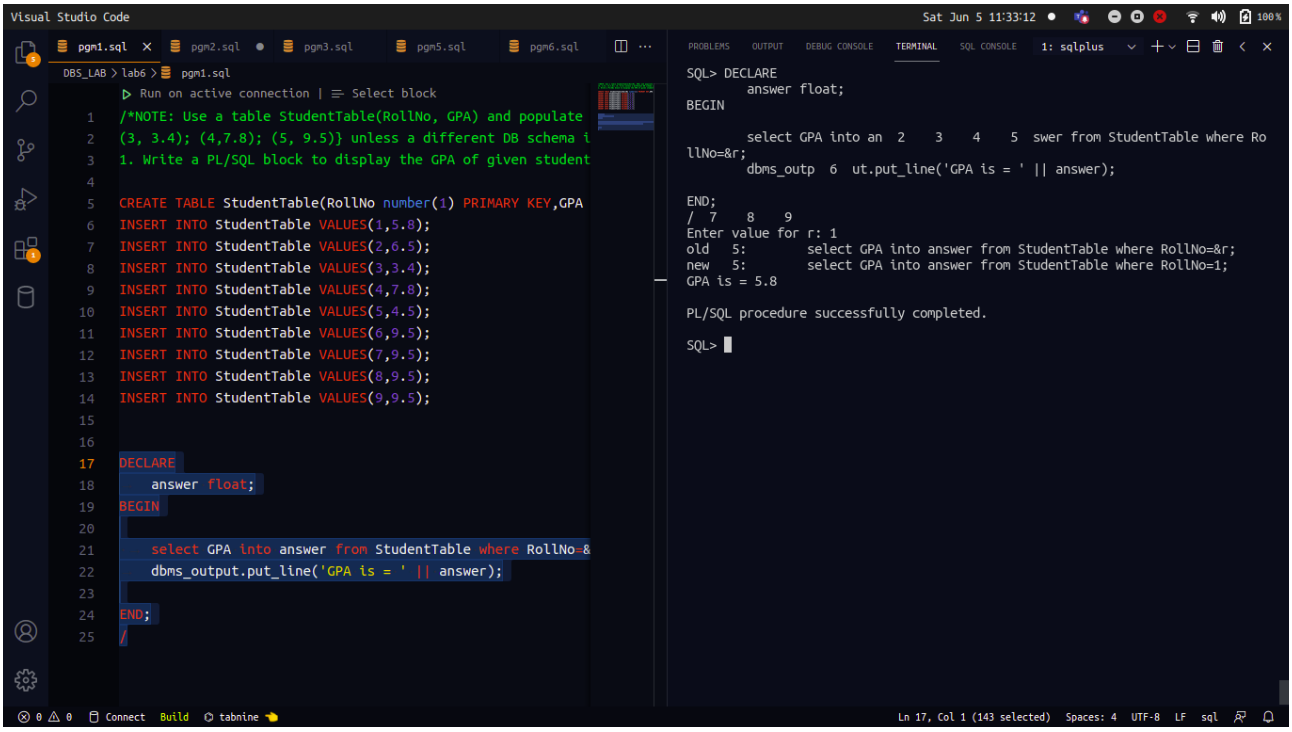
LAB 6

LAB EXERCISE

1)

```
CREATE TABLE StudentTable(RollNo number(1) PRIMARY KEY,GPA number(2,1));
INSERT INTO StudentTable VALUES(1,5.8);
INSERT INTO StudentTable VALUES(2,6.5);
INSERT INTO StudentTable VALUES(3,3.4);
INSERT INTO StudentTable VALUES(4,7.8);
INSERT INTO StudentTable VALUES(5,4.5);
INSERT INTO StudentTable VALUES(6,9.5);
INSERT INTO StudentTable VALUES(7,9.5);
INSERT INTO StudentTable VALUES(8,9.5);
INSERT INTO StudentTable VALUES(9,9.5);

DECLARE
answer float;
BEGIN

select GPA into answer from StudentTable where RollNo=&r;
dbms_output.put_line('GPA is = ' || answer);

END;
/
```

Output:


Visual Studio Code
Sat Jun 5 11:33:12
100%
pgm1.sql
pgm2.sql
pgm3.sql
pgm5.sql
pgm6.sql
DBS_LAB > lab6 > pgm1.sql
Run on active connection | Select block
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
SQL CONSOLE
1: sqlplus
0 0 Connect Build tabnine
Ln 17, Col 1 (143 selected) Spaces: 4 UTF-8 LF sql

```sql
/*NOTE: Use a table StudentTable(RollNo, GPA) and populate
(3, 3.4); (4,7.8); (5, 9.5)} unless a different DB schema i
1. Write a PL/SQL block to display the GPA of given student

CREATE TABLE StudentTable(RollNo number(1) PRIMARY KEY,GPA
INSERT INTO StudentTable VALUES(1,5.8);
INSERT INTO StudentTable VALUES(2,6.5);
INSERT INTO StudentTable VALUES(3,3.4);
INSERT INTO StudentTable VALUES(4,7.8);
INSERT INTO StudentTable VALUES(5,4.5);
INSERT INTO StudentTable VALUES(6,9.5);
INSERT INTO StudentTable VALUES(7,9.5);
INSERT INTO StudentTable VALUES(8,9.5);
INSERT INTO StudentTable VALUES(9,9.5);


DECLARE
    answer float;
BEGIN

    select GPA into answer from StudentTable where RollNo=&
    dbms_output.put_line('GPA is = ' || answer);

END;
/
```

```
SQL> DECLARE
        answer float;
BEGIN

        select GPA into an  2    3    4    5  swer from StudentTable where Ro
llNo=&r;
        dbms_outp  6  ut.put_line('GPA is = ' || answer);

END;
/  7    8    9
Enter value for r: 1
old   5:        select GPA into answer from StudentTable where RollNo=&r;
new   5:        select GPA into answer from StudentTable where RollNo=1;
GPA is = 5.8

PL/SQL procedure successfully completed.

SQL> █
```

