Lab 6 : Dipesh Singh – 190905520

Question 1 : Write a PL/SQL block to display the GPA of given student.

```
DECLARE roll studenttable.rollno %TYPE;
gp studenttable.gpa %TYPE;
BEGIN roll := '&r';
select gpa into gp
from studenttable
where rollno = roll;
DBMS_OUTPUT.PUT_LINE(
    'The gpa for roll number : ' || TO_CHAR(roll) || ' is : ' || gp
);
END;
/
```

```
SQL> DECLARE roll studenttable.rollno %TYPE;
  2  gp studenttable.gpa %TYPE;
  3  BEGIN roll := '&r';
  4  select gpa into gp
  5  from studenttable
  6  where rollno = roll;
  7  DBMS_OUTPUT.PUT_LINE(
  8     'The gpa for roll number : ' || TO_CHAR(roll) || ' is : ' || gp
  9  );
 10  END;
 11  /
Enter value for r: 4
old   3: BEGIN roll := '&r';
new   3: BEGIN roll := '4';
The gpa for roll number : 4 is : 7.8

PL/SQL procedure successfully completed.

SQL>
```

Question 2 : Write a PL/SQL block to display the letter grade(0-4: F; 4-5: E; 5-6: D; 6-7: C, 7-8: B; 8-9: A; 9-10: A+} of given student.

```
declare roll studenttable.rollno %type;
gp studenttable.gpa %type;
grade varchar(2);
begin roll := '&r';
select gpa into gp
from studenttable
where rollno = roll;
if gp > 0
and gp <= 4 then grade := 'F';
elsif gp > 4
and gp <= 5 then grade := 'E';
elsif gp > 5
```

```
and gp <= 6 then grade := 'D';
elsif gp > 6
and gp <= 7 then grade := 'C';
elsif gp > 7
and gp <= 8 then grade := 'B';
elsif gp > 8
and gp <= 9 then grade := 'A';
else grade := 'A+';
end if;
dbms_output.put_line(
    'The grade for the roll number : ' || to_char(roll) || ' is : ' || gr
ade
);
end;
/
```

```
SQL> declare roll studenttable.rollno %type;
  2  gp studenttable.gpa %type;
  3  grade varchar(2);
  4  begin roll := '&r';
  5  select gpa into gp
  6  from studenttable
  7  where rollno = roll;
  8  if gp > 0
  9  and gp ≤ 4 then grade := 'F';
 10  elsif gp > 4
 11  and gp ≤ 5 then grade := 'E';
 12  elsif gp > 5
 13  and gp ≤ 6 then grade := 'D';
 14  elsif gp > 6
 15  and gp ≤ 7 then grade := 'C';
 16  elsif gp > 7
 17  and gp ≤ 8 then grade := 'B';
 18  elsif gp > 8
 19  and gp ≤ 9 then grade := 'A';
 20  else grade := 'A+';
 21  end if;
 22  dbms_output.put_line(
 23     'The grade for the roll number : ' || to_char(roll) || ' is : ' || grade
 24  );
 25  end;
 26  /
Enter value for r: 3
old   4: begin roll := '&r';
new   4: begin roll := '3';
The grade for the roll number : 3 is : F

PL/SQL procedure successfully completed.
```

Question 3 : Input the date of issue and date of return for a book. Calculate and display the fine with the appropriate message using a PL/SQL block. The fine is charged as per the table 8.1:

```
declare issue varchar(20);
retur varchar(20);
days number(5);
fine number(5);
begin issue := '&i';
retur := '&r';
days := to_date(retur, 'dd/mm/yy') - to_date(issue, 'dd/mm/yy');
dbms_output.put_line('The number of days is : ' || days);
if days <= 7 then fine := 0;
elsif days >= 8
and days <= 15 then fine := days;
elsif days >= 16
and days <= 30 then fine := 2 * days;
else fine := 5 * days;
end if;
dbms_output.put_line('The fine is : ' || fine);
end;
/
```

```
SQL> declare issue varchar(20);
  2  retur varchar(20);
  3  days number(5);
  4  fine number(5);
  5  begin issue := '&i';
  6  retur := '&r';
  7  days := to_date(retur, 'dd/mm/yy') - to_date(issue, 'dd/mm/yy');
  8  dbms_output.put_line('The number of days is : ' || days);
  9  if days ≤ 7 then fine := 0;
 10  elsif days ≥ 8
 11  and days ≤ 15 then fine := days;
 12  elsif days ≥ 16
 13  and days ≤ 30 then fine := 2 * days;
 14  else fine := 5 * days;
 15  end if;
 16  dbms_output.put_line('The fine is : ' || fine);
 17  end;
 18  /
Enter value for i: 10/01/20
old   5: begin issue := '&i';
new   5: begin issue := '10/01/20';
Enter value for r: 10/02/20
old   6: retur := '&r';
new   6: retur := '10/02/20';
The number of days is : 31
The fine is : 155

PL/SQL procedure successfully completed.
```

