculates the average of all values in the specified column.
   - Example:
     -- Average size of documents uploaded
     SELECT AVG(document_size) AS average_size
     FROM documents;
     Result: average_size = 1536 (assuming document sizes are 1024 and 2048)

3. COUNT():
  - Syntax:
   SELECT COUNT(column_name) AS count_column
   FROM table_name;
  - Explanation: COUNT() function counts the number of non-null values in the specified column.
  - Example:
   -- Total number of documents uploaded
   SELECT COUNT(*) AS total_documents
   FROM documents;
   Result: total_documents = 2

4. MIN():
  - Syntax:
   SELECT MIN(column_name) AS min_column
   FROM table_name;
  - Explanation: MIN() function returns the smallest value in the specified column.
  - Example:
   -- Document with the smallest size
   SELECT MIN(document_size) AS smallest_size
   FROM documents;
   Result: smallest_size = 1024 (assuming document sizes are 1024 and 2048)

5. MAX():
  - Syntax:
   SELECT MAX(column_name) AS max_column
   FROM table_name;
  - Explanation: MAX() function returns the largest value in the specified column.
  - Example:
   -- Document with the largest size
   SELECT MAX(document_size) AS largest_size
   FROM documents;
   Result: largest_size = 2048 (assuming document sizes are 1024 and 2048)