2)

CREATE TABLE StudentTable1(RollNo number(2) PRIMARY KEY,gpa VARCHAR(10));

INSERT INTO StudentTable1 values(4,'F');

INSERT INTO StudentTable1 values(5,'E');

INSERT INTO StudentTable1 values(6,'D');

INSERT INTO StudentTable1 values(7,'C');

INSERT INTO StudentTable1 values(8,'B');

INSERT INTO StudentTable1 values(9,'A');

INSERT INTO StudentTable1 values(10,'A+');

DECLARE

result number(2);

grade VARCHAR(10);

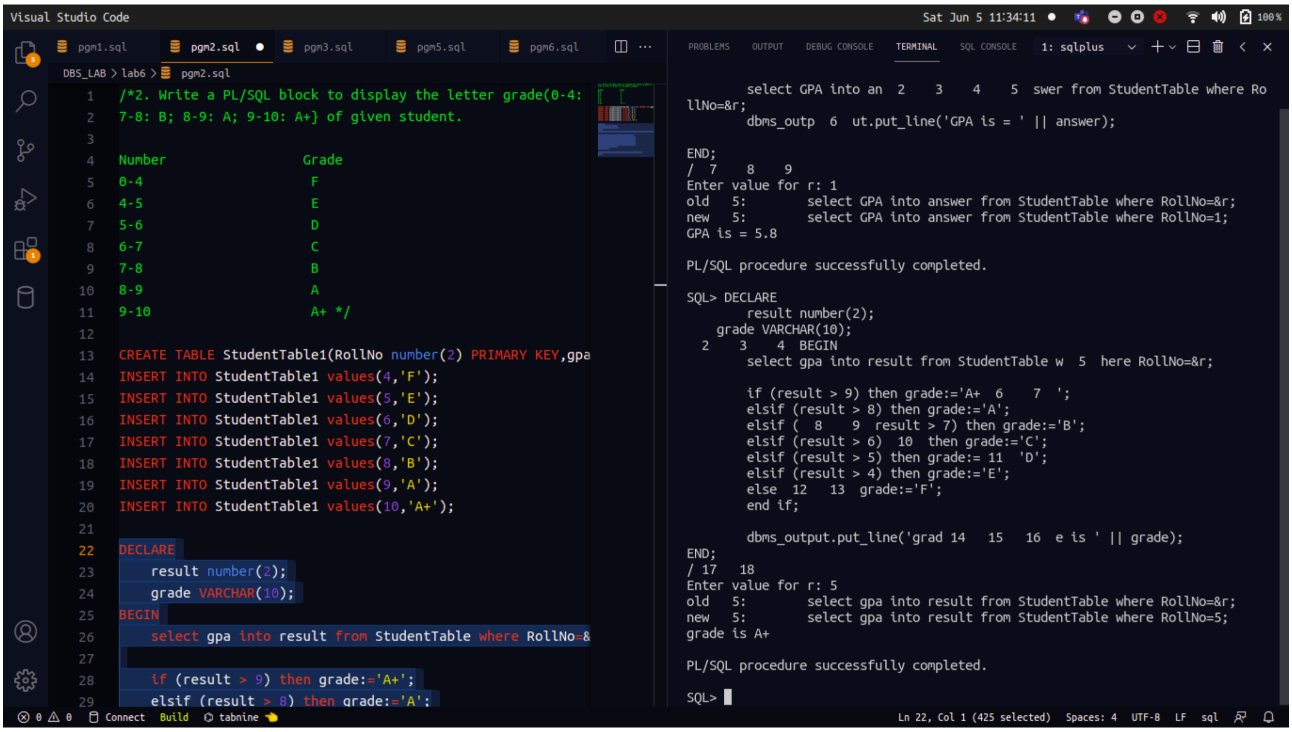
BEGIN

select gpa into result from StudentTable where RollNo=&r;

if (result > 9) then grade:='A+';

```
elsif (result > 8) then grade:='A';
elsif (result > 7) then grade:='B';
elsif (result > 6) then grade:='C';
elsif (result > 5) then grade:='D';
elsif (result > 4) then grade:='E';
else grade:='F';
end if;

dbms_output.put_line('grade is ' || grade);
END;
/
```

Output:



```
/*2. Write a PL/SQL block to display the letter grade(0-4:
7-8: B; 8-9: A; 9-10: A+} of given student.

Number                     Grade
0-4                         F
4-5                         E
5-6                         D
6-7                         C
7-8                         B
8-9                         A
9-10                        A+ */

CREATE TABLE StudentTable1(RollNo number(2) PRIMARY KEY,gpa
INSERT INTO StudentTable1 values(4,'F');
INSERT INTO StudentTable1 values(5,'E');
INSERT INTO StudentTable1 values(6,'D');
INSERT INTO StudentTable1 values(7,'C');
INSERT INTO StudentTable1 values(8,'B');
INSERT INTO StudentTable1 values(9,'A');
INSERT INTO StudentTable1 values(10,'A+');

DECLARE
    result number(2);
    grade VARCHAR(10);
BEGIN
    select gpa into result from StudentTable where RollNo=&

    if (result > 9) then grade:='A+';
    elsif (result > 8) then grade:='A';
```

