1. Write a procedure to display the name of employee getting highest salary.

```
--
CREATE OR REPLACE PROCEDURE display_highest_salary_employee
IS
  v_highest_salary NUMBER;
  v_employee_name VARCHAR2(50);
BEGIN
  -- Find the highest salary
  SELECT MAX(sal) INTO v_highest_salary FROM emp;

  -- Find the employee name with the highest salary
  SELECT ename INTO v_employee_name
  FROM emp
  WHERE sal = v_highest_salary;

  -- Display the result
  DBMS_OUTPUT.PUT_LINE('Employee with the highest salary: ' || v_employee_name || '
(Salary: ' || v_highest_salary || ')');
END display_highest_salary_employee;
/
```

2. Create a PL/SQL function named Get_Name that accepts an employee number as input and returns
the employee's name. Write a PL/SQL block to call this function and display the output.

```
CREATE OR REPLACE FUNCTION Get_Name(p_empno IN NUMBER)
  RETURN VARCHAR2
IS
  v_ename VARCHAR2(50);
BEGIN
  -- Retrieve the employee name based on the employee number
  SELECT ename INTO v_ename
  FROM emp
  WHERE empno = p_empno;

  -- Return the employee name
  RETURN v_ename;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    RETURN NULL; -- Handle the case where the employee number is not found
END Get_Name;
/

-- PL/SQL block to call the function and display the output
DECLARE
```

```
  v_employee_number NUMBER := 7839; -- You can change this to the desired employee
number
  v_employee_name VARCHAR2(50);
BEGIN
  -- Call the function to get the employee name
  v_employee_name := Get_Name(v_employee_number);

  -- Display the result
  IF v_employee_name IS NOT NULL THEN
    DBMS_OUTPUT.PUT_LINE('Employee Name for Employee ' || v_employee_number || ': '
|| v_employee_name);
  ELSE
    DBMS_OUTPUT.PUT_LINE('Employee not found for Employee Number ' ||
v_employee_number);
  END IF;
END;
/
```

3. Write a PL/SQL block to display the name of employee with salary and Department name, of any
employee.(Assume: empno = 7839). Write Exception if no employee exists with the input empno.

```
DECLARE
  v_empno NUMBER := 7839; -- Specify the desired employee number
  v_ename VARCHAR2(50);
  v_salary NUMBER;
  v_dname VARCHAR2(50);
BEGIN
  -- Retrieve employee details based on the employee number
  SELECT e.ename, e.sal, d.dname
  INTO v_ename, v_salary, v_dname
  FROM emp e
  JOIN dept d ON e.deptno = d.deptno
  WHERE e.empno = v_empno;

  -- Display the result
  DBMS_OUTPUT.PUT_LINE('Employee Name: ' || v_ename);
  DBMS_OUTPUT.PUT_LINE('Salary: ' || v_salary);
  DBMS_OUTPUT.PUT_LINE('Department Name: ' || v_dname);
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('No employee found for Employee Number ' || v_empno);
END;
/
```

4. Write a PL/SQL procedure to display the name of the employee who receives the highest salary in
the EMP table.

```
CREATE OR REPLACE PROCEDURE Display_Highest_Salary_Employee
IS
  v_employee_name VARCHAR2(50);
  v_highest_salary NUMBER;
BEGIN
  -- Find the highest salary in the EMP table
  SELECT MAX(sal) INTO v_highest_salary FROM emp;

  -- Find the employee name(s) with the highest salary
  SELECT ename INTO v_employee_name
  FROM emp
  WHERE sal = v_highest_salary;

  -- Display the result
  IF v_employee_name IS NOT NULL THEN
    DBMS_OUTPUT.PUT_LINE('Employee(s) with the highest salary: ' || v_employee_name ||
' (Salary: ' || v_highest_salary || ')');
  ELSE
    DBMS_OUTPUT.PUT_LINE('No employee found in the EMP table.');
  END IF;
END Display_Highest_Salary_Employee;
/
```

5. Develop a PL/SQL block to calculate and display the average salary for employees in a specific
department (e.g., DEPTNO = 10).

```
DECLARE
  v_department_number NUMBER := 10; -- Specify the desired department number
  v_avg_salary NUMBER;
BEGIN
  -- Calculate the average salary for employees in the specified department
  SELECT AVG(sal) INTO v_avg_salary
  FROM emp
  WHERE deptno = v_department_number;

  -- Display the result
  IF v_avg_salary IS NOT NULL THEN
```

```
    DBMS_OUTPUT.PUT_LINE('Average Salary for Department ' || v_department_number ||
': ' || TO_CHAR(v_avg_salary, '99999.99'));
  ELSE
    DBMS_OUTPUT.PUT_LINE('No employees found for Department ' ||
v_department_number);
  END IF;
END;
/
```

