**Lab\_5\_Suraj\_BHAttarai\_38**

1. Write a stored procedure named “GetEmployee()” to get name, birthdate, address of employees.

Ans:

Query:

DELIMITER //

CREATE PROCEDURE GetEmployee()

BEGIN

SELECT Ename, Bdate, Address FROM Employee;

END //

DELIMITER ;

Result:

Query OK, 0 rows affected (0.01 sec)

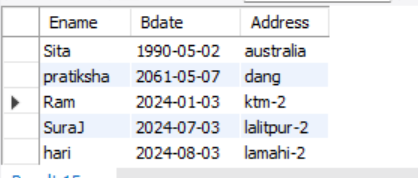
1. Execute the procedure in Q.1 and show the result.

Ans:

Query:

CALL GetEmployee();

Result:



1. Write a stored procedure to get PF category name, Amount and start date where the amount is greater than provided input value. Your procedure should contain an IN parameter named amt to take input value of amount. Call the procedure with inputs 1000 and 3000 respectively.

Ans:

Query:

DELIMITER //

CREATE PROCEDURE GetPFCategories(IN amt INT)

BEGIN

SELECT PFCategoryName, Amount, Start\_date FROM PF WHERE Amount > amt;

END;

DELIMITER ;

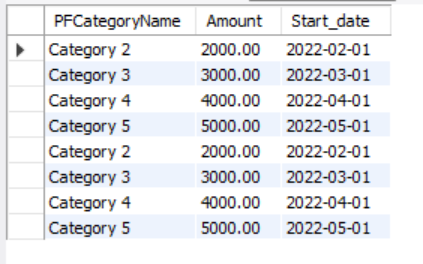
**Result:**

Query OK, 0 rows affected (0.01 sec)

**Calling the procedure with inputs 1000 and 3000 respectively:**

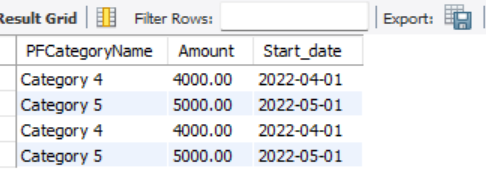
**Query:** CALL GetPFCategories(1000);

**Result**



**Query:** CALL GetPFCategories(3000);

**Result:**

****

1. Write a stored procedure to get number of PF records where the amount of PF is equal to the provided input value. Your procedure should contain an IN parameter named amt to take input value of amount and should contain OUT parameter named total to return the total number of PF records satisfying the condition.

Ans:

**Query**:

DELIMITER //

CREATE PROCEDURE GetTotalPFForAmount(IN amt INT, OUT total INT)

BEGIN

SELECT COUNT(\*) INTO total FROM PF WHERE Amount = amt;

END//

DELIMITER ;  
**Result:**

Query OK, 0 rows affected (0.01 sec

1. Call the procedure in Q4. with input of 3000 and print the @total.

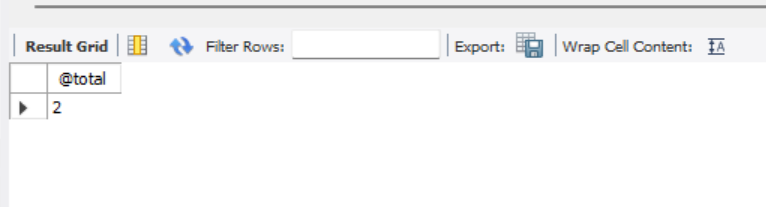
Ans:

**Query:**

CALL GetTotalPFForAmount(3000, @total);

SELECT @total;

**Result:**

****

1. Write before insert trigger before inserting a record into the Employee table. Show some action on the event.

Ans:

**Query:**

DELIMITER //

CREATE TRIGGER before\_employee\_insert

BEFORE INSERT ON Employee

FOR EACH ROW

BEGIN

-- Perform some action on the event

DECLARE employeeCount INT;

-- Check if the number of employees exceeds a certain limit

SELECT COUNT(\*) INTO employeeCount FROM Employee;

IF employeeCount >4 THEN

-- Raise an error if the limit is exceeded

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Maximum employee limit reached.';

END IF;

END //

DELIMITER ;

Result:

0 row(s) affected

**When I try to insert then following error came that mean trigger is activated**

**Query:**

INSERT INTO employee(SSN, Ename, Gender, Bdate, Address, Salary, Ono,Years\_of\_experience, Marital\_status) VALUES( 7, 'padam', 'M', '1994-04-25','ktm',50000,5,6,'single')

**Result:**

ERROR 1644 (45000): Maximum employee limit reached.

mysql>

1. Write after delete trigger on PF table during delete operation. Print “It is deleted”.

Ans:

Query:

DELIMITER //

CREATE TRIGGER after\_pf\_delete

AFTER DELETE ON PF

FOR EACH ROW

BEGIN

-- Perform some action after the delete operation

INSERT INTO LogTable (Message) VALUES ('It is deleted');

END //

DELIMITER //

**Result:**

0 row(s) affected

**After that to check I tried following query**

**Query:**

DELETE FROM pf where PFID = 9

**Result:**

ERROR 1146 (42S02): Table 'suraj\_38\_company.LogTable' doesn't exist