Question 4 : Write a PL/SQL block to print the letter grade of all the students(RollNo: 1 -5).

```
declare gp studenttable.gpa %type;
grade varchar(2);
begin for i in 1..5 loop
select gpa into gp
from studenttable
where rollno = i;
if gp > 0
and gp <= 4 then grade := 'F';
elsif gp > 4
and gp <= 5 then grade := 'E';
elsif gp > 5
and gp <= 6 then grade := 'D';
elsif gp > 6
and gp <= 7 then grade := 'C';
elsif gp > 7
and gp <= 8 then grade := 'B';
elsif gp > 8
and gp <= 9 then grade := 'A';
else grade := 'A+';
end if;
dbms_output.put_line(
    'The grade for the roll number : ' || to_char(i) || ' is : ' || grade
);
end loop;
end;
/
```

```
SQL> declare gp studenttable.gpa %type;
  2  grade varchar(2);
  3  begin for i in 1..5 loop
  4  select gpa into gp
  5  from studenttable
  6  where rollno = i;
  7  if gp > 0
  8  and gp ≤ 4 then grade := 'F';
  9  elsif gp > 4
 10  and gp ≤ 5 then grade := 'E';
 11  elsif gp > 5
 12  and gp ≤ 6 then grade := 'D';
 13  elsif gp > 6
 14  and gp ≤ 7 then grade := 'C';
 15  elsif gp > 7
 16  and gp ≤ 8 then grade := 'B';
 17  elsif gp > 8
 18  and gp ≤ 9 then grade := 'A';
 19  else grade := 'A+';
 20  end if;
 21  dbms_output.put_line(
 22     'The grade for the roll number : ' || to_char(i) || ' is : ' || grade
 23  );
 24  end loop;
 25  end;
 26  /
The grade for the roll number : 1 is : D
The grade for the roll number : 2 is : C
The grade for the roll number : 3 is : F
The grade for the roll number : 4 is : B
The grade for the roll number : 5 is : A+

PL/SQL procedure successfully completed.
```

Question 5 : Alter StudentTable by appending an additional column LetterGrade Varchar2(2). Then write a PL/SQL block to update the table with letter grade of each student.

```
alter table studenttable
add lettergrade varchar(2);
declare gp studenttable.gpa %type;
grade varchar(2);
begin for i in 1..5 loop
select gpa into gp
from studenttable
where rollno = i;
if gp > 0
and gp <= 4 then grade := 'F';
elsif gp > 4
and gp <= 5 then grade := 'E';
elsif gp > 5
and gp <= 6 then grade := 'D';
elsif gp > 6
and gp <= 7 then grade := 'C';
elsif gp > 7
and gp <= 8 then grade := 'B';
elsif gp > 8
and gp <= 9 then grade := 'A';
else grade := 'A+';
end if;
dbms_output.put_line(
    'The grade for the roll number : ' || to_char(i) || ' is : ' || grade
);
update studenttable
set lettergrade = grade
where rollno = i;
end loop;
end;
/
```

```
SQL> declare gp studenttable.gpa %type;
  2  grade varchar(2);
  3  begin for i in 1..5 loop
  4  select gpa into gp
  5  from studenttable
  6  where rollno = i;
  7  if gp > 0
  8  and gp ≤ 4 then grade := 'F';
  9  elsif gp > 4
 10  and gp ≤ 5 then grade := 'E';
 11  elsif gp > 5
 12  and gp ≤ 6 then grade := 'D';
 13  elsif gp > 6
 14  and gp ≤ 7 then grade := 'C';
 15  elsif gp > 7
 16  and gp ≤ 8 then grade := 'B';
 17  elsif gp > 8
 18  and gp ≤ 9 then grade := 'A';
 19  else grade := 'A+';
 20  end if;
 21  dbms_output.put_line(
 22     'The grade for the roll number : ' || to_char(i) || ' is : ' || grade
 23  );
 24  update studenttable
 25  set lettergrade = grade
 26  where rollno = i;
 27  end loop;
 28  end;
 29  /
The grade for the roll number : 1 is : D
The grade for the roll number : 2 is : C
The grade for the roll number : 3 is : F
The grade for the roll number : 4 is : B
The grade for the roll number : 5 is : A+

PL/SQL procedure successfully completed.

SQL> select * from studenttable;

    ROLLNO        GPA LE
---------- ---------- --
         1        5.8 D
         2        6.5 C
         3        3.4 F
         4        7.8 B
         5        9.5 A+
```

