PRACTICAL 5

Aim: To explore and implement GROUP functions like AVG, COUNT, MAX, MIN, and SUM in SQL.

Objectives:

To understand the use and implementation of GROUP functions in SQL.

To analyse data using aggregate functions to perform computations on data sets.

Introduction to GROUP Functions:

GROUP functions, also known as aggregate functions, operate on a set of rows and return a single value. They are necessary for performing complex calculations over a group of rows and for generating reports. These functions include:

1. AVG: Calculates the average value of a numeric column.
2. COUNT: Counts the rows in a specified column, including or excluding NULLs.
3. MAX: Returns the maximum value in a specified column.
4. MIN: Returns the minimum value in a specified column.
5. SUM: Calculates the total sum of a numeric column.

Create a table named 'Sales' to store sales data.

```
CREATE TABLE Sales (
SaleID int PRIMARY KEY,
Product varchar(255),
Quantity int,
Price decimal(10,2),
SaleDate date
);
```

Insert sample data into the 'Sales' table.

```
INSERT INTO Sales (SaleID, Product, Quantity, Price, SaleDate) VALUES
(1, 'Laptop', 1, 100000, '2023-01-01'),
(2, 'Laptop', 1, 50000, '2023-01-02'),
(3, 'Mouse', 10, 700, '2023-01-03'),
(4, 'Keyboard', 5, 2000, '2023-01-04');
```

Implement GROUP Functions:

1. **AVG (Average Price): Calculate the average selling price of laptops.**

- ***SYNTAX:*** *SELECT AVG(column_name) FROM table_name WHERE condition;*

```
SELECT AVG(Price) FROM Sales WHERE Product = 'Laptop';
```

2. **COUNT (Total Sales): Count the total number of sales for each product.**

SYNTAX: *SELECT COUNT(column_name) FROM table_name WHERE condition;*

```
SELECT Product, COUNT(*) AS TotalSales FROM Sales GROUP BY
Product;
```

3. **MAX (Maximum Price): Find the maximum price of any product sold.**

SYNTAX: *SELECT MAX(column_name) FROM table_name WHERE condition;*

```
SELECT MAX(Price) FROM Sales;
```

4. **MIN (Minimum Price): Determine the minimum price of any product sold.**
   SYNTAX: *SELECT MIN(column_name) FROM table_name WHERE condition;*

```
SELECT MIN(Price) FROM Sales;
```

5. **SUM (Total Revenue): Calculate the total revenue from all sales.**
   SYNTAX: *SELECT SUM(column_name) FROM table_name WHERE condition;*

```
SELECT SUM(Quantity * Price) FROM Sales;
```

Grouping Data with GROUP BY Clause:

SYNTAX: *SELECT column_name, AGG_FUNC(column_name)*

FROM table_name

WHERE condition

GROUP BY column_name;

```
SELECT Product, AVG(Price) FROM Sales GROUP BY Product;
```

Filtering Groups with HAVING Clause

SYNTAX: *SELECT column_name, AGG_FUNC(column_name)*

FROM table_name

WHERE condition

GROUP BY column_name

HAVING AGG_FUNC(column_name) condition;

```
SELECT Product, AVG(Price)
FROM Sales
GROUP BY Product
HAVING AVG(Price) > 100;
```

Using GROUP BY with multiple columns

```
SELECT Product, SaleDate
FROM Sales
GROUP BY Product, SaleDate;
```