**Implementing SQL Functions and Aggregate Functions**

**Step 1: Prepare for the Application**

You’ll create a small database and populate it with sample data using MySQL. This database will allow you to practice using SQL functions and aggregate functions.

* Open your MySQL environment or terminal.
* Create a new database called EmployeeDB using the following SQL command:

**CREATE** **DATABASE** EmployeeDB;

USE EmployeeDB;

* Create a table called **Employees** with the following schema:

**CREATE** **TABLE** Employees (

EmployeeID INT AUTO\_INCREMENT **PRIMARY** **KEY**,

FirstName VARCHAR(**50**),

LastName VARCHAR(**50**),

Department VARCHAR(**50**),

Salary DECIMAL(**10**, **2**),

HireDate DATE

);

* Insert sample data into the **Employees** table:

**INSERT** **INTO** Employees (FirstName, LastName, Department, Salary, HireDate) **VALUES**

('Liam', 'Nguyen', 'Engineering', **85000**.**00**, '2020-03-15'),

('Sophia', 'Smith', 'Marketing', **72000**.**00**, '2019-05-22'),

('Raj', 'Patel', 'Sales', **64000**.**00**, '2021-07-01'),

('Aisha', 'Khan', 'HR', **60000**.**00**, '2020-09-12'),

('Carlos', 'Martinez', 'Engineering', **93000**.**00**, '2018-12-01'),

('Chen', 'Zhao', 'Marketing', **77000**.**00**, '2017-11-05'),

('Amara', 'Okafor', 'Sales', **67000**.**00**, '2022-03-18');

**Step 2: Using SQL Functions for Data Manipulation**

Write queries that use basic SQL functions such as **CONCAT**, **UPPER**, **LOWER**, **LENGTH**, and **SUBSTRING** to manipulate data.

1. Write a query to concatenate the first and last names of employees into a single column called **FullName**.
2. Use the **UPPER** function to convert the **Department** column to uppercase.
3. Use the **LOWER** function to convert the **LastName** column to lowercase.
4. Use the **LENGTH** function to calculate the length of each employee's **FirstName**.
5. Use the **SUBSTRING** function to extract the first three characters of the **LastName** column.

**Step З: Using Aggregate Functions**

Write queries that use aggregate functions such as COUNT, SUM, AVG, MIN, and MAX to analyze data.

1. Count the total number of employees in the company.
2. Calculate the total salary expenditure for all employees.
3. Find the average salary of employees in the Engineering department.
4. Identify the minimum salary in the company.
5. Determine the maximum salary in the Sales department.

**Step 4: Combining Aggregate Functions with GROUP ВУ**

Write queries that group data by specific columns and apply aggregate functions.

1. Group employees by Department and calculate the total salary for each department.
2. Group employees by Department and find the average salary in each department.
3. Use the GROUP BY clause to count the number of employees in each department.

**Step 5: Exploring Advanced Functions**

Experiment with combinations of SQL functions and aggregate functions.

1. Concatenate FirstName and LastName, then calculate the length of the resulting full name.
2. Use a combination of COUNT and GROUP BY to determine how many employees were hired in each year (extract the year from the HireDate column).
3. Use SUM and GROUP BY to calculate the total salary expenditure per year of hiring.

**lab\_functions.sql:**

-- ============================================

-- Lab: Implementing SQL Functions and Aggregate Functions

-- Database: EmployeeDB

-- ============================================

-- Step 1. Prepare for the Application

-- --------------------------------------------

**CREATE** **DATABASE** **IF** **NOT** **EXISTS** EmployeeDB;

USE EmployeeDB;

-- Drop table if exists (clean start)

**DROP** **TABLE** **IF** **EXISTS** Employees;

-- Create Employees table

**CREATE** **TABLE** Employees (

EmployeeID INT AUTO\_INCREMENT **PRIMARY** **KEY**,

FirstName VARCHAR(**50**),

LastName VARCHAR(**50**),

Department VARCHAR(**50**),

Salary DECIMAL(**10**, **2**),

HireDate DATE

);

