CSE5011 –

Database Systems and Design

Assessment – 4

Cycle sheet –II - PL/SQL

(Ex-7, Ex-8, Ex-9 and Ex-10)

Rahul Pathak

21MAI0069

Vellore Institute of Technology

Cycle sheet –II - PL/SQL

Consider the following schema for PL/SQL programming:

Table Name: Employee

Attribute	Data Type
First Name	VARCHAR(15)
Mid Name	CHAR(2)
Last Name	VARCHAR(15)
SSN Number	CHAR(9)
Birthday	DATE
Address	VARCHAR(50)
Sex	CHAR(1)
Salary	NUMBER (7)
Supervisor SSN	CHAR(9)
Department Number	NUMBER (5)

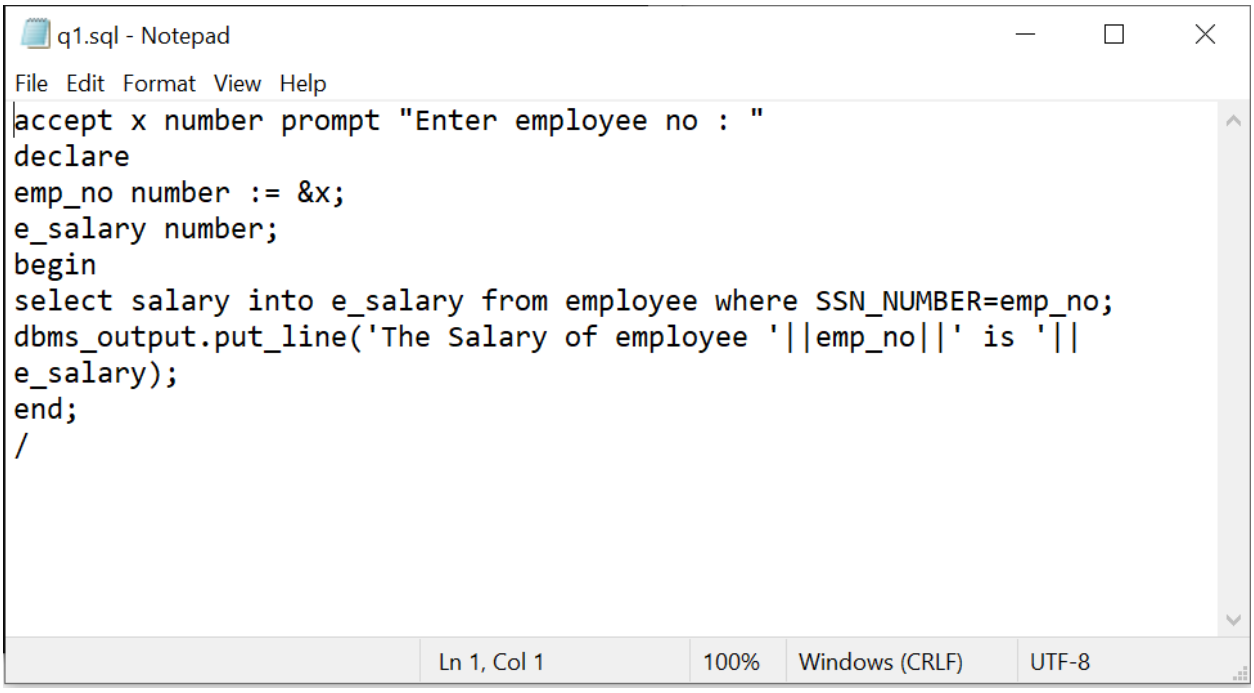
Table Name: Department

Attribute	Data Type
Department Name	Varchar(15)
Department Number	Number(5)
ManagerSSN	CHAR(9)
ManageStartDate	DATE

Exercise - VII:

Aim: To understand the concept of PL/SQL Programming:

1. **Write a PL/SQL block to accept an empno and display the salary of the person.**



```
SQL> set serveroutput on;
SQL> @D:\pl_sql\q1.sql
Enter employee no : 123
old   2: emp_no number := &x;
new   2: emp_no number :=        123;
The Salary of employee 123 is 80000

PL/SQL procedure successfully completed.

SQL>
```

2. Write a PL/SQL program to delete one record in employee table.

q2.sql - Notepad

File Edit Format View Help

```
accept x number prompt "Enter employee SSN to delete : "
declare
emp_no number := &x;
begin
delete from employee where SSN_NUMBER=emp_no;
dbms_output.put_line('Employee record with SSN = '||emp_no||'
deleted');
end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> edit D:\pl_sql\q2.sql

SQL> @D:\pl_sql\q2.sql
Enter employee SSN to delete : 999
old   2: emp_no number := &x;
new   2: emp_no number :=        999;
Employee record with SSN = 999 deleted

PL/SQL procedure successfully completed.

SQL>
```