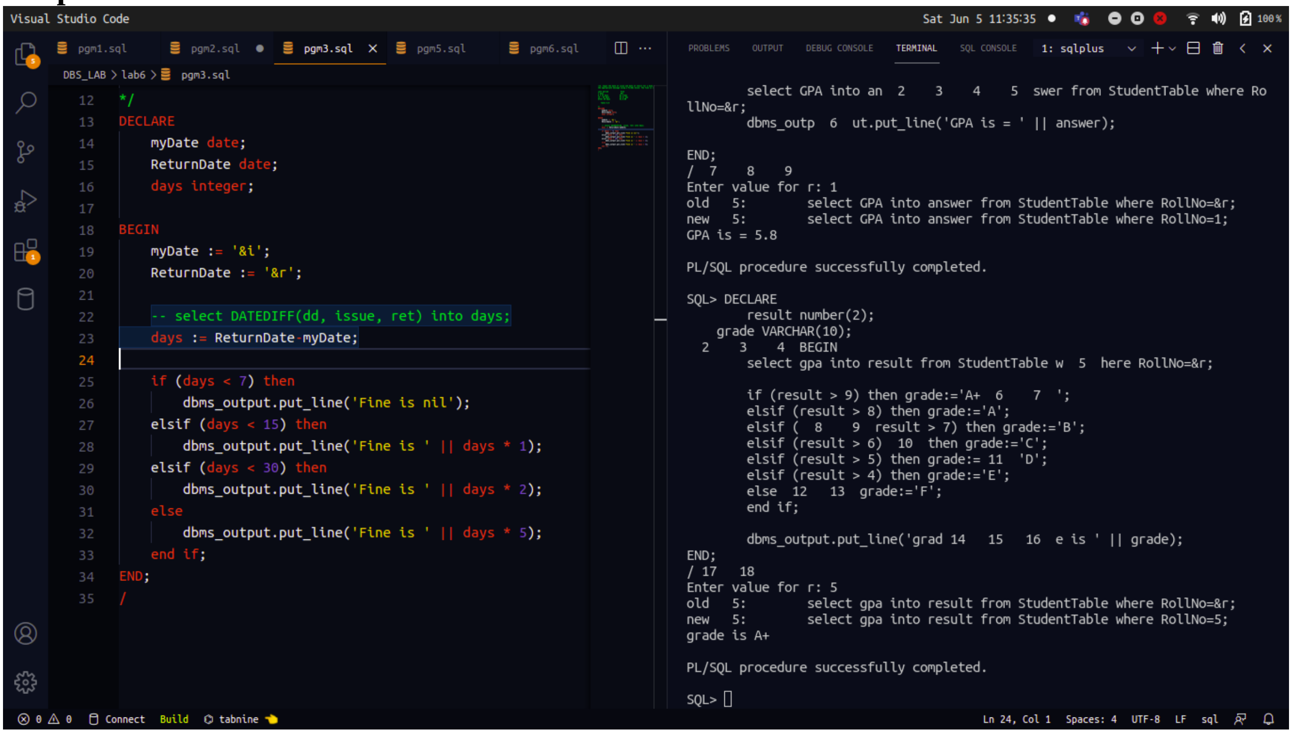
```
        select GPA into an  2    3    4    5  swer from StudentTable where Ro
llNo=&r;
        dbms_outp  6  ut.put_line('GPA is = ' || answer);

END;
/  7    8    9
Enter value for r: 1
old   5:        select GPA into answer from StudentTable where RollNo=&r;
new   5:        select GPA into answer from StudentTable where RollNo=1;
GPA is = 5.8

PL/SQL procedure successfully completed.

SQL> DECLARE
        result number(2);
    grade VARCHAR(10);
  2    3    4  BEGIN
        select gpa into result from StudentTable w  5  here RollNo=&r;

        if (result > 9) then grade:='A+  6    7  ';
        elsif (result > 8) then grade:='A';
        elsif (  8    9  result > 7) then grade:='B';
        elsif (result > 6)  10  then grade:='C';
        elsif (result > 5) then grade:= 11  'D';
        elsif (result > 4) then grade:='E';
        else  12   13  grade:='F';
        end if;

        dbms_output.put_line('grad 14   15   16  e is ' || grade);
END;
/ 17   18
Enter value for r: 5
old   5:        select gpa into result from StudentTable where RollNo=&r;
new   5:        select gpa into result from StudentTable where RollNo=5;
grade is A+

PL/SQL procedure successfully completed.

SQL> 
```

3)

```
DECLARE
myDate date;
ReturnDate date;
days integer;
```

```
BEGIN
myDate := '&i';
ReturnDate := '&r';

-- select DATEDIFF(dd, issue, ret) into days;
days := ReturnDate-myDate;

if (days < 7) then
dbms_output.put_line('Fine is nil');
elsif (days < 15) then
dbms_output.put_line('Fine is ' || days * 1);
elsif (days < 30) then
dbms_output.put_line('Fine is ' || days * 2);
else
dbms_output.put_line('Fine is ' || days * 5);
end if;
END;
/
```

Output:



