DBs LabNo.10

Name: Umair Hafeez | CMS ID: 023-24-0473 | BSCS II

DATABASE SYSTEM

LAB No: 10

Instructor: Marina Gul

Objective of Lab No. 10:

After performing lab 10, students will be able to:

- To learn Stored Procedures
- To learn Stored Functions

1. Stored procedure

2. Stored Function

1. **Create a stored procedure DISPLAY without parameters. The procedure must display empno, ename and salary of all the employees of DEPTNO = 10.**

Query: delimiter $$

```
create procedure Display()
begin select employee_id, first_name, salary from employees where department_id=10;
end $$
delimiter ;
call Display();
```

2. **Create a stored procedure DISPLAY2 with parameters. It must take DEPTNO as an input and must return the DNAME and TOTAL SALARY of the input department number.**

Query: delimiter $$

```
create procedure Display2(in p_depart int)
begin select d.department_name, sum(e.salary) as Total_Salary from employees e
join departments d on e.department_id=d.department_id
where e.department_id=p_depart group by department_name;
end $$
delimiter ;
call display2(10);
```

3. **Create a stored procedure DISPLAY3 with parameters. It must take DEPTNO as an input and must return the DNAME, SMALLEST and HIGHEST SALARIES of the input department**

number. DISPLAY3 must also display empno,ename,total salary (sal+comm) of all the employees of the input department number.

Query: delimiter $$

create procedure Display3(in p1_depart int)

begin select d.department_name, min(e.salary)as lowest_salary, max(e.salary) as highest_Salary from employees e

join departments d on e.department_id=d.department_id

where e.department_id=p1_depart group by department_name;

select employee_id, first_name, (salary+(salary * IFNULL(commission_pct, 0))) as Total_salary from employees where department_id=p1_depart;

end $$

delimiter ;

call Display3(10);

4. **Create a stored function MANAGER without input parameters. It must return the total salary of all the managers in the EMP.**

Query:

delimiter $$

create function MANAGER()

returns decimal(10,2)

deterministic

begin

declare total_salary decimal(10,2);

select sum(salary) into total_salary

from employees

where employee_id in (select distinct manager_id from employees where manager_id is not null);

return total_salary;

end $$

delimiter ;

5. **Create a stored function MANAGER2 with parameters. It must take empno as an input and must return its manager name. Write a SELECT statement to display all employees' names and their manager names. Manager names must be displayed using MANAGER2 stored function.**

Query:

```
delimiter $$
create function MANAGER2(p_empno int)
returns varchar(50)
deterministic
begin
 declare mgr_name varchar(50);
 select concat(first_name, ' ', last_name) into mgr_name
 from employees
 where employee_id = (select manager_id from employees where employee_id = p_empno);
 return mgr_name;
end $$
delimiter ;
```

6. **Create a stored function MANAGER3 with parameters. It must take MANAGER NAME as an input and must return average salary of its employees.**

Query:

```
delimiter $$
create function MANAGER3(p_mgr_name varchar(50))
returns decimal(10,2)
deterministic
begin
 declare avg_sal decimal(10,2);
 declare mgr_id int;
 select employee_id into mgr_id
 from employees
 where concat(first_name, ' ', last_name) = p_mgr_name
 limit 1;
 select avg(salary) into avg_sal
 from employees
 where manager_id = mgr_id;
```

```
  return avg_sal;
end $$
delimiter ;
```