6. Write a PL/SQL block to update the salary of an employee (e.g., empno = 7369) by a specified
percentage (e.g., 10%). Display the updated salary and the original salary.

```
DECLARE
  v_employee_number NUMBER := 7369; -- Specify the desired employee number
  v_percentage_increase NUMBER := 10; -- Specify the desired percentage increase
  v_original_salary NUMBER;
  v_updated_salary NUMBER;
BEGIN
  -- Retrieve the original salary
  SELECT sal INTO v_original_salary
  FROM emp
  WHERE empno = v_employee_number;

  -- Update the salary by the specified percentage
  v_updated_salary := v_original_salary + (v_original_salary * v_percentage_increase / 100);

  -- Update the salary in the EMP table
  UPDATE emp
  SET sal = v_updated_salary
  WHERE empno = v_employee_number;

  -- Display the result
  DBMS_OUTPUT.PUT_LINE('Employee Number: ' || v_employee_number);
  DBMS_OUTPUT.PUT_LINE('Original Salary: ' || TO_CHAR(v_original_salary, '99999.99'));
  DBMS_OUTPUT.PUT_LINE('Updated Salary: ' || TO_CHAR(v_updated_salary, '99999.99'));
END;
/
```

7. Create a PL/SQL procedure that takes a department number as input and displays the details
(Empno, Ename, Salary) of all employees in that department.

```
CREATE OR REPLACE PROCEDURE
Display_Employees_In_Department(p_department_number IN NUMBER)
IS
BEGIN
  -- Display the details of employees in the specified department
  FOR emp_record IN (SELECT empno, ename, sal
                     FROM emp
                     WHERE deptno = p_department_number)
  LOOP
    DBMS_OUTPUT.PUT_LINE('Employee Number: ' || emp_record.empno);
    DBMS_OUTPUT.PUT_LINE('Employee Name: ' || emp_record.ename);
    DBMS_OUTPUT.PUT_LINE('Salary: ' || TO_CHAR(emp_record.sal, '99999.99'));
    DBMS_OUTPUT.PUT_LINE('----------------------');
  END LOOP;

  -- Check if any employees were found in the specified department
  IF SQL%NOTFOUND THEN
    DBMS_OUTPUT.PUT_LINE('No employees found for Department ' ||
p_department_number);
  END IF;
END Display_Employees_In_Department;
/
```

8. Write a PL/SQL block to calculate and display the count of employees in a specific department.
Allow the department number to be a parameter in your block.

```
DECLARE
  v_department_number NUMBER := 20; -- Specify the desired department number
  v_employee_count NUMBER;
BEGIN
  -- Calculate the count of employees in the specified department
  SELECT COUNT(*) INTO v_employee_count
  FROM emp
  WHERE deptno = v_department_number;

  -- Display the result
  DBMS_OUTPUT.PUT_LINE('Employee Count for Department ' || v_department_number ||
': ' || v_employee_count);
END;
/
```

9. Create a PL/SQL function named Get_Avg_Salary () that calculates and returns the average

salary of all employees in a given department. Write a PL/SQL block to call this function with a
specific DEPTNO and display the result.

```
CREATE OR REPLACE FUNCTION Get_Avg_Salary(p_department_number IN NUMBER)
  RETURN NUMBER
IS
  v_avg_salary NUMBER;
BEGIN
  -- Calculate the average salary for employees in the specified department
  SELECT AVG(sal) INTO v_avg_salary
  FROM emp
  WHERE deptno = p_department_number;

  -- Return the result
  RETURN v_avg_salary;
END Get_Avg_Salary;
/

-- PL/SQL block to call the function and display the result
DECLARE
  v_department_number NUMBER := 30; -- Specify the desired department number
  v_result_avg_salary NUMBER;
BEGIN
  -- Call the function to get the average salary
  v_result_avg_salary := Get_Avg_Salary(v_department_number);

  -- Display the result
  IF v_result_avg_salary IS NOT NULL THEN
    DBMS_OUTPUT.PUT_LINE('Average Salary for Department ' || v_department_number ||
': ' || TO_CHAR(v_result_avg_salary, '99999.99'));
  ELSE
    DBMS_OUTPUT.PUT_LINE('No employees found for Department ' ||
v_department_number);
  END IF;
END;
/
```

10. Develop a PL/SQL procedure that takes a DEPTNO as input and displays information about the
department, including the department name, location, and the count of employees in that department. Write a PL/SQL block to execute this procedure for a specified DEPTNO.

```
CREATE OR REPLACE PROCEDURE Display_Department_Info(p_deptno IN NUMBER)
IS
  v_department_name VARCHAR2(50);
  v_location VARCHAR2(50);
```

```
  v_employee_count NUMBER;
BEGIN
  -- Retrieve department information and employee count
  SELECT d.dname, d.loc, COUNT(e.empno)
  INTO v_department_name, v_location, v_employee_count
  FROM dept d
  LEFT JOIN emp e ON d.deptno = e.deptno
  WHERE d.deptno = p_deptno
  GROUP BY d.dname, d.loc;

  -- Display the result
  DBMS_OUTPUT.PUT_LINE('Department Name: ' || v_department_name);
  DBMS_OUTPUT.PUT_LINE('Location: ' || v_location);
  DBMS_OUTPUT.PUT_LINE('Employee Count: ' || v_employee_count);
END Display_Department_Info;
/

-- PL/SQL block to execute the procedure
DECLARE
  v_deptno NUMBER := 20; -- Specify the desired department number
BEGIN
  Display_Department_Info(v_deptno);
END;
/
```