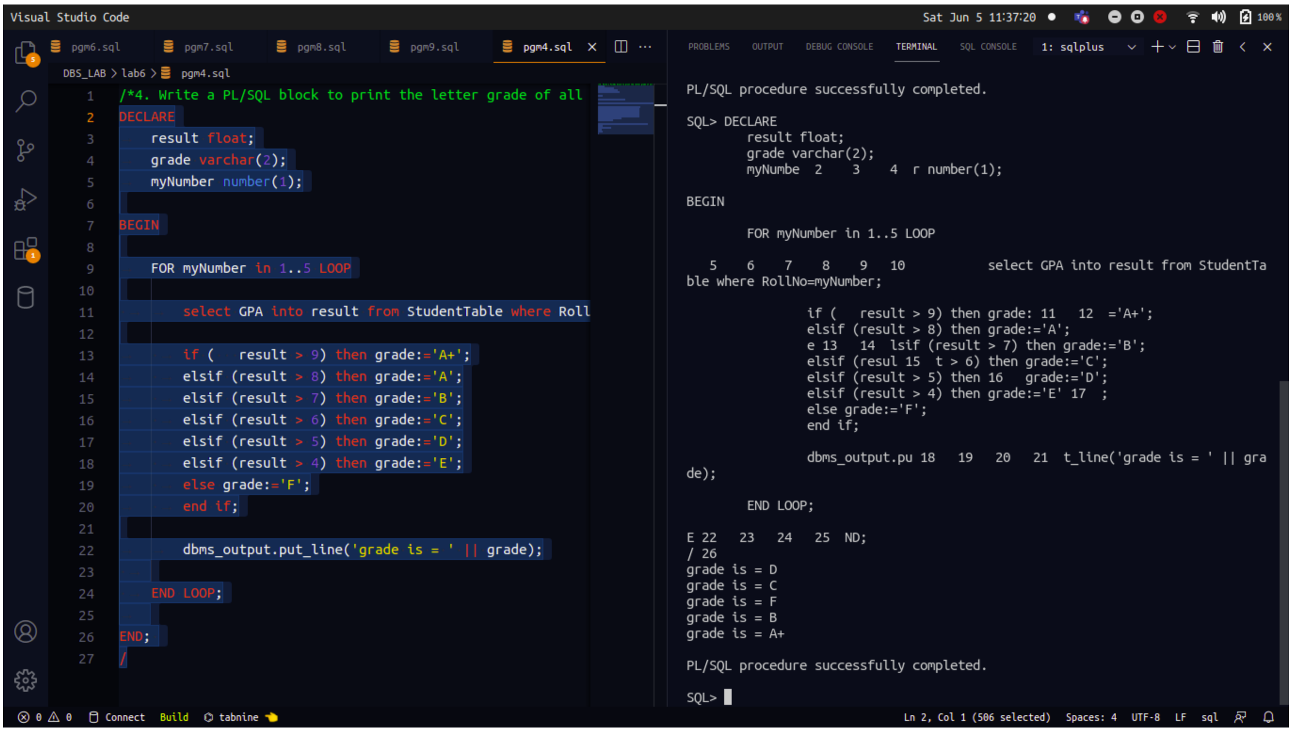
```
*/
DECLARE
    myDate date;
    ReturnDate date;
    days integer;

BEGIN
    myDate := '&i';
    ReturnDate := '&r';

    -- select DATEDIFF(dd, issue, ret) into days;
    days := ReturnDate-myDate;

    if (days < 7) then
        dbms_output.put_line('Fine is nil');
    elsif (days < 15) then
        dbms_output.put_line('Fine is ' || days * 1);
    elsif (days < 30) then
        dbms_output.put_line('Fine is ' || days * 2);
    else
        dbms_output.put_line('Fine is ' || days * 5);
    end if;
END;
/
```

```
        select GPA into an  2    3    4    5  swer from StudentTable where Ro
llNo=&r;
        dbms_outp  6  ut.put_line('GPA is = ' || answer);

END;
/  7    8    9
Enter value for r: 1
old   5:        select GPA into answer from StudentTable where RollNo=&r;
new   5:        select GPA into answer from StudentTable where RollNo=1;
GPA is = 5.8

PL/SQL procedure successfully completed.

SQL> DECLARE
        result number(2);
    grade VARCHAR(10);
  2    3    4  BEGIN
        select gpa into result from StudentTable w  5  here RollNo=&r;

        if (result > 9) then grade:='A+  6    7  ';
        elsif (result > 8) then grade:='A';
        elsif (  8    9  result > 7) then grade:='B';
        elsif (result > 6)  10  then grade:='C';
        elsif (result > 5) then grade:= 11  'D';
        elsif (result > 4) then grade:='E';
        else  12   13  grade:='F';
        end if;

        dbms_output.put_line('grad 14   15   16  e is ' || grade);
END;
/ 17   18
Enter value for r: 5
old   5:        select gpa into result from StudentTable where RollNo=&r;
new   5:        select gpa into result from StudentTable where RollNo=5;
grade is A+

PL/SQL procedure successfully completed.

SQL> 
```

4)

```
DECLARE
result float;
grade varchar(2);
myNumber number(1);

BEGIN

FOR myNumber in 1..5 LOOP

select GPA into result from StudentTable where RollNo=myNumber;

if ( result > 9) then grade:='A+';
elsif (result > 8) then grade:='A';
elsif (result > 7) then grade:='B';
elsif (result > 6) then grade:='C';
elsif (result > 5) then grade:='D';
elsif (result > 4) then grade:='E';
else grade:='F';
end if;

dbms_output.put_line('grade is = ' || grade);
END LOOP;
END;
/
```