Question 6 : Write a PL/SQL block to find the student with max. GPA without using aggregate function.

```
declare mx studenttable.gpa %type;
cur studenttable.gpa %type;
mxr studenttable.rollno %type;
begin
select gpa into mx
from studenttable
where rollno = 1;
mxr := 1;
for i in 1..5 loop
select gpa into cur
from studenttable
where rollno = i;
if cur > mx then mx := cur;
mxr := i;
end if;
end loop;
dbms_output.put_line(
    'The maximum gpa is of roll no : ' || mxr || ' and the gpa is : ' ||
mx
);
end;
/
```

```
SQL> declare mx studenttable.gpa %type;
  2  cur studenttable.gpa %type;
  3  mxr studenttable.rollno %type;
  4  begin
  5  select gpa into mx
  6  from studenttable
  7  where rollno = 1;
  8  mxr := 1;
  9  for i in 1..5 loop
 10  select gpa into cur
 11  from studenttable
 12  where rollno = i;
 13  if cur > mx then mx := cur;
 14  mxr := i;
 15  end if;
 16  end loop;
 17  dbms_output.put_line(
 18     'The maximum gpa is of roll no : ' || mxr || ' and the gpa is : ' || mx
 19  );
 20  end;
 21  /
The maximum gpa is of roll no : 5 and the gpa is : 9.5

PL/SQL procedure successfully completed.
```

Question 7 : Implement lab exercise 4 using GOTO.

```
DECLARE gp studenttable.gpa %TYPE;
grade varchar(2);
BEGIN for i in 1..5 loop
select gpa into gp
from studenttable
where rollno = i;
if (
    gp >= 9
    and gp <= 10
) then goto ap;
elsif (
    gp >= 8
    and gp < 9
) then goto aa;
elsif (
    gp >= 7
    and gp < 8
) then goto bb;
elsif (
    gp >= 6
    and gp < 7
) then goto cc;
elsif (
    gp >= 5
    and gp < 6
) then goto dd;
elsif (
    gp >= 4
    and gp < 5
) then goto ee;
else goto ff;
end if;
<< ap >> grade := 'A+';
goto prnt;
<< aa >> grade := 'A';
goto prnt;
<< bb >> grade := 'B';
goto prnt;
<< cc >> grade := 'C';
goto prnt;
<< dd >> grade := 'D';
goto prnt;
<< ee >> grade := 'E';
goto prnt;
<< ff >> grade := 'F';
<< prnt >> dbms_output.put_line(
    'The roll number is :  ' || i || ' and the grade is :  ' || grade
```

```
);
end loop;
END;
/
```

```
 28     gp >= 4
 29     and gp < 5
 30  ) then goto ee;
 31  else goto ff;
 32  end if;
 33  << ap >> grade := 'A+';
 34  goto prnt;
 35  << aa >> grade := 'A';
 36  goto prnt;
 37  << bb >> grade := 'B';
 38  goto prnt;
 39  << cc >> grade := 'C';
 40  goto prnt;
 41  << dd >> grade := 'D';
 42  goto prnt;
 43  << ee >> grade := 'E';
 44  goto prnt;
 45  << ff >> grade := 'F';
 46  << prnt >> dbms_output.put_line(
 47     'The roll number is :  ' || i || ' and the grade is :  ' || grade
 48  );
 49  end loop;
 50  END;
 51  /
The roll number is :  1 and the grade is :  D
The roll number is :  2 and the grade is :  C
The roll number is :  3 and the grade is :  F
The roll number is :  4 and the grade is :  B
The roll number is :  5 and the grade is :  A+

PL/SQL procedure successfully completed.
```

