**Lab Exercise 1: Creating a Simple Function**

**Task:** Create a function named get\_total\_salary(dept\_id INT) that returns the total salary for a given department.

**Table Structure**

CREATE TABLE employees (

emp\_id INT PRIMARY KEY,

emp\_name VARCHAR(50),

dept\_id INT,

salary DECIMAL(10,2)

);

INSERT INTO employees VALUES

(1, 'Alice', 1, 5000),

(2, 'Bob', 1, 7000),

(3, 'Charlie', 2, 6000),

(4, 'David', 2, 8000);

**Solution**

DELIMITER //

CREATE FUNCTION get\_total\_salary(dept\_id INT) RETURNS DECIMAL(10,2)

DETERMINISTIC

BEGIN

DECLARE total DECIMAL(10,2);

SELECT SUM(salary) INTO total FROM employees WHERE dept\_id = dept\_id;

RETURN total;

END //

DELIMITER ;

**Lab Exercise 2: Creating a Function for Employee Bonus**

**Task:** Write a function calculate\_bonus(emp\_id INT) that returns 10% of the employee’s salary as a bonus.

**Solution**

DELIMITER //

CREATE FUNCTION calculate\_bonus(emp\_id INT) RETURNS DECIMAL(10,2)

DETERMINISTIC

BEGIN

DECLARE bonus DECIMAL(10,2);

SELECT salary \* 0.1 INTO bonus FROM employees WHERE emp\_id = emp\_id;

RETURN bonus;

END //

DELIMITER ;

**Lab Exercise 3: Creating a View for High Earners**

**Task:** Create a view high\_earners that displays employees earning more than 6000.

**Solution**

CREATE VIEW high\_earners AS

SELECT emp\_id, emp\_name, salary

FROM employees

WHERE salary > 6000;

**Lab Exercise 4: Function to Count Employees in a Department**

**Task:** Write a function count\_employees(dept\_id INT) that returns the number of employees in a department.

**Solution**

DELIMITER //

CREATE FUNCTION count\_employees(dept\_id INT) RETURNS INT

DETERMINISTIC

BEGIN

DECLARE total INT;

SELECT COUNT(\*) INTO total FROM employees WHERE dept\_id = dept\_id;

RETURN total;

END //

DELIMITER ;

**Lab Exercise 5: View for Department-wise Salary**

**Task:** Create a view department\_salary that shows department-wise total salary.

**Solution**

CREATE VIEW department\_salary AS

SELECT dept\_id, SUM(salary) AS total\_salary

FROM employees

GROUP BY dept\_id;

**Lab Exercise 6: Function to Fetch Employee Name**

**Task:** Write a function get\_employee\_name(emp\_id INT) that returns the name of an employee.

**Solution**

DELIMITER //

CREATE FUNCTION get\_employee\_name(emp\_id INT) RETURNS VARCHAR(50)

DETERMINISTIC

BEGIN

DECLARE name VARCHAR(50);

SELECT emp\_name INTO name FROM employees WHERE emp\_id = emp\_id;

RETURN name;

END //

DELIMITER ;

**Lab Exercise 7: View for Employees in a Specific Department**

**Task:** Create a view dept\_employees that filters employees by department.

**Solution**

CREATE VIEW dept\_employees AS

SELECT emp\_id, emp\_name, dept\_id

FROM employees

WHERE dept\_id = 1;

**Lab Exercise 8: Function to Calculate Average Salary**

**Task:** Write a function average\_salary(dept\_id INT) that returns the average salary of a department.

**Solution**

DELIMITER //

CREATE FUNCTION average\_salary(dept\_id INT) RETURNS DECIMAL(10,2)

DETERMINISTIC

BEGIN

DECLARE avg\_salary DECIMAL(10,2);

SELECT AVG(salary) INTO avg\_salary FROM employees WHERE dept\_id = dept\_id;

RETURN avg\_salary;

END //

DELIMITER ;

**Lab Exercise 9: View for Employee Details with Bonus**

**Task:** Create a view employee\_bonus that shows employees with their bonus amount.

**Solution**

CREATE VIEW employee\_bonus AS

SELECT emp\_id, emp\_name, salary, salary \* 0.1 AS bonus

FROM employees;

**Lab Exercise 10: Function for Maximum Salary in a Department**

**Task:** Write a function max\_salary(dept\_id INT) to return the maximum salary of a department.

**Solution**

DELIMITER //

CREATE FUNCTION max\_salary(dept\_id INT) RETURNS DECIMAL(10,2)

DETERMINISTIC

BEGIN

DECLARE max\_sal DECIMAL(10,2);

SELECT MAX(salary) INTO max\_sal FROM employees WHERE dept\_id = dept\_id;

RETURN max\_sal;

END //

DELIMITER ;

1. **Function to find minimum salary in a department**
2. **View for employees earning below average salary**
3. **Function to return the department name given an employee ID**
4. **View for employees sorted by salary**
5. **Function to get the count of employees earning above a given salary**
6. **View for employees working in a specific department using JOINs**
7. **Function to calculate total salary expenditure**
8. **View for employees hired within the last 6 months**
9. **Function to return the highest-paid employee’s name in a department**
10. **View for listing employees along with their department details**