Output:



```sql
/*4. Write a PL/SQL block to print the letter grade of all
DECLARE
    result float;
    grade varchar(2);
    myNumber number(1);

BEGIN

    FOR myNumber in 1..5 LOOP

        select GPA into result from StudentTable where Roll

        if (   result > 9) then grade:='A+';
        elsif (result > 8) then grade:='A';
        elsif (result > 7) then grade:='B';
        elsif (result > 6) then grade:='C';
        elsif (result > 5) then grade:='D';
        elsif (result > 4) then grade:='E';
        else grade:='F';
        end if;

        dbms_output.put_line('grade is = ' || grade);

    END LOOP;

END;
/
```

```
PL/SQL procedure successfully completed.

SQL> DECLARE
        result float;
        grade varchar(2);
        myNumbe  2    3    4  r number(1);

BEGIN

        FOR myNumber in 1..5 LOOP

   5    6    7    8    9   10           select GPA into result from StudentTa
ble where RollNo=myNumber;

                if (   result > 9) then grade: 11   12  ='A+';
                elsif (result > 8) then grade:='A';
                e 13   14  lsif (result > 7) then grade:='B';
                elsif (resul 15  t > 6) then grade:='C';
                elsif (result > 5) then 16   grade:='D';
                elsif (result > 4) then grade:='E' 17  ;
                else grade:='F';
                end if;

                dbms_output.pu 18   19   20   21  t_line('grade is = ' || gra
de);

        END LOOP;

E 22   23   24   25  ND;
/ 26
grade is = D
grade is = C
grade is = F
grade is = B
grade is = A+

PL/SQL procedure successfully completed.

SQL> █
```

5)

alter table StudentTable

add LetterGrade varchar2(2);

DECLARE

result float;

grade varchar(2);

myNumber number(1);

BEGIN

FOR myNumber in 1..5 LOOP

select GPA into result from StudentTable where RollNo=myNumber;

if (result > 9) then grade:='A+';

elsif (result > 8) then grade:='A';

elsif (result > 7) then grade:='B';

elsif (result > 6) then grade:='C';

elsif (result > 5) then grade:='D';

elsif (result > 4) then grade:='E';

else grade:='F';

end if;

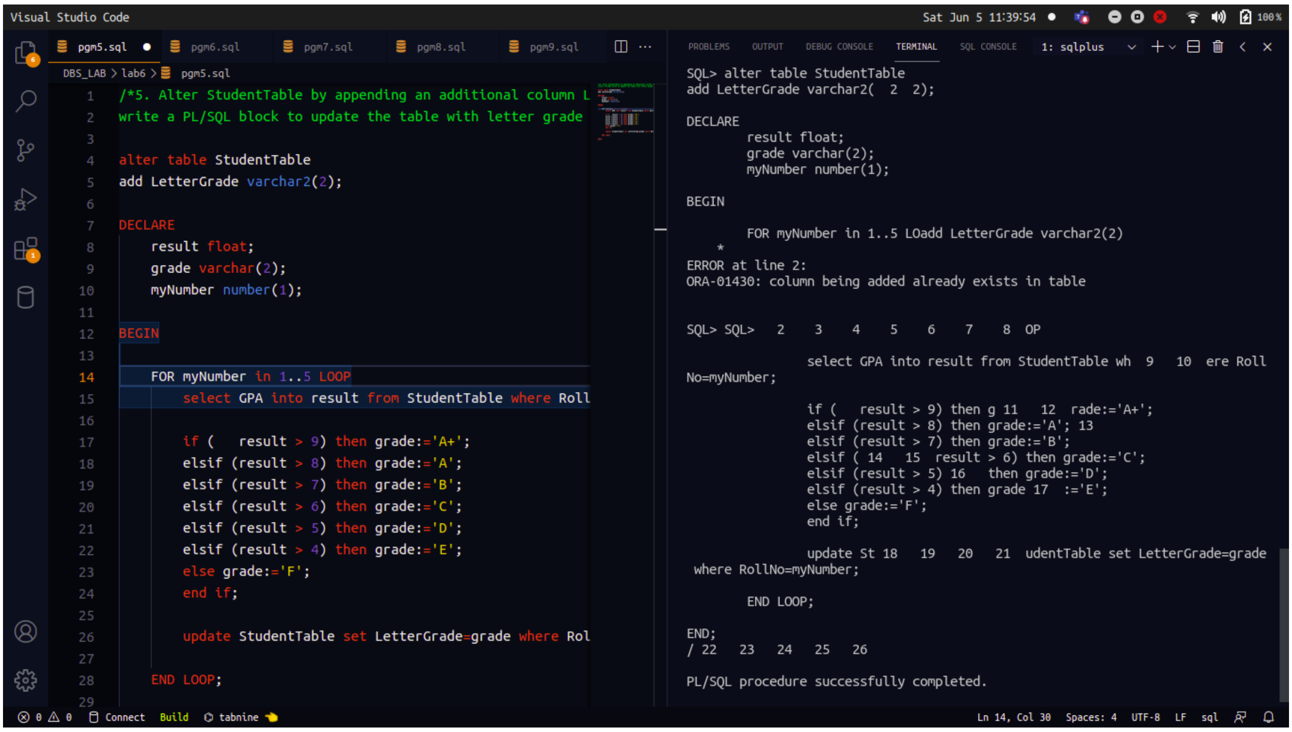
update StudentTable set LetterGrade=grade where RollNo=myNumber;

END LOOP;

END;

