PL/SQL Assignment

PL/SQL Collection & Object Oriented & Debugging

support@intellipaat.com
+91-7022374614
US: 1-800-216-8930(Toll Free)

EXERCISE

Writ a PL/SQL procedure to find the number of students ranging from 100-70%, 69-60%, 59-50% & below 49% in each course from the student_course table given by the procedure as parameter.

SOLUTION:

```
SQL> select * from student_enrollment;

   ROLL_NO COURSE     COURSE_COD        SEM TOTAL_MARKS PERCENTAGE
---------- ---------- ---------- ---------- ----------- ----------
       111 cs         1001                1         300         50
       112 cs         1001                1         400         66
       113 is         1002                1         465         77
       114 is         1002                1         585         97

SQL> get e:/p13.sql;

create or replace procedure rank(crc varchar)
is
dis number:=0;
first number:=0;
sec number:=0;
pass number:=0;
cursor st is select * from student_enrollment;
r st%rowtype;
begin
open st;
loop
fetch st into r;
exit when st%notfound;
if(r.course=crc)
then
if(r.percentage>=70 and r.percentage<=100)
then
dis:=dis+1;
end if;
 if(r.percentage>=60 and r.percentage<70)
then
first:=first+1;
end if;
if(r.percentage>=50 and r.percentage<60)
then
```

```
 26  sec:=sec+1;
 27  end if;

 28  if(r.percentage>=35 and r.percentage<50)
 29  then

 30  pass:=pass+1;
 31  end if;

 32  end if;

 33  end loop;
 34  close st;

 35  dbms_output.put_line('distinction is '||dis);
 36  dbms_output.put_line('first class is '||first);

 37  dbms_output.put_line('second class is '||sec);
 38  dbms_output.put_line('just pass is '||pass);

 39* end;

 40  .

SQL> /


Procedure created.


SQL> exec rank('cs');


distinction is 0

first class is 1


second class is 1


just pass is 0


PL/SQL procedure successfully completed.


SQL> exec rank('is');


distinction is 2

first class is 0

second class is 0


just pass is 0


PL/SQL procedure successfully completed.
```

EXERCISE

Create a store function that accepts 2 numbers and returns the addition of passed values. Also write the code to call your function.

SOLUTION:

```
SQL> get e:/p14.sql;

  1  create or replace function addition(a number,b number)
  2  return number
  3  is
  4  begin
  5  dbms_output.put('the sum of '||a||' and '||b||' is :');
  6  return (a+b);
  7* end;

  8  .
SQL> /

Function created.

SQL> begin

  2  dbms_output.put_line(addition(6,78));
  3  end;
  4  .
SQL> /

the sum of 6 and 78 is: 84

PL/SQL procedure successfully completed.
```

EXERCISE

Write a PL/SQL function that accepts department number and returns the total salary of the department. Also write a function to call the function.

SOLUTION:

```
SQL> select * from works;


    EMP_NO COMPANY_NAME  JOINING_D DESIGNATION        SALARY
                         DEPTNO

           -------------------- --------- -------------------- ---------
----------1  abc          23-NOV-00  project lead        40000         1

        2    abc          25-DEC-10  software engg       20000         2

        3    abc          15-JAN-11  software engg       1900          1

        4    abc          19-JAN-11  software engg       19000         2

        5    abc          06-FEB-11  software engg       18000         1


SQL> get e:/plsql/p15.sql;

  1  create or replace function tot_sal_of_dept(dno number)
  2  return number

  3  is
  4  tot_sal number:=0;

  5  begin

  6  select sum(salary) into tot_sal from works where deptno=dno;
  7  return tot_sal;
  8* end;

SQL> .

SQL> /


Function created.


SQL> begin


  2  dbms_output.put_line('Total salary of DeptNo 1 is :'
     || tot_sal_of_dept(1));

  3  end;
  4  .

SQL> set serveroutput on;
SQL> /

Total salary of DeptNo 1 is :77000


PL/SQL procedure successfully completed.


SQL> begin


  2  dbms_output.put_line('total salary of dept 2
is :'||tot_sal_of_dept(2));

  3  end;

  4  .
SQL> /

Total salary of DeptNo 2 is :39000


PL/SQL procedure successfully completed.
```

EXERCISE 16

Write a PL/SQL code to create,

a) Package specification

b) Package body.

For the insert, retrieve, update and delete operations on a student table.

SOLUTION:

```
SQL> get e:/plsql/l16p.sql;

  1  create or replace package alloperation
  2  is
  3  procedure forinsert(rno number,sname varchar,crc
     varchar,gen varchar);
  4  procedure forretrive(rno number);
  5  procedure forupdate(rno number,sname varchar);
  6  procedure fordelete(rno number);
  7* end alloperation;
SQL> .

SQL> /

Package created.

SQL> get e:/plsql/l16pbody.sql;

  1  create or replace package body alloperation
  2  is
  3  procedure forinsert(rno number,sname varchar,crc
     varchar,gen varchar)
  4  is
  5  begin
  6  insert into student values(rno,sname,crc,gen);
  7  end forinsert;
  8  procedure forretrive(rno number)
  9  is
 10  sname student.student_name%type;
 11  crc student.course%type;
 12  gen student.gender%type;
 13  begin
 14  select student_name,course,gender into sname,crc,gen
 15  from student where roll_no=rno;
 16  dbms_output.put_line(sname||' '||crc||' '||gen);
 17  end forretrive;
 18  procedure forupdate(rno number,sname varchar)
 19  is
 20  begin
 21  update student set student_name=sname where roll_no=rno;
 22  end forupdate;
 23  procedure fordelete(rno number)
 24  is
 25  begin
 26  delete student where roll_no=rno;
 27  end fordelete;
 28* end alloperation;
 29  .
SQL> /

Package body created.


SQL> select * from student;


   ROLL_NO STUDENT_NAME         COURS GENDER
---------- -------------------- ----- ----------
       111 ravi                 cs    male
       112 praveen              cs    male
       113 bhuvana              is    female
       114 apparna              is    female

SQL> begin

  2  alloperation.forinsert(444,'vivekananda','ec','male');
  3  alloperation.forretrive(444);
  4  alloperation.forupdate(111,'swamy');
  5  end;
  6  .
SQL> /

vivekananda ec male


PL/SQL procedure successfully completed.


SQL> select * from student;
```

```
   ROLL_NO STUDENT_NAME         COURS GENDER
---------- -------------------- ----- ----------
       111 swamy                cs    male
       112 praveen              cs    male
       113 bhuvana              is    female
       114 apparna              is    female
       444 vivekananda          ec    male

SQL> begin
  2  alloperation.fordelete(444);
  3  end;
  4  .
SQL> /

PL/SQL procedure successfully completed.

SQL> select * from student;

   ROLL_NO STUDENT_NAME         COURS GENDER
---------- -------------------- ----- ----------
       111 swamy                cs    male
       112 praveen              cs    male
       113 bhuvana              is    female
       114 apparna              is    female
```