Assignment 9

Stored Procedures

Consider the Worker table with following fields: Worker_Id INT FirstName CHAR(25), LastName CHAR(25), Salary INT(15), JoiningDate DATETIME, Department CHAR(25).

Query and Result

```
create database assignment_9;
use assignment_9;
/*Consider the Worker table with following fields: Worker_Id INT FirstName CHAR(25),
LastName CHAR(25), Salary INT(15), JoiningDate DATETIME, Department CHAR(25))*/
create table worker(worker_id int,
                    firstname char(25),
                    lastname char(25),
                    salary int,
                    joiningdate datetime,
                    department char(25));
desc worker;
insert into worker(worker_id, firstname, lastname, salary, joiningdate, department)
values(1, 'Nithya', 'Jayaram', 20000, '2024-11-03 09:00:00', 'IT'),
(2, 'Femi', 'Azad', 21000, '02024-11-04 09:00:00', 'Accounts'),
(3, 'Divya', 'John', 19000, '2024-11-05 09:00:00', 'IT'),
(4, 'Ajay', 'Raj', 22000, '2024-11-06 09:00:00', 'Administration'),
(5, 'Sindhu', 'Suresh', 18000, '2024-11-07 09:00:00', 'IT');
select* from worker;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

worker_id	firstname	lastname	salary	joiningdate	department
1	Nithya	Jayaram	20000	2024-11-03 09:00:00	IT
2	Femi	Azad	21000	2024-11-04 09:00:00	Accounts
3	Divya	John	19000	2024-11-05 09:00:00	IT
4	Ajay	Raj	22000	2024-11-06 09:00:00	Administration
5	Sindhu	Suresh	18000	2024-11-07 09:00:00	IT

1. Create a stored procedure that takes in IN parameters for all the columns in the Worker table and adds a new record to the table and then invokes the procedure call.

Query and Result

```
21    Delimiter $$
22 •  create procedure worker(in p_worker_id int,
23                            in p_firstname char(25),
24                            in p_lastname char(25),
25                            in p_salary int,
26                            in p_joiningdate datetime,
27                            in p_department char(25))
28    begin
29    insert into worker(worker_id, firstname, lastname, salary, joiningdate, department)
30    values(p_worker_id, p_firstname, p_lastname, p_salary, p_joiningdate, p_department);
31    end $$
32    Delimiter ;
33 •  call worker(6, 'Sanya', 'Ram', 20000, '2024-10-12 09:00:00', 'Accounts');
34 •  select * from worker;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

worker_id	firstname	lastname	salary	joiningdate	department
1	Nithya	Jayaram	20000	2024-11-03 09:00:00	IT
2	Femi	Azad	21000	2024-11-04 09:00:00	Accounts
3	Divya	John	19000	2024-11-05 09:00:00	IT
4	Ajay	Raj	22000	2024-11-06 09:00:00	Administration
5	Sindhu	Suresh	18000	2024-11-07 09:00:00	IT
6	Sanya	Ram	20000	2024-10-12 09:00:00	Accounts

2. Write stored procedure takes in an IN parameter for WORKER_ID and an OUT parameter for SALARY. It should retrieve the salary of the worker with the given ID and returns it in the p_salary parameter. Then make the procedure call.

Query and Result

```
38    Delimiter $$
39 •  create procedure GetWorkerSalary(in n_worker_id int, out n_salary int)
40    begin
41        select salary into n_salary from worker where worker_id = n_worker_id;
42    end$$
43    Delimiter ;
44 •  set @salary_result = 0;
45 •  call GetWorkerSalary(1, @salary_result);
46 •  select @salary_result AS WorkerSalary;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

WorkerSalary
20000

3. Create a stored procedure that takes in IN parameters for WORKER_ID and DEPARTMENT. It should update the department of the worker with the given ID. Then make a procedure call.

Query and Result

```
49    Delimiter $$
50    create procedure UpdateWorkerDepartment(in p_worker_id int, in p_new_department char(25))
51    begin
52        update worker set department = p_new_department where worker_id = p_worker_id;
53    end $$
54    Delimiter ;
55    set sql_safe_updates =0;
56    call UpdateWorkerDepartment(3, 'Marketing');
57    select worker_id, department from worker where worker_id = 3;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

worker_id	department
3	Marketing

4. Write a stored procedure that takes in an IN parameter for DEPARTMENT and an OUT parameter for p_workerCount. It should retrieve the number of workers in the given department and returns it in the p_workerCount parameter. Make procedure call.

Query and Result

```
61    Delimiter $$
62    create procedure GetWorkerCountByDepartment(in p_department char(25), out p_workerCount int)
63    begin
64        select count(*) into p_workerCount from worker where department = p_department;
65    end$$
66    Delimiter ;
67    set @worker_count = 0;
68    call GetWorkerCountByDepartment('IT', @worker_count);
69    select @worker_count as WorkerCount;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

WorkerCount
2

5. Write a stored procedure that takes in an IN parameter for DEPARTMENT and an OUT parameter for p_avgSalary. It should retrieve the average salary of all workers in the given department and returns it in the p_avgSalary parameter and call the procedure.

Query and Result

```
74    Delimiter $$
75    create procedure GetAvgSalaryByDepartment(in p_department char(25), out p_avgSalary DECIMAL(10,2))
76    begin
77        select avg(salary) into p_avgSalary from worker where department = p_department;
78    end$$
79    Delimiter ;
80    set @avg_salary = 0;
81    call GetAvgSalaryByDepartment('Accounts', @avg_salary);
82    select @avg_salary as AvgSalary;
```