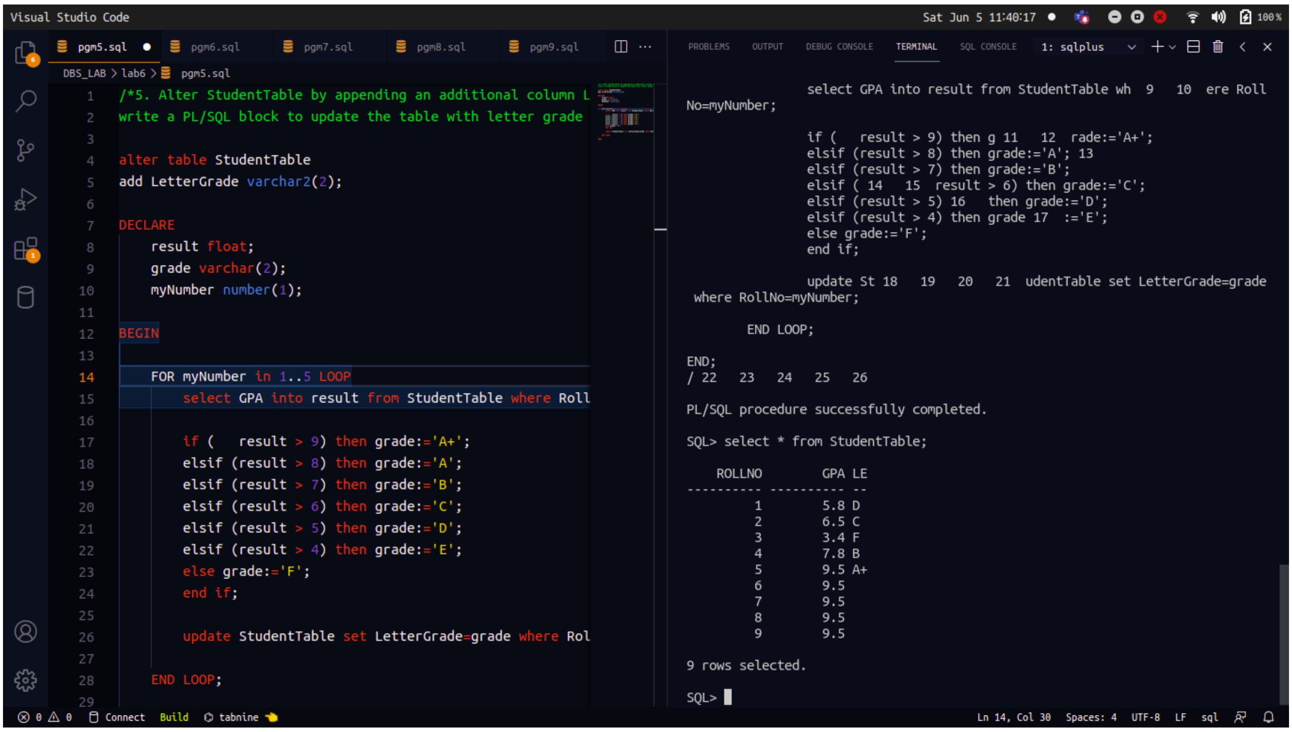
/

Output:



```
/*5. Alter StudentTable by appending an additional column L
write a PL/SQL block to update the table with letter grade

alter table StudentTable
add LetterGrade varchar2(2);

DECLARE
    result float;
    grade varchar(2);
    myNumber number(1);

BEGIN

    FOR myNumber in 1..5 LOOP
        select GPA into result from StudentTable where Roll

        if (   result > 9) then grade:='A+';
        elsif (result > 8) then grade:='A';
        elsif (result > 7) then grade:='B';
        elsif (result > 6) then grade:='C';
        elsif (result > 5) then grade:='D';
        elsif (result > 4) then grade:='E';
        else grade:='F';
        end if;

        update StudentTable set LetterGrade=grade where Rol

    END LOOP;
```

```
SQL> alter table StudentTable
add LetterGrade varchar2(  2  2);

DECLARE
        result float;
        grade varchar(2);
        myNumber number(1);

BEGIN

        FOR myNumber in 1..5 LOadd LetterGrade varchar2(2)
    *
ERROR at line 2:
ORA-01430: column being added already exists in table


SQL> SQL>   2    3    4    5    6    7    8  OP

                select GPA into result from StudentTable wh  9   10  ere Roll
No=myNumber;

                if (   result > 9) then g 11   12  rade:='A+';
                elsif (result > 8) then grade:='A'; 13
                elsif (result > 7) then grade:='B';
                elsif ( 14   15  result > 6) then grade:='C';
                elsif (result > 5) 16   then grade:='D';
                elsif (result > 4) then grade 17  :='E';
                else grade:='F';
                end if;

                update St 18   19   20   21  udentTable set LetterGrade=grade
 where RollNo=myNumber;

        END LOOP;

END;
/ 22   23   24   25   26

PL/SQL procedure successfully completed.
```



```sql
/*5. Alter StudentTable by appending an additional column L
write a PL/SQL block to update the table with letter grade

alter table StudentTable
add LetterGrade varchar2(2);

DECLARE
    result float;
    grade varchar(2);
    myNumber number(1);

BEGIN

    FOR myNumber in 1..5 LOOP
        select GPA into result from StudentTable where Roll

        if (   result > 9) then grade:='A+';
        elsif (result > 8) then grade:='A';
        elsif (result > 7) then grade:='B';
        elsif (result > 6) then grade:='C';
        elsif (result > 5) then grade:='D';
        elsif (result > 4) then grade:='E';
        else grade:='F';
        end if;

        update StudentTable set LetterGrade=grade where Rol

    END LOOP;

```

```
                select GPA into result from StudentTable wh  9   10  ere Roll
No=myNumber;

                if (   result > 9) then g 11   12  rade:='A+';
                elsif (result > 8) then grade:='A'; 13
                elsif (result > 7) then grade:='B';
                elsif ( 14   15  result > 6) then grade:='C';
                elsif (result > 5) 16   then grade:='D';
                elsif (result > 4) then grade 17  :='E';
                else grade:='F';
                end if;

                update St 18   19   20   21  udentTable set LetterGrade=grade
 where RollNo=myNumber;

        END LOOP;

END;
/ 22   23   24   25   26

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+
         6        9.5
         7        9.5
         8        9.5
         9        9.5

9 rows selected.

SQL> 
```