3. Write a program to delete employee details who are having age >60.

q3.sql - Notepad

File Edit Format View Help

```
begin

for row in
(select SSN_number from employee where TRUNC((SYSDATE -
BIRTHDAY)/365.35) > 60) loop

dbms_output.put_line('Employee record with SSN = '||row.SSN_number||'
deleted');

end loop;


delete from employee where TRUNC((SYSDATE - BIRTHDAY)/365.35) > 60;

end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> edit D:\pl_sql\q3.sql

SQL> @D:\pl_sql\q3.sql
Employee record with SSN = 999       deleted
Employee record with SSN = 998       deleted

PL/SQL procedure successfully completed.

SQL>
```

4. Write a PL/SQL block to display employees must make a minimum salary of $1,000.

q4.sql - Notepad

File Edit Format View Help

```
declare
i number := 1;

begin

for row in (select first_name, last_name, salary from employee where
salary>1000 order by salary asc) loop

dbms_output.put_line(i||'. '||row.first_name||' '||row.last_name||'
'||row.salary);

i := i+1;
end loop;
end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> @D:\pl_sql\q4.sql
1. John Smith 30000
2. Ramesh Narayan 38000
3. Frankin Wong 40000
4. Jennifer Wallace 43000
5. Joyce PAN 70000
6. Doug Gilbert 80000

PL/SQL procedure successfully completed.

SQL>
```

5. Write a PL/SQL to delete a records whose basic salary is <2000 from Emp table.

q5.sql - Notepad

File Edit Format View Help

```
begin

for row in
(select SSN_number, salary from employee where salary<2000) loop

dbms_output.put_line('Employee record with SSN = '||
row.SSN_number||'and salary = '||row.salary||' deleted');

end loop;

delete from employee where salary<2000;

end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> edit D:\pl_sql\q5.sql

SQL> @D:\pl_sql\q5.sql
Employee record with SSN = 998      and salary = 1100 deleted

PL/SQL procedure successfully completed.

SQL>
```

Exercise - VIII:

Aim: To know the usage of different sequential control structures in PL/SQL effective programming.

1. Write a PL/SQL block to find the greatest of three numbers.

q8_1.sql - Notepad

File Edit Format View Help

```
declare
a number;
b number;
c number;
begin
a:=&a;
b:=&b;
c:=&c;

if(a>b) and (a>c) then
dbms_output.put_line(a ||' is Greatest!');
elsif(b>c) then
dbms_output.put_line(b ||' is Greatest!');
else
dbms_output.put_line(c ||' is Greatest!');
end if;

end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> edit D:\pl_sql\q8_1.sql

SQL> @D:\pl_sql\q8_1.sql
Enter value for a: 23
old   6: a:=&a;
new   6: a:=23;
Enter value for b: 45
old   7: b:=&b;
new   7: b:=45;
Enter value for c: 26
old   8: c:=&c;
new   8: c:=26;
45 is Greatest!

PL/SQL procedure successfully completed.

SQL>
```

2. Write a PL/SQL code to print the student's grade accepting their marks in three subjects (hint use: case selector….)

q8_2.sql - Notepad

File Edit Format View Help

```
declare
x number := &subject_1;
y number := &subject_2;
z number := &subject_3;
average number;
grade varchar(5);
begin
average := round(((x+y+z)/3), 0);
grade := CASE
                when average>90 and average<=100 then 'S'
                when average>80 and average<90 then 'A'
                when average>70 and average<80 then 'B'
                when average>60 and average<70 then 'C'
                when average>50 and average<60 then 'D'
                when average>40 and average<50 then 'E'
                when average>0 and average<40 then 'F'
                ELSE 'NA'
        end;
        dbms_output.put_line('GRADE : '||grade);
end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> edit D:\pl_sql\q8_2.sql

SQL> @D:\pl_sql\q8_2.sql
Enter value for subject_1: 56
old   2: x number := &subject_1;
new   2: x number := 56;
Enter value for subject_2: 78
old   3: y number := &subject_2;
new   3: y number := 78;
Enter value for subject_3: 99
old   4: z number := &subject_3;
new   4: z number := 99;
GRADE : B

PL/SQL procedure successfully completed.

SQL>
```

Exercise – IX:

Aim: To understand the concepts of Iterations and Subprogram (Procedures and Functions)

Iterations:

1. Write a PL/SQL code to print the numbers in reverse order from 100 to 1.

q9_1.sql - Notepad

File Edit Format View Help

```
begin
for i in reverse 1 .. 100 loop
        dbms_output.put(i||' ');
end loop;
        dbms_output.new_line;
end;
/
```

```
SQL> edit D:\pl_sql\q9_1.sql

SQL> @D:\pl_sql\q9_1.sql
100 99 98 97 96 95 94 93 92 91 90 89 88 87 86 85 84 83 82 81 80 79 78 77 76 75
74 73 72 71 70 69 68 67 66 65 64 63 62 61 60 59 58 57 56 55 54 53 52 51 50 49 48
47 46 45 44 43 42 41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21
20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

PL/SQL procedure successfully completed.
```

