**JAGAN INSTITUTE OF MANAGEMENT STUDIES**

**SECTOR – 5, ROHINI,NEW DELHI**



**(Affiliated to)**

**GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY**

**SECTOR – 16 C, DWARKA, NEW DELHI**



**PRACTICAL FILE : DATABAS EMANAGEMENT SYSTEMS**

Submitted To : Dr. Deepti Khanna Submitted By: Kartik Sharma

Professor(IT) Enrollment No.: 01914004424

MCA 1st Year [Section-A(1st Semester)]

# 

# ACKNOWLEGEMENT

I take this opportunity to present my votes of thanks to my faculty who really acted as pillars to help my way throughout the execution of lab exercises that has led to successful and satisfactory completion of the study of Subject (Database Management Systems).

I feel great sense of gratitude for **Dr. Deepti Khanna** under whose guidance and motivation this work has been performed.

I would also like to express my thanks to all **lab assistants** for giving me opportunity to work under their esteemed guidance. This project would not have completed without their guidance and coordination.

The inspiration of the faculty members of the Information Technology Department of JIMS Rohini (Sec-5) enabled me to make a thorough study of these subjects.

**Student Name : Kartik Sharma**

**Enrollment No. : 01914004424**

**CERTIFICATE**

This is certified to be the bonafide work of the student,

Name:**Kartik Sharma**, Enrollment No.:01914004424for the purpose of subject OOPS and Java Programing of MCA, Ist semester under the supervision of Dr. Deepti Khanna during the academic year 2024-2026.

**Dr. Deepti Khanna**

**Professor (IT)**

**JIMS, Rohini**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Problem Statement** | **Topic** | **Faculty Sign** | **Remarks** |
| 1 | E-Commerce system with CLIENT, PRODUCT and  SALESMAN tables and various operations on them | DDL,DQL and DML |  |  |
| 2 | College Library System with BRANCH, STUDENT,  AUTHOR, BOOK and BORROW tables and various  operations on them | DDL, DQL and DML |  |  |
| 3 | Write a PL/SQL program which will display your details. | PL/SQL basics |  |  |
| 4 | Write a PL/SQL program to accept 2 numbers and display  which one of them is greater. | PL/SQL basics |  |  |
| 5 | Write a PL/SQL program which will display total salary if salary is selected from employee where employee ID is 101, where salary = salary\*0.5 | PL/SQL basics |  |  |
| 6 | Write a PL/SQL program which will accept salary from the  user. If the salary is greater than 10000,it will display High  salary. If the salary is greater than 5000, it will display  Moderate salary, else it will display Low salary. | CONTROLS |  |  |
| 7 | Write a PL/SQL program which will accept age from user. If  age is greater than 18 and less than 61, the person is  eligible for employment. | CONTROLS |  |  |
| 8 | Write a PL/SQL program which will display an exception when no data is found. | EXCEPTION  HANDLING |  |  |
| 9 | Write a PL/SQL program which will display an exception whenever user divides the number by zero. | EXCEPTION  HANDLING |  |  |
| 10 | Create a cursor which will display first name of the employee. CURSOR |  |  |  |
| 11 | Create a cursor which will fetch employee name and their  salary. | CURSOR |  |  |
| 12 | Create a PL/SQL procedure which will calculate total salary  where salary also includes commission. | PROCEDURE |  |  |

**DBMS ASSIGNMENT-2 (PL/SQL)**

**--KARTIK SHARMA--**

1. **Write a PL/SQL program which will display your details.**

SQL> SET SERVEROUTPUT ON;

SQL> BEGIN

2 DBMS\_OUTPUT.PUT\_LINE('Name:Kartik Sharma');

3 DBMS\_OUTPUT.PUT\_LINE('Age: 22');

4 DBMS\_OUTPUT.PUT\_LINE('Department: IT');

5 END;

6 /

**OUTPUT:**

Name:Kartik Sharma

Age: 22

Department: IT

PL/SQL procedure successfully completed.

1. **Write a PL/SQL program to accept 2 numbers and display which one of them is greater.**

SQL> DECLARE

2 num1 NUMBER := &num1;

3 num2 NUMBER := &num2;

4 BEGIN

5 IF num1 > num2 THEN

6 DBMS\_OUTPUT.PUT\_LINE('The greater number is ' || num1);

7 ELSE

8 DBMS\_OUTPUT.PUT\_LINE('The greater number is ' || num2);

9 END IF;

10 END;

11 /

**OUTPUT:**

Enter value for num1: 21

old 2: num1 NUMBER := &num1;

new 2: num1 NUMBER := 21;

Enter value for num2: 13

old 3: num2 NUMBER := &num2;

new 3: num2 NUMBER := 13;

The greater number is 21

1. **Write a PL/SQL program which will display total salary if salary is selected from employee where employee ID is 101, where salary = salary\*0.5**

**EMP\_ID FNAME LNAME SALARY**

**--------------------------------------------------------------**

**101 Kartik Sharma 25000**

**102 Manpreet Singh 30000**

**103 Anjali Arora 40000**

**104 Utsav Sharma 37000**

SQL> DECLARE

2 total\_salary NUMBER;

3 BEGIN

4 SELECT salary \* 1.5 INTO total\_salary

5 FROM employee

6 WHERE emp\_id = 101;

7

8 DBMS\_OUTPUT.PUT\_LINE('Total salary for emp\_id 101: ' || total\_salary);

9 END;

10 /

**OUTPUT:**

Total salary for emp\_id 101: 37500

PL/SQL procedure successfully completed.