6)

DECLARE

result float;

helloMaximum float;

studentNumber number(1);

myNumber number(1);

BEGIN

select GPA into helloMaximum from StudentTable where RollNo=1;

studentNumber:=1;

FOR myNumber in 2..5 LOOP

select GPA into result from StudentTable where RollNo=myNumber;

if (result > helloMaximum) then studentNumber:=myNumber;

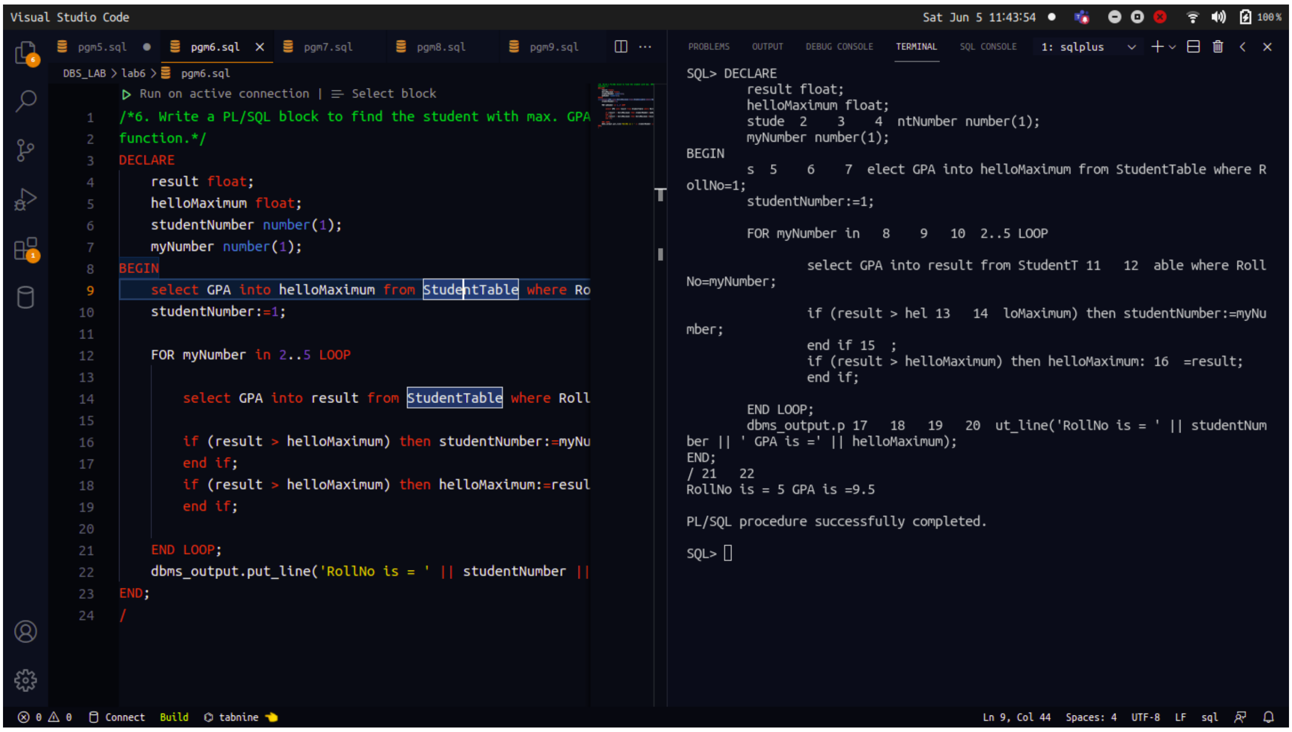
end if;

if (result > helloMaximum) then helloMaximum:=result;

end if;

```
END LOOP;
dbms_output.put_line('RollNo is = ' || studentNumber || ' GPA is =' || helloMaximum);
END;
/
```

Output:



```
/*6. Write a PL/SQL block to find the student with max. GPA
function.*/
DECLARE
    result float;
    helloMaximum float;
    studentNumber number(1);
    myNumber number(1);
BEGIN
    select GPA into helloMaximum from StudentTable where Ro
    studentNumber:=1;

    FOR myNumber in 2..5 LOOP

        select GPA into result from StudentTable where Roll

        if (result > helloMaximum) then studentNumber:=myNu
        end if;
        if (result > helloMaximum) then helloMaximum:=resul
        end if;

    END LOOP;
    dbms_output.put_line('RollNo is = ' || studentNumber ||
END;
/
```

```
SQL> DECLARE
        result float;
        helloMaximum float;
        stude  2    3    4  ntNumber number(1);
        myNumber number(1);
BEGIN
        s  5    6    7  elect GPA into helloMaximum from StudentTable where R
ollNo=1;
        studentNumber:=1;

        FOR myNumber in   8    9   10  2..5 LOOP

                select GPA into result from StudentT 11   12  able where Roll
No=myNumber;

                if (result > hel 13   14  loMaximum) then studentNumber:=myNu
mber;
                end if 15  ;
                if (result > helloMaximum) then helloMaximum: 16  =result;
                end if;

        END LOOP;
        dbms_output.p 17   18   19   20  ut_line('RollNo is = ' || studentNum
ber || ' GPA is =' || helloMaximum);
END;
/ 21   22
RollNo is = 5 GPA is =9.5

PL/SQL procedure successfully completed.

SQL> 
```