-- Insert sample data

**INSERT** **INTO** Employees (FirstName, LastName, Department, Salary, HireDate) **VALUES**

('Liam', 'Nguyen', 'Engineering', **85000**.**00**, '2020-03-15'),

('Sophia', 'Smith', 'Marketing', **72000**.**00**, '2019-05-22'),

('Raj', 'Patel', 'Sales', **64000**.**00**, '2021-07-01'),

('Aisha', 'Khan', 'HR', **60000**.**00**, '2020-09-12'),

('Carlos', 'Martinez', 'Engineering', **93000**.**00**, '2018-12-01'),

('Chen', 'Zhao', 'Marketing', **77000**.**00**, '2017-11-05'),

('Amara', 'Okafor', 'Sales', **67000**.**00**, '2022-03-18');

-- Verify data

**SELECT** \* **FROM** Employees;

-- ============================================

-- Step 2. Using SQL Functions for Data Manipulation

-- ============================================

-- 2.1 Concatenate first and last names into FullName

**SELECT** CONCAT(FirstName, ' ', LastName) **AS** FullName

**FROM** Employees;

-- 2.2 Convert Department to uppercase

**SELECT** **UPPER**(Department) **AS** UppercaseDepartment

**FROM** Employees;

-- 2.3 Convert LastName to lowercase

**SELECT** **LOWER**(LastName) **AS** LowercaseLastName

**FROM** Employees;

-- 2.4 Calculate length of each employee's FirstName

**SELECT** FirstName, **LENGTH**(FirstName) **AS** NameLength

**FROM** Employees;

-- 2.5 Extract first 3 characters of LastName

**SELECT** LastName, **SUBSTRING**(LastName, **1**, **3**) **AS** FirstThreeLetters

**FROM** Employees;

-- ============================================

-- Step 3. Using Aggregate Functions

-- ============================================

-- 3.1 Count total number of employees

**SELECT** **COUNT**(\*) **AS** TotalEmployees **FROM** Employees;

-- 3.2 Total salary expenditure

**SELECT** **SUM**(Salary) **AS** TotalSalaryExpenditure **FROM** Employees;

-- 3.3 Average salary of Engineering department

**SELECT** **AVG**(Salary) **AS** AvgEngineeringSalary

**FROM** Employees

**WHERE** Department = 'Engineering';

-- 3.4 Minimum salary in the company

**SELECT** **MIN**(Salary) **AS** MinSalary **FROM** Employees;

-- 3.5 Maximum salary in Sales department

**SELECT** **MAX**(Salary) **AS** MaxSalesSalary

**FROM** Employees

**WHERE** Department = 'Sales';

-- ============================================

-- Step 4. Combining Aggregate Functions with GROUP BY

-- ============================================

-- 4.1 Total salary per department

**SELECT** Department, **SUM**(Salary) **AS** TotalDeptSalary

**FROM** Employees

**GROUP** **BY** Department;

-- 4.2 Average salary per department

**SELECT** Department, **AVG**(Salary) **AS** AvgDeptSalary

**FROM** Employees

**GROUP** **BY** Department;

-- 4.3 Number of employees per department

**SELECT** Department, **COUNT**(\*) **AS** EmployeeCount

**FROM** Employees

**GROUP** **BY** Department;

-- ============================================

-- Step 5. Exploring Advanced Functions

-- ============================================

-- 5.1 Concatenate FirstName + LastName and calculate length of FullName

**SELECT** CONCAT(FirstName, ' ', LastName) **AS** FullName,

**LENGTH**(CONCAT(FirstName, ' ', LastName)) **AS** FullNameLength

**FROM** Employees;

-- 5.2 Count employees hired each year

**SELECT** **YEAR**(HireDate) **AS** HireYear, **COUNT**(\*) **AS** EmployeesHired

**FROM** Employees

**GROUP** **BY** **YEAR**(HireDate)

**ORDER** **BY** HireYear;

-- 5.3 Total salary expenditure per hire year

**SELECT** **YEAR**(HireDate) **AS** HireYear, **SUM**(Salary) **AS** TotalSalary

**FROM** Employees

**GROUP** **BY** **YEAR**(HireDate)

**ORDER** **BY** HireYear;