**4. Write a PL/SQL program which will accept salary from the user. If the salary is greater than 10000,it will display High salary. If the salary is greater than 5000, it will display Moderate salary, else it will display Low salary.**

SQL> DECLARE

2 salary NUMBER := &salary; BEGIN

3 IF salary > 30000 THEN

4 DBMS\_OUTPUT.PUT\_LINE('High salary');

5 ELSIF salary > 20000 THEN

6 DBMS\_OUTPUT.PUT\_LINE('Moderate salary');

7 ELSE

8 DBMS\_OUTPUT.PUT\_LINE('Low salary');

9 END IF;

10 END;

11 /

**OUTPUT:**

Enter value for salary: 50000

old 2: salary NUMBER := &salary; BEGIN

new 2: salary NUMBER := 50000; BEGIN

High salary

PL/SQL procedure successfully completed.

1. **Write a PL/SQL program which will accept age from user. If age is greater than 18 and less than 61, the person is eligible for employment**..

SQL> DECLARE

2 age NUMBER := &age;

3 BEGIN

4 IF age > 18 AND age < 61 THEN

5 DBMS\_OUTPUT.PUT\_LINE('Eligible for employment');

6 ELSE

7 DBMS\_OUTPUT.PUT\_LINE('Not eligible for employment');

8 END IF;

9 END;

10 /

**OUTPUT:**

Enter value for age: 45

old 2: age NUMBER := &age;

new 2: age NUMBER := 45;

Eligible for employment

PL/SQL procedure successfully completed.

1. **Write a PL/SQL program which will display an exception when no data is found.**

SQL> DECLARE

2 emp\_salary NUMBER;

3 BEGIN

4 BEGIN

5 SELECT salary INTO emp\_salary

6 FROM employee

7 WHERE emp\_id = 999; -- Non-existent ID for testing

8 EXCEPTION

9 WHEN NO\_DATA\_FOUND THEN

10 DBMS\_OUTPUT.PUT\_LINE('No data found for the given employee ID');

11 END;

12 END;

13 /

**OUTPUT:**

No data found for the given employee ID.

PL/SQL procedure successfully completed.

1. **Write a PL/SQL program which will display an exception whenever user divides the number by zero.**

SQL> DECLARE

2 num1 NUMBER := &num1;

3 num2 NUMBER := &num2;

4 result NUMBER;

5 BEGIN

6 BEGIN

7 result := num1 / num2;

8 DBMS\_OUTPUT.PUT\_LINE('Result: ' || result);

9 EXCEPTION

10 WHEN ZERO\_DIVIDE THEN

11 DBMS\_OUTPUT.PUT\_LINE('Error: Division by zero is not allowed');

12 END;

13 END;

14 /

**OUTPUT:**

Enter value for num1: 15

old 2: num1 NUMBER := &num1;

new 2: num1 NUMBER := 15;

Enter value for num2: 3

old 3: num2 NUMBER := &num2;

new 3: num2 NUMBER := 3;

Result: 5

PL/SQL procedure successfully completed.

1. **Create a cursor which will display first name of the employee.**

SQL> DECLARE

2 CURSOR emp\_cursor IS

3 SELECT fname FROM employee;

4 emp\_fname employee.fname%TYPE;

5 BEGIN

6 OPEN emp\_cursor;

7 LOOP

8 FETCH emp\_cursor INTO emp\_fname;

9 EXIT WHEN emp\_cursor%NOTFOUND;

10 DBMS\_OUTPUT.PUT\_LINE('First Name: ' || emp\_fname);

11 END LOOP;

12 CLOSE emp\_cursor;

13 END;

14 /

**OUTPUT:**

First Name: Kartik

First Name: Manpreet

First Name: Anjali

First Name: Utsav

PL/SQL procedure successfully completed.

1. **Create a cursor which will fetch employee name and their salary.**

SQL> DECLARE

2 CURSOR emp\_cursor IS

3 SELECT fname, salary FROM employee;

4 emp\_fname employee.fname%TYPE;

5 emp\_salary employee.salary%TYPE;

6 BEGIN

7 OPEN emp\_cursor;

8 LOOP

9 FETCH emp\_cursor INTO emp\_fname, emp\_salary;

10 EXIT WHEN emp\_cursor%NOTFOUND;

11 DBMS\_OUTPUT.PUT\_LINE('Name: ' || emp\_fname || ', Salary: ' || emp\_salary);

12 END LOOP;

13 CLOSE emp\_cursor;

14 END;

15 /

**OUTPUT:**

Name: Kartik , Salary: 25000

Name: Manpreet, Salary: 30000

Name: Anjali, Salary: 40000

Name: Utsav, Salary: 37000

PL/SQL procedure successfully completed**.**

1. **Create a PL/SQL procedure which will calculate total salary where salary also includes commission.**

SQL> CREATE OR REPLACE PROCEDURE calculate\_total\_salary (

2 emp\_id IN NUMBER,

3 commission IN NUMBER,

4 total\_salary OUT NUMBER

5 ) AS

6 emp\_salary NUMBER;

7 BEGIN

8 SELECT salary INTO emp\_salary

9 FROM employee

10 WHERE emp\_id = emp\_id;

11

12 total\_salary := emp\_salary + commission;

13 END;

14 /

**OUTPUT:**

Procedure created

.

PL/SQL procedure successfully completed**.**