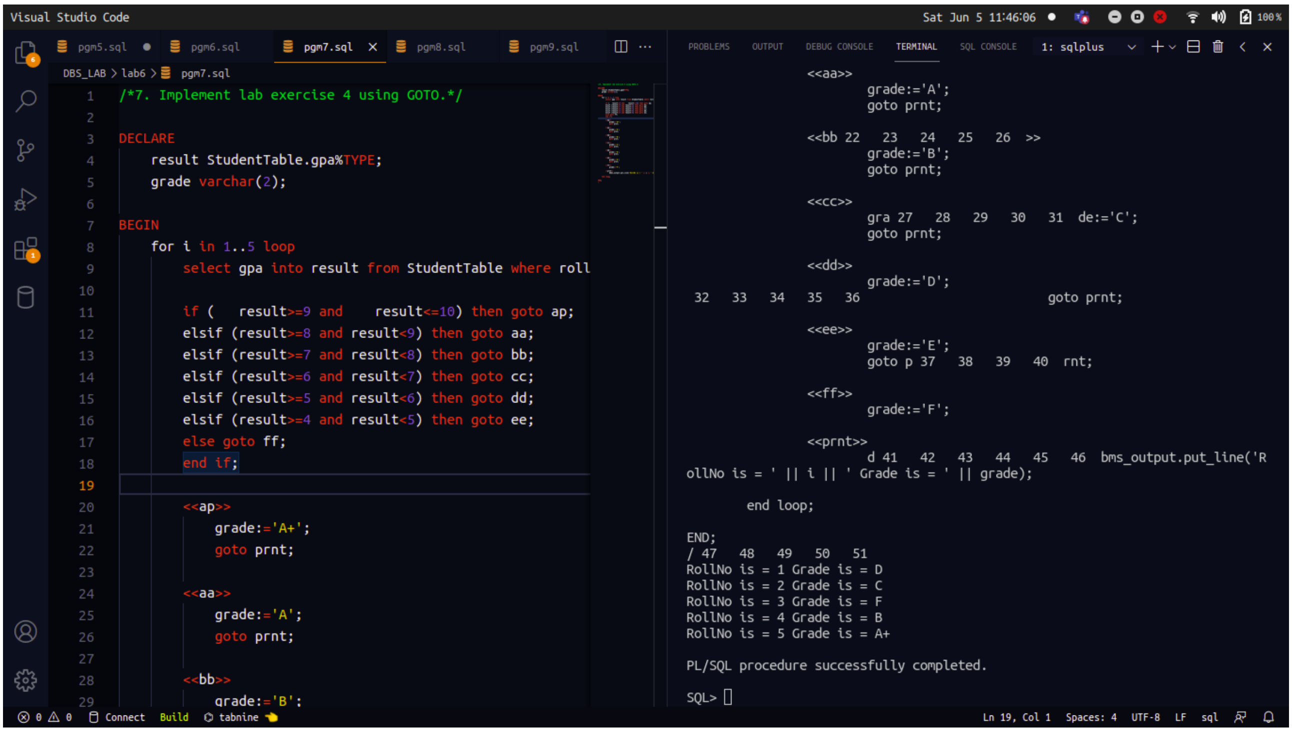
7)

```
DECLARE
result StudentTable.gpa%TYPE;
grade varchar(2);

BEGIN
for i in 1..5 loop
select gpa into result from StudentTable where rollno = i;

if ( result>=9 and result<=10) then goto ap;
elsif (result>=8 and result<9) then goto aa;
elsif (result>=7 and result<8) then goto bb;
elsif (result>=6 and result<7) then goto cc;
elsif (result>=5 and result<6) then goto dd;
elsif (result>=4 and result<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('RollNo is = ' || i || ' Grade is = ' || grade);

end loop;
```

END;

/

Output:



```
/*7. Implement lab exercise 4 using GOTO.*/

DECLARE
    result StudentTable.gpa%TYPE;
    grade varchar(2);

BEGIN
    for i in 1..5 loop
        select gpa into result from StudentTable where roll

        if (   result>=9 and    result<=10) then goto ap;
        elsif (result>=8 and result<9) then goto aa;
        elsif (result>=7 and result<8) then goto bb;
        elsif (result>=6 and result<7) then goto cc;
        elsif (result>=5 and result<6) then goto dd;
        elsif (result>=4 and result<5) then goto ee;
        else goto ff;
        end if;

        <<ap>>
            grade:='A+';
            goto prnt;

        <<aa>>
            grade:='A';
            goto prnt;

        <<bb>>
            grade:='B';
```

```
                <<aa>>
                        grade:='A';
                        goto prnt;

                <<bb 22   23   24   25   26  >>
                        grade:='B';
                        goto prnt;

                <<cc>>
                        gra 27   28   29   30   31  de:='C';
                        goto prnt;

                <<dd>>
                        grade:='D';
 32   33   34   35   36                       goto prnt;

                <<ee>>
                        grade:='E';
                        goto p 37   38   39   40  rnt;

                <<ff>>
                        grade:='F';

                <<prnt>>
                        d 41   42   43   44   45   46  bms_output.put_line('R
ollNo is = ' || i || ' Grade is = ' || grade);

        end loop;

END;
/ 47   48   49   50   51
RollNo is = 1 Grade is = D
RollNo is = 2 Grade is = C
RollNo is = 3 Grade is = F
RollNo is = 4 Grade is = B
RollNo is = 5 Grade is = A+

PL/SQL procedure successfully completed.

SQL> 
```

8)

DECLARE

namestudent instructor.name%TYPE;

val instructor%ROWTYPE;

BEGIN

namestudent := '&myNumber';