2. Create a pl/sql block to find the sum of series 1+3+5+……+n.

q9_2.sql - Notepad

File Edit Format View Help

```
declare
n number := &n;
res number := 0;
begin
for i in 0 .. n loop
        if (mod(i,2) = 1) then
                res := res+i;
        end if;
end loop;
        dbms_output.put_line('Sum = '|| res);
end;
/
```

```
SQL> edit D:\pl_sql\q9_2.sql

SQL> @D:\pl_sql\q9_2.sql
Enter value for n: 7
old   2: n number := &n;
new   2: n number := 7;
Sum = 16

PL/SQL procedure successfully completed.

SQL>
```

Functions

3. Write a function to give the number of employees for a given Department name.

q9_3.sql - Notepad

File Edit Format View Help

```
declare
i number;
depn_no number;
no_of_emp number;

function no_of_emp_dept(dep_no number) return number is
begin
select count(*) into i from employee e, department d where
e.department_number = d.department_number and d.department_number =
dep_no;
return i;
end;

begin
depn_no := &depn_no;
no_of_emp := no_of_emp_dept(depn_no);
dbms_output.put_line('Total employee in department '||depn_no||' = '||
no_of_emp);
end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> edit D:\pl_sql\q9_3.sql

SQL> @D:\pl_sql\q9_3.sql
Enter value for depn_no: 3
old  13: depn_no := &depn_no;
new  13: depn_no := 3;
Total employee in department 3 = 1

PL/SQL procedure successfully completed.

SQL>
```

4. Write a PL/SQL to find the factorial of the given number using function.

q9_4.sql - Notepad

File Edit Format View Help

```
declare
n number;
f number;
res number;

function fact(x number) return number is
begin
        if x = 0 then
           f := 1;
        else
           f := x * fact(x-1);
        end if;
return f;
end;

begin
n := &n;
res := fact(n);
dbms_output.put_line('Factorial of '||n||' = '||res);
end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> edit D:\pl_sql\q9_4.sql

SQL> @D:\pl_sql\q9_4.sql
Enter value for n: 4
old  17: n := &n;
new  17: n := 4;
Factorial of 4 = 24

PL/SQL procedure successfully completed.

SQL>
```

Procedure

5. Write a procedure to accept an employee name and display his Department names.

q9_5.sql - Notepad

File Edit Format View Help

```
create or replace procedure get_dept_name
(emp_name in employee.first_name%type,
dept_name out department.department_name%type)
is
begin
select department_name into dept_name from employee e, department d
where e.department_number = d.department_number and e.first_name =
emp_name;
end get_dept_name;
/
```