Question 8 : Based on the University database schema, write a PL/SQL block to display the details of the Instructor whose name is supplied by the user. Use exceptions to show appropriate error message for the following cases:

a. Multiple instructors with the same name

b. No instructor for the given name

```
declare row instructor %rowtype;
names instructor.name %type;
begin names := '&n';
select * into row
from instructor
where name = names;
dbms_output.put_line(
    row.id || ' ' || row.name || ' ' || row.salary || ' ' || row.dept_nam
e
);
exception
when TOO_MANY_ROWS then dbms_output.put_line('Multiple values with same n
ame exist');
when NO_DATA_FOUND then dbms_output.put_line('Name not found');
end;
/
```

```
SQL> declare row instructor %rowtype;
  2  names instructor.name %type;
  3  begin names := '&n';
  4  select * into row
  5  from instructor
  6  where name = names;
  7  dbms_output.put_line(
  8     row.id || ' ' || row.name || ' ' || row.salary || ' ' || row.dept_
name
  9  );
 10  exception
 11  when TOO_MANY_ROWS then dbms_output.put_line('Multiple values with sa
me name exist');
 12  when NO_DATA_FOUND then dbms_output.put_line('Name not found');
 13  end;
 14  /
Enter value for n: Wieland
old   3: begin names := '&n';
new   3: begin names := 'Wieland';
19368 Wieland 124651.41 Pol. Sci.

PL/SQL procedure successfully completed.
```

```
SQL> declare row instructor %rowtype;
  2  names instructor.name %type;
  3  begin names := '&n';
  4  select * into row
  5  from instructor
  6  where name = names;
  7  dbms_output.put_line(
  8     row.id || ' ' || row.name || ' ' || row.salary || ' ' || row.dept_
name
  9  );
 10  exception
 11  when TOO_MANY_ROWS then dbms_output.put_line('Multiple values with sa
me name exist');
 12  when NO_DATA_FOUND then dbms_output.put_line('Name not found');
 13  end;
 14  /
Enter value for n: Singh
old   3: begin names := '&n';
new   3: begin names := 'Singh';
Name not found

PL/SQL procedure successfully completed.
```

Question 9 : Extend lab exercise5 to validate the GPA value used to find letter grade. If it is outside the range, 0 –10, display an error message, 'Out of Range' via an exception handler.

```
declare incorrectGpa exception;
gp studenttable.gpa %type;
grad varchar(2);
begin for i in 1..6 loop
select gpa into gp
from studenttable
where rollno = i;
if gp<0 or gp>10 then raise incorrectGpa;
elsif gp > 0
and gp <= 4 then grad := 'F';
elsif gp > 4
and gp <= 5 then grad := 'E';
elsif gp > 5
and gp <= 6 then grad := 'D';
elsif gp > 6
and gp <= 7 then grad := 'C';
elsif gp > 7
and gp <= 8 then grad := 'B';
elsif gp > 8
and gp <= 9 then grad := 'A';
else grad := 'A+';
end if;
dbms_output.put_line(
    'The grade for the roll number : ' || to_char(i) || ' is : ' || grad
```

```
);
update studenttable
set grade = grad
where rollno = i;
end loop;
exception when incorrectGpa then dbms_output.put_line('INcorrect gpa');
end;
```

```
The grade for the roll number : 1 is : D
The grade for the roll number : 2 is : C
The grade for the roll number : 3 is : F
The grade for the roll number : 4 is : B
The grade for the roll number : 5 is : A+
INcorrect gpa

PL/SQL procedure successfully completed.

SQL> select * from studenttable;

    ROLLNO        GPA GR
---------- ---------- --
         1        5.8 D
         2        6.5 C
         3        3.4 F
         4        7.8 B
         5        9.5 A+
         6        -.2

6 rows selected.
```