PL/SQL program to demonstrate a **procedure for updating or checking an employee's salary**

**1. Aim**

To create a PL/SQL procedure that takes an employee ID as input and displays their salary.

The procedure can be extended to update the salary or perform validations.

**2. Design**

* The procedure accepts an employee ID.
* It checks if the employee exists in the table.
* If found, it retrieves and displays the salary.
* The procedure handles exceptions for invalid employee IDs.

**Assumption**: The table used is EMPLOYEES(emp\_id, emp\_name, salary).

**3. Algorithm/Steps**

1. Define a procedure get\_salary with input parameter p\_emp\_id.
2. Declare a variable to store the salary.
3. Use SELECT INTO to fetch the salary from the EMPLOYEES table.
4. Print the salary using SELECT CONCAT AS message;
5. Add exception handling to catch:
   * No data found (employee doesn't exist).
   * Others (any other errors).

### 1. ****Table Creation in MySQL****

Mysql>CREATE TABLE EMPLOYEE (

emp\_id INT PRIMARY KEY,

emp\_name VARCHAR(20),

salary DECIMAL(10, 2)

);

### 2.Insert ****Data into EMPLOYEE Table****

Mysql>INSERT INTO EMPLOYEE (emp\_id, emp\_name, salary) VALUES (101, 'Alice Johnson', 55000.00);

INSERT INTO EMPLOYEE (emp\_id, emp\_name, salary) VALUES (102, 'Bob Smith', 62000.00);

INSERT INTO EMPLOYEE (emp\_id, emp\_name, salary) VALUES (103, 'Charlie Davis', 48000.00);

INSERT INTO EMPLOYEE (emp\_id, emp\_name, salary) VALUES (104, 'Diana Moore', 75000.00);

INSERT INTO EMPLOYEE (emp\_id, emp\_name, salary) VALUES (105, 'Ethan Brown', 51000.00);

OR

Mysql>INSERT INTO EMPLOYEE VALUES (101, 'Alice Johnson', 55000.00);

INSERT INTO EMPLOYEE VALUES (102, 'Bob Smith', 62000.00);

INSERT INTO EMPLOYEE VALUES (103, 'Charlie Davis', 48000.00);

INSERT INTO EMPLOYEE VALUES (104, 'Diana Moore', 75000.00);

INSERT INTO EMPLOYEE VALUES (105, 'Ethan Brown', 51000.00);

### 3. ****Stored Procedure in MySQL (Equivalent to PL/SQL)****

Mysql>DELIMITER $$

Mysql>CREATE PROCEDURE get\_salary(IN p\_emp\_id INT)

BEGIN

DECLARE v\_salary DECIMAL(10,2);

DECLARE CONTINUE HANDLER FOR NOT FOUND

BEGIN

SELECT CONCAT('Employee with ID ', p\_emp\_id, ' not found.') AS message;

END;

-- Try to fetch salary

SELECT salary INTO v\_salary FROM EMPLOYEE WHERE emp\_id = p\_emp\_id;

-- If found, show salary

SELECT CONCAT('Salary of Employee ID ', p\_emp\_id, ' is: ', v\_salary) AS message;

END$$

Mysql>DELIMITER $$;

### 4. ****Calling the Procedure****

Mysql>CALL get\_salary(101); -- Expected Output: Salary of Employee ID 101 is: 55000.00

Mysql>CALL get\_salary(999); -- Expected Output: Employee with ID 999 not found.

### 5. ****Output****

MySQL uses SELECT statements to simulate message output.

### 6. ****Test Cases Summary Table****

| **Test Case** | **Input (emp\_id)** | **Expected Output** | **Status** |
| --- | --- | --- | --- |
| TC1 | 101 | Salary of Employee ID 101 is: 55000.00 | Pass |
| TC2 | 999 | Employee with ID 999 not found. | Pass |
| TC3 | NULL/invalid | Error (procedure will raise an exception) | Pass |

Program Tested on Mysql Software

mysql> CREATE TABLE EMPLOYEE (

-> emp\_id INT PRIMARY KEY,

-> emp\_name VARCHAR(20),

-> salary DECIMAL(10, 2)

-> );

Query OK, 0 rows affected (0.09 sec)

mysql> INSERT INTO EMPLOYEE VALUES (101, 'Alice Johnson', 55000.00);

Query OK, 1 row affected (0.03 sec)

mysql> INSERT INTO EMPLOYEE VALUES (102, 'Bob Smith', 62000.00);

Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO EMPLOYEE VALUES (103, 'Charlie Davis', 48000.00);

Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO EMPLOYEE VALUES (104, 'Diana Moore', 75000.00);

Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO EMPLOYEE VALUES (105, 'Ethan Brown', 51000.00);

Query OK, 1 row affected (0.01 sec)

mysql>

mysql> commit;

Query OK, 0 rows affected (0.00 sec)

mysql> select \* from employee;

+--------+---------------+----------+

| emp\_id | emp\_name | salary |

+--------+---------------+----------+

| 101 | Alice Johnson | 55000.00 |

| 102 | Bob Smith | 62000.00 |

| 103 | Charlie Davis | 48000.00 |

| 104 | Diana Moore | 75000.00 |

| 105 | Ethan Brown | 51000.00 |

+--------+---------------+----------+

5 rows in set (0.01 sec)

mysql> DELIMITER $$

mysql> CREATE PROCEDURE get\_salary(IN p\_emp\_id INT)

BEGIN

DECLARE v\_salary DECIMAL(10,2);

DECLARE CONTINUE HANDLER FOR NOT FOUND

BEGIN

SELECT CONCAT('Employee with ID ', p\_emp\_id, ' not found.') AS message;

END;

-- Try to fetch salary

SELECT salary INTO v\_salary FROM EMPLOYEE WHERE emp\_id = p\_emp\_id;

-- If found, show salary

SELECT CONCAT('Salary of Employee ID ', p\_emp\_id, ' is: ', v\_salary) AS message;

END$$

Query OK, 0 rows affected (0.05 sec)

mysql> CALL get\_salary(101);

-> $$

+----------------------------------------+

| message |

+----------------------------------------+

| Salary of Employee ID 101 is: 55000.00 |

+----------------------------------------+

1 row in set (0.02 sec)

Query OK, 0 rows affected (0.02 sec)

mysql>

mysql> CALL get\_salary(999);

-> $$

+---------------------------------+

| message |

+---------------------------------+

| Employee with ID 999 not found. |

+---------------------------------+

1 row in set (0.00 sec)

+---------+

| message |

+---------+

| NULL |

+---------+

1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)

mysql>

Modified Program

mysql> DELIMITER $$

mysql> CREATE PROCEDURE get\_salary(IN p\_emp\_id INT)

BEGIN

DECLARE v\_salary DECIMAL(10,2);

DECLARE CONTINUE HANDLER FOR NOT FOUND

BEGIN

SELECT CONCAT('Employee with ID ', p\_emp\_id, ' not found.') AS message;

END;

Update employee set salary=salary \*1.25 where emp\_id=p\_emp\_id;

-- Try to fetch salary

SELECT salary INTO v\_salary FROM EMPLOYEE WHERE emp\_id = p\_emp\_id;

-- If found, show salary

SELECT CONCAT('Updated Salary of Employee ID ', p\_emp\_id, ' is: ', v\_salary) AS message;

END$$