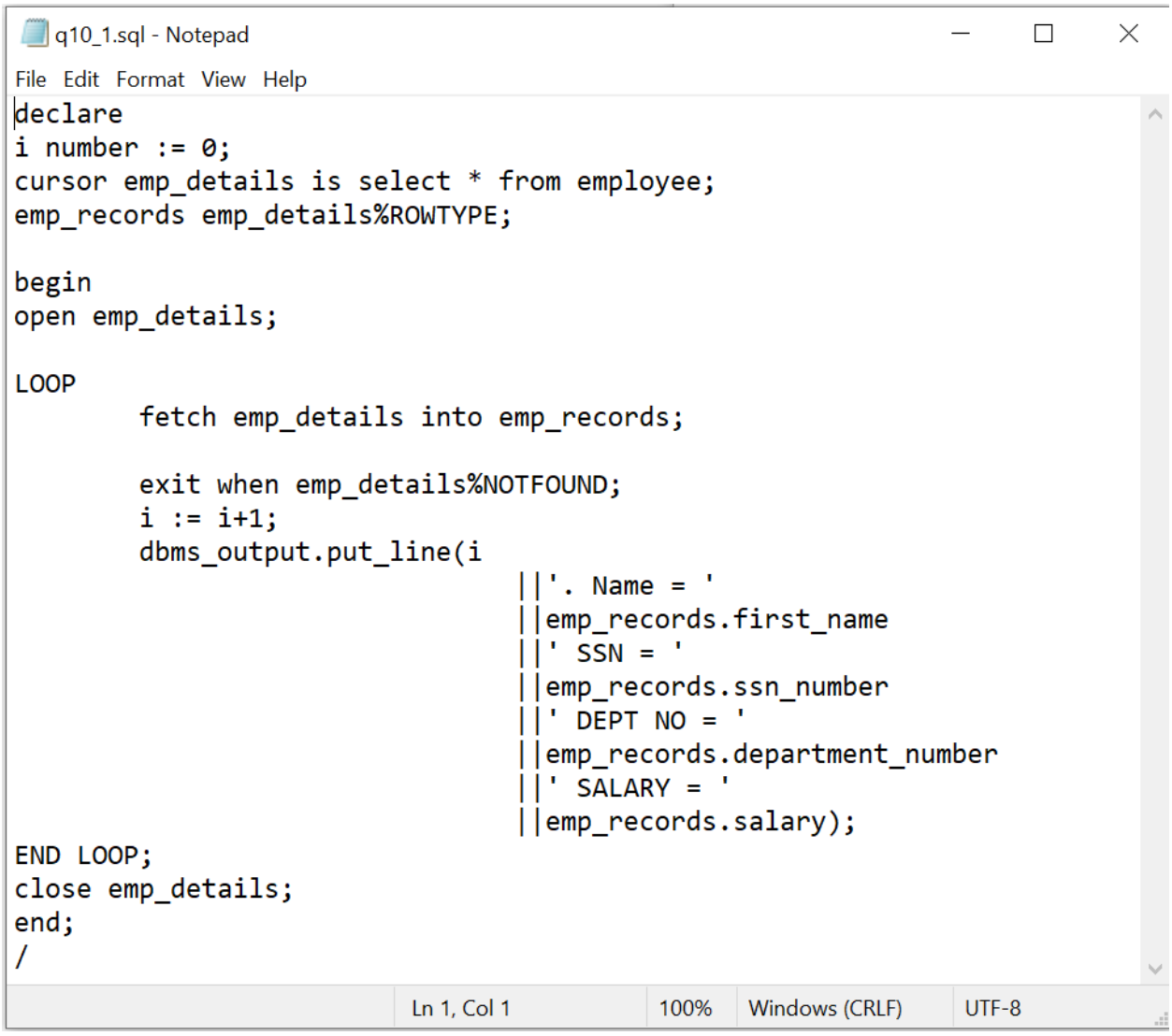
```
SQL> edit D:\pl_sql\q9_5.sql

SQL> @D:\pl_sql\q9_5.sql

Procedure created.

SQL> variable dept_name varchar2(50);
SQL> execute get_dept_name('John', :dept_name);

PL/SQL procedure successfully completed.

SQL> print dept_name;

DEPT_NAME
--------------------------------------------------
Headquarter

SQL>
```

Exercise - X:

Cursor

Aim: To understand implicit and explicit cursor in PL/SQL

1. **Retrieve the employee details using cursors.**



```
SQL> edit D:\pl_sql\q10_1.sql

SQL> @D:\pl_sql\q10_1.sql
1. Name = Doug SSN = 123       DEPT NO = 1 SALARY = 80000
2. Name = Joyce SSN = 124       DEPT NO =  SALARY = 70000
3. Name = Frankin SSN = 125       DEPT NO = 2 SALARY = 40000
4. Name = Jennifer SSN = 564       DEPT NO = 2 SALARY = 43000
5. Name = John SSN = 678       DEPT NO = 1 SALARY = 30000
6. Name = Ramesh SSN = 234       DEPT NO = 3 SALARY = 38000

PL/SQL procedure successfully completed.

SQL>
```

2. Write a cursor program to display all the employee and department details

q10_2.sql - Notepad

File Edit Format View Help

```
declare
i number := 0;
cursor emp_details is select e.first_name, e.ssn_number,
d.department_name, e.salary, d.manager_ssn from employee e, department
d where e.department_number = d.department_number;

emp_records emp_details%ROWTYPE;

begin
open emp_details;

dbms_output.put_line('SN'||' Name  '||'   SSN       '||'   DEPT Name
'||'       SALARY   '||'    MANAGER SSN   ');
LOOP
        fetch emp_details into emp_records;

        exit when emp_details%NOTFOUND;
        i := i+1;
        dbms_output.put_line(i ||'  '||emp_records.first_name
                        ||'   '||emp_records.ssn_number
                        ||' '||emp_records.department_name
                        ||'        '||emp_records.salary
                        ||'        '||emp_records.manager_ssn);
END LOOP;
close emp_details;
end;
/
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
SQL> edit D:\pl_sql\q10_2.sql

SQL> @D:\pl_sql\q10_2.sql
SN Name     SSN          DEPT Name          SALARY       MANAGER SSN
1  Doug   123       Headquarter        80000        678
2  Frankin   125       Administration        40000        564
3  Jennifer   564       Administration        43000        564
4  John   678       Headquarter        30000        678
5  Ramesh   234       Finance        38000        234

PL/SQL procedure successfully completed.

SQL>
```