Name: Samriddhi Verma
Reg.No.: 16BCE1375

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Programme	:	**B.Tech - CSE**	Semester	:	**Winter 18 - 19**
Course	:	**Database Management Systems (Embedded Lab)**	Code	:	**CSE2004**
Faculty	:	**Prof. A. Vijayalakshmi** **Prof. M. Premalatha**	Slot	:	**L3 + L4**

Ex. No: 8
11-03-19

PL-SQL Basics and Procedures

1. **Write a PL/SQL structure to find the greatest number among three numbers using if-else.**

Code:

```
SQL> declare
  2  a number;
  3  b number;
  4  c number;
  5  begin
  6  dbms_output.put_line('Enter the three numbers to compare:');
  7  a:=&a;
  8  b:=&b;
  9  c:=&c;
 10  dbms_output.put_line('a='||a||' b='||b||' c='||c);
 11  if a>b AND a>c
 12  then
 13  dbms_output.put_line('a is greatest');
 14  else
 15  if b>a AND b>c
 16  then
 17  dbms_output.put_line('b is greatest');
 18  else
 19  dbms_output.put_line('c is greatest');
 20  end if;
 21  end if;
 22  end;
 23  /
Enter value for a: 2
old   7: a:=&a;
new   7: a:=2;
Enter value for b: 4
old   8: b:=&b;
new   8: b:=4;
Enter value for c: 1
```

```
old   9: c:=&c;
new   9: c:=1;
b is greatest

PL/SQL procedure successfully completed.
```

2. **Write a PL/SQL structure to find a given number is Armstrong number or not using loops.**
   [Individual digits cube is summed – result will be the original number]

   Example 1:

   a=153

   b= $1^3+5^3+3^3$= 1+125+27=153

   a=153 is equal to b=153, so 153 is an Armstrong number.

   Example 2:

   a=121
   b= $1^3+2^3+1^3$= 1+8+1=10
   a=121 is not equal to b=10, so 121 is not an Armstrong number.

   CODE:

```
SQL> declare
  2  n number:=153;
  3  s number:=0;
  4  r number;
  5  len number;
  6  m number;
  7  begin
  8  m:=n;
  9  len:=length(to_char(n));
 10  while n>0
 11  loop
 12   r:=mod(n,10);
 13   s:=s+power(r,len);
 14   n:=trunc(n/10);
 15  end loop;
 16  if m=s
 17  then
 18   dbms_output.put_line('armstrong number');
 19  else
 20   dbms_output.put_line('not armstrong number');
```

```
end if;
end;
/
armstrong number

PL/SQL procedure successfully completed.
```

3. **Write a PL/SQL structure to find addition, subtraction, multiplication and division of two numbers using ‘case’.**

Code:

```
declare
a NUMBER :=55;
b NUMBER :=5;
arth_operation VARCHAR2(20):='MULTIPLY';
begin
case (arth_operation)
WHEN 'ADD' THEN dbms_output.put_line('Addition of the numbers are: '||
a+b );
WHEN 'SUBTRACT' THEN dbms_output.put_line('Subtraction of the numbers
are: '||a-b );
WHEN 'MULTIPLY' THEN dbms_output.put_line('Multiplication of the
numbers are: '|| a*b);
WHEN 'DIVIDE' THEN dbms_output.put_line('Division of the numbers are:
'|| a/b);
ELSE dbms_output.put_line('No operation action defined. Invalid
operation');
end case;
dbms_output.put_line('Program completed.');
end;
/
Multiplication of the numbers are: 275

PL/SQL procedure successfully completed.
```

Name: Samriddhi Verma
Reg.No.: 16BCE1375

4. **Write PLSQL for the following**

   An employee is identified by employee number and has attributes employee name, salary, date of joining, who works at the department identified by department number and has attribute department name. Consider the scenario and create table for each relations and associations with key constraints

- **Insert 5 records into the employee table**

Code:

```
SQL> create table employee(empid varchar2(8), empname
char(10),salary number(10),doj date,constraint pk_empp primary
key(empid));
Table created.

SQL> create table department(deptid varchar2(8), deptname
char(10),constraint pk_deptt primary key(deptid));

Table created.

SQL> create table empl_deptt(empid varchar(8),deptid varchar(8),
constraint fk_empp foreign key(empid) references emp,constraint
fk_deptt foreign key(deptid) references dept);

Table created.

SQL> create or replace procedure pro_insert(x1 varchar2,x2
char,x3 number,x4 date) as
     empid varchar(8);
     empname char(10);
     salary number(10);
     doj date;
     Begin
     empid:=x1;
     empname:=x2;
     salary:=x3;
     doj:=x4;
     insert into employee values(empid,empname,salary,doj);
     dbms_output.put_line('Record Inserted');
     end;
     /
Procedure created.
SQL> begin
     pro_insert('101','sri',50000,'10-dec-2017');
```

```
  3  end;
  4  /

PL/SQL procedure successfully completed.

SQL> begin
  2  pro_insert('102','sai',60000,'23-may-2017');
  3  end;
  4  /

PL/SQL procedure successfully completed.

SQL> begin
  2  pro_insert('103','som',70000,'16-oct-2016');
  3  end;
  4  /

PL/SQL procedure successfully completed.

SQL> begin
  2  pro_insert('104','ray',70000,'12-jun-2016');
  3  end;
  4  /

PL/SQL procedure successfully completed.

SQL> begin
  2  pro_insert('105','jay',40000,'05-aug-2015');
  3  end;
  4  /

PL/SQL procedure successfully completed.

SQL> select * from employee;

EMPID    EMPNAME        SALARY DOJ
-------- ---------- ---------- ---------
101      sri             50000 10-DEC-17
102      sai             60000 23-MAY-17
103      som             70000 16-OCT-16
104      ray             70000 12-JUN-16
105      jay             40000 05-AUG-15
```

- **Update the salary of sai.**

```
SQL> create or replace procedure pro_update(x1 in char,x2 in number) as
ename char(10);
sal number(10);
Begin

ename:=x1;
sal:=x2;
update employee set salary=sal where empname=ename;
if SQL%FOUND then
dbms_output.put_line('Record Updated');
else
dbms_output.put_line('No Record Found');
end if;
end;
/

Procedure created.

SQL> exec pro_update('sai',80000);

PL/SQL procedure successfully completed.

SQL> select * from employee;

EMPID    EMPNAME        SALARY DOJ
-------- ---------- ---------- ---------
102      sai             80000 23-MAY-17
103      som             70000 16-OCT-16
104      ray             70000 12-JUN-16
105      jay             40000 05-AUG-15
```

Name: Samriddhi Verma
Reg.No.: 16BCE1375

- **Delete the employee with employee number 101.**

```
SQL> create or replace procedure pro_del(x1 varchar2) as
  2  eid varchar2(8);
  3
  4  begin
  5
  6  eid:=x1;
  7  delete from employee where empid=eid;
  8  dbms_output.put_line('Record Deleted');
  9
 10  end;
 11  /

Procedure created.

SQL> exec pro_del(101);

PL/SQL procedure successfully completed.

SQL> select * from employee;

EMPID    EMPNAME        SALARY DOJ
-------- ---------- ---------- ---------
102      sai             60000 23-MAY-17
103      som             70000 16-OCT-16
104      ray             70000 12-JUN-16
105      jay             40000 05-AUG-15
```