

With respect to employee table and department table in the last assignment, Create Stored Procedure with the name GetLimitedEmployee with the input parameter.

Procedure display the list of employee with their first_name, last_name, salary, hire_date. Input parameter must be a integer which is used to display limited number of records.

```
create database b;
```

```
use b;
```

```
-- Create the department table
```

```
CREATE TABLE department (  
    dept_id INT PRIMARY KEY,  
    dept_name VARCHAR(50),  
    manager_id INT,  
    location_id INT  
);
```

```
-- Create the employee table
```

```
CREATE TABLE employee (  
    emp_id INT PRIMARY KEY,  
    first_name VARCHAR(50),  
    last_name VARCHAR(50),  
    email VARCHAR(50),  
    mobile VARCHAR(50),  
    hire_date DATE,  
    job_id INT,  
    salary FLOAT,  
    manager_id INT,  
    dept_id INT,  
    FOREIGN KEY (dept_id) REFERENCES department(dept_id)  
);
```

```
-- Inserting into department table
```

```
INSERT INTO department (dept_id, dept_name, manager_id, location_id) VALUES
```

```
(1, 'HR', 101, 1001),
(2, 'IT', 102, 1002),
(3, 'Finance', 103, 1003),
(4, 'Marketing', 104, 1004),
(5, 'Sales', 105, 1005),
(6, 'Operations', 106, 1006),
(7, 'Research', 107, 1007),
(8, 'Development', 108, 1008),
(9, 'Support', 109, 1009),
(10, 'Admin', 110, 1010);
```

-- Inserting into employee table

```
INSERT INTO employee (emp_id, first_name, last_name, email, mobile, hire_date, job_id,
salary, manager_id, dept_id) VALUES
```

```
(1, 'John', 'Doe', 'john.doe@example.com', '1234567890', '2020-01-01', 201, 60000, 101, 1),
(2, 'Jane', 'Smith', 'jane.smith@example.com', '0987654321', '2019-02-01', 202, 65000, 102,
2),
(3, 'Jim', 'Beam', 'jim.beam@example.com', '1230984567', '2018-03-01', 203, 55000, 103, 3),
(4, 'Jill', 'Stark', 'jill.stark@example.com', '3216549870', '2021-04-01', 204, 50000, 104, 4),
(5, 'Jack', 'Black', 'jack.black@example.com', '2135468790', '2020-05-01', 205, 70000, 105,
5),
(6, 'Julia', 'Roberts', 'julia.roberts@example.com', '2315648790', '2017-06-01', 206, 80000,
106, 6),
(7, 'Jeremy', 'Clark', 'jeremy.clark@example.com', '4567891230', '2016-07-01', 207, 90000,
107, 7),
(8, 'Jessica', 'Alba', 'jessica.alba@example.com', '3214567890', '2015-08-01', 208, 75000,
108, 8),
(9, 'Jason', 'Momoa', 'jason.momoa@example.com', '7894561230', '2014-09-01', 209,
65000, 109, 9),
(10, 'Jake', 'Gyllenhaal', 'jake.gyllenhaal@example.com', '1236547890', '2022-10-01', 210,
60000, 110, 10);
```

```
DELIMITER $$
```

```
CREATE PROCEDURE GetLimitedEmployee(IN limit_count INT)
```

BEGIN

SELECT

first_name,

last_name,

salary,

hire_date

FROM

employee

LIMIT

limit_count;

END\$\$

CALL GetLimitedEmployee(5);

The screenshot displays the MySQL Workbench interface. The 'Query Editor' window shows the following SQL code:

```
1  use $;
2  SETIMITER $$
3  CREATE PROCEDURE GetLimitedEmployee(IN limit_count INT)
4  BEGIN
5      SELECT
6          first_name,
7          last_name,
8          salary,
9          hire_date
10     FROM
11         employee
12     LIMIT
13         limit_count;
```

The 'Results' window shows the output of the procedure call, displaying a table with 5 rows:

| first_name | last_name | salary | hire_date |
|------------|-----------|--------|------------|
| John | Deere | 8000 | 2000-01-01 |
| Jane | Smith | 9000 | 2000-02-01 |
| Bob | Stevens | 10000 | 2000-03-01 |
| Jack | Smith | 8000 | 2001-04-01 |
| Jack | Smith | 7000 | 2002-05-01 |

The 'Log' window shows the execution details:

| Step | Time | Action | Message | Rows affected | Duration / Path |
|------|---------------------|------------------|--|-----------------|-------------------|
| 1 | 2023-01-01 10:00:00 | USE | USE | 0 rows affected | 0:00:00 |
| 2 | 2023-01-01 10:00:00 | CREATE PROCEDURE | CREATE PROCEDURE GetLimitedEmployee(IN INT) BEGIN SELECT first_name, last_name, salary, hire_date FROM employee LIMIT limit_count; | 0 rows affected | 0:00:00 |
| 3 | 2023-01-01 10:00:00 | CALL | CALL GetLimitedEmployee(5); | 5 rows returned | 0:00:00 / 0:00:00 |