**EX.NO:5**

**Question 5**

**Create cursor for Employee table & extract the values from the table. Declare the variables,Open the cursor & extract the values from the cursor. Close the cursor.**

**CUSTOMERS(ID,NAME,AGE,ADDRESS,SALARY)**

Solution

1. Creating the Employee Table and insert few records

CREATE DATABASE COMPANY05;

USE COMPANY05;

CREATE TABLE Employee (

E\_id INT,

E\_name VARCHAR(255),

Age INT,

Salary DECIMAL(10, 2)

);

INSERT INTO Employee (E\_id, E\_name, Age, Salary)

VALUES

(1, 'Samarth', 30, 50000.00),

(2, 'Ramesh Kumar', 25, 45000.00),

(3, 'Seema Banu', 35, 62000.00),

(4, 'Dennis Anil', 28, 52000.00),

(5, 'Rehman Khan', 32, 58000.00);

2. Create a Stored Procedure with Cursor

To create a cursor for the Employee table, extract values using the cursor, and then close the cursor in MySQL, you’ll need to use stored procedures that support cursor operations.

DELIMITER //

CREATE PROCEDURE fetch\_employee\_data()

BEGIN

-- Declare variables to store cursor values

DECLARE emp\_id INT;

DECLARE emp\_name VARCHAR(255);

DECLARE emp\_age INT;

DECLARE emp\_salary DECIMAL(10, 2);

-- Declare a cursor for the Employee table

DECLARE emp\_cursor CURSOR FOR

SELECT E\_id, E\_name, Age, Salary

FROM Employee;

-- Declare a continue handler for the cursor

DECLARE CONTINUE HANDLER FOR NOT FOUND

SET @finished = 1;

-- Open the cursor

OPEN emp\_cursor;

-- Initialize a variable to control cursor loop

SET @finished = 0;

-- Loop through the cursor results

cursor\_loop: LOOP

-- Fetch the next row from the cursor into variables

FETCH emp\_cursor INTO emp\_id, emp\_name, emp\_age, emp\_salary;

-- Check if no more rows to fetch

IF @finished = 1 THEN

LEAVE cursor\_loop;

END IF;

-- Output or process each row (for demonstration, print the values)

SELECT CONCAT('Employee ID: ', emp\_id, ', Name: ', emp\_name, ', Age: ', emp\_age, ', Salary: ', emp\_salary) AS Employee\_Info;

END LOOP;

-- Close the cursor

CLOSE emp\_cursor;

END//

DELIMITER ;

In this stored procedure (fetch\_employee\_data):

We declare variables (emp\_id, emp\_name, emp\_age, emp\_salary) to store values retrieved from the cursor.

A cursor (emp\_cursor) is declared to select E\_id, E\_name, Age, and Salary from the Employee table.

We declare a continue handler (CONTINUE HANDLER) for NOT FOUND condition to handle the end of cursor data.

The cursor is opened (OPEN emp\_cursor), and a loop (cursor\_loop) is used to fetch each row from the cursor.

We fetch values into the variables and process them within the loop (for demonstration, we print the values using a SELECT statement).

The loop continues until all rows are fetched (@finished = 1).

Finally, the cursor is closed (CLOSE emp\_cursor).

3. Execute the Stored Procedure

Once the stored procedure fetch\_employee\_data is created, you can execute it to fetch and process data from the Employee table:

mysql> CALL fetch\_employee\_data();

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

| Employee\_Info |

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

| Employee ID: 1, Name: Samarth, Age: 30, Salary: 50000.00 |

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

1 row in set (0.07 sec)

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

| Employee\_Info |

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

| Employee ID: 2, Name: Ramesh Kumar, Age: 25, Salary: 45000.00 |

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

1 row in set (0.07 sec)

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

| Employee\_Info |

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

| Employee ID: 3, Name: Seema Banu, Age: 35, Salary: 62000.00 |

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

1 row in set (0.07 sec)

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

| Employee\_Info |

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

| Employee ID: 4, Name: Dennis Anil, Age: 28, Salary: 52000.00 |

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

1 row in set (0.07 sec)

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

| Employee\_Info |

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

| Employee ID: 5, Name: Rehman Khan, Age: 32, Salary: 58000.00 |

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

1 row in set (0.07 sec)

Query OK, 0 rows affected (0.07 sec)

The stored procedure fetch\_employee\_data declares variables (emp\_id, emp\_name, emp\_age, emp\_salary) to store values retrieved from the cursor.

A cursor (emp\_cursor) is declared for the Employee table to select E\_id, E\_name, Age, and Salary.

The cursor is opened (OPEN emp\_cursor), and the FETCH statement retrieves the first row from the cursor into the declared variables.

A WHILE loop processes each row fetched by the cursor (SQLSTATE() = '00000' checks for successful fetching).

Within the loop, you can perform operations or output the values of each row.

The CLOSE statement closes the cursor after processing all rows.

This example demonstrates how to create and use a cursor in MySQL to extract values from the Employee table row by row. Adjust the cursor query and processing logic based on your table structure and desired operations.