

Create Tables for the following schemas

EMPLOYEE (eno, ename, sal, age, dno)

Department (dno, dname, location)

Constraints

1. Consider 4 departments in Department Table (10,20,30,40)
 2. Insert employees details for at least 4 members in each department.
 3. Dno is primary key in Department Table
 4. Eno is primary key in Employee Table
 5. Dno in Employee Table Referring to Dno in the Department Table
- ② write a PL/SQL procedure or function to display output as follows

Number of employees : 15 (sample output)

Number of Departments : 4

Average salary in the Employee Table : 65000 (sample)

Employee Details who are getting more salary than the average salary of the Employee Table.

(Sample output)

E.no	Ename	Salary	Age	D.No
501	Amitash	56000	25	5

① Table Creation

Create Table Employees

```
(eno number PRIMARY KEY, ename varchar2(30),
    sal number, age number(3), dno number,
    FOREIGN KEY dno references DEPARTMENT(dno))
```

CREATE TABLE DEPARTMENT

```
(dno number PRIMARY KEY,
    dname varchar2(30),
    location varchar2(50));
```

PL/SQL Procedure

Create or Replace PROCEDURE display
Insert

→ Insert into EMPLOYEES

```
(101, "Tarakesh", *40000, 20, 20);
```

→ Insert into EMPLOYEES

```
(102, "Abhinav", 45000, 20, 30);
```

→ Insert into DEPARTMENT

```
(20, "CSE", "VastuNagar");
```

②

→ Insert into Department.

(30, "ECE", "Zbrahimpatnam");

② PL/SQL Procedure

CREATE OR REPLACE PROCEDURE display

IS

no number;
depno number;
avgsal number;
cursor employeesalary IS select * from EMPLOYEES
WHERE SAL > (SELECT AVG(SAL) FROM EMPLOYEES);
demo EMPLOYEES%rowtype;

BEGIN

Select count(*) into no from employees;
Select count(*) into depno from DEPARTMENT;
Select Avg(sal) into avgsal from EMPLOYEES;
dbms^{output}.put_line ("Number of employees : " || no);
dbms^{output}.put_line ("Number of departments : " || deptno);
dbms^{output}.put_line ("Average salary in the Employee Table : "
|| avgsal);

dbms^{output}.put_line ("Employee Details who are getting
more salary than the average
of the Employee Table : ");

③

OPEN employee salary;

LOOP
FOR

LOOP

Fetch employeesalary into demo;
EXIT WHEN employeesalary%notfound;
dbms_output.put_line(demo.eno || demo.ename ||
demo.sal || demo.age || demo.dno);

END LOOP;

CLOSE employee salary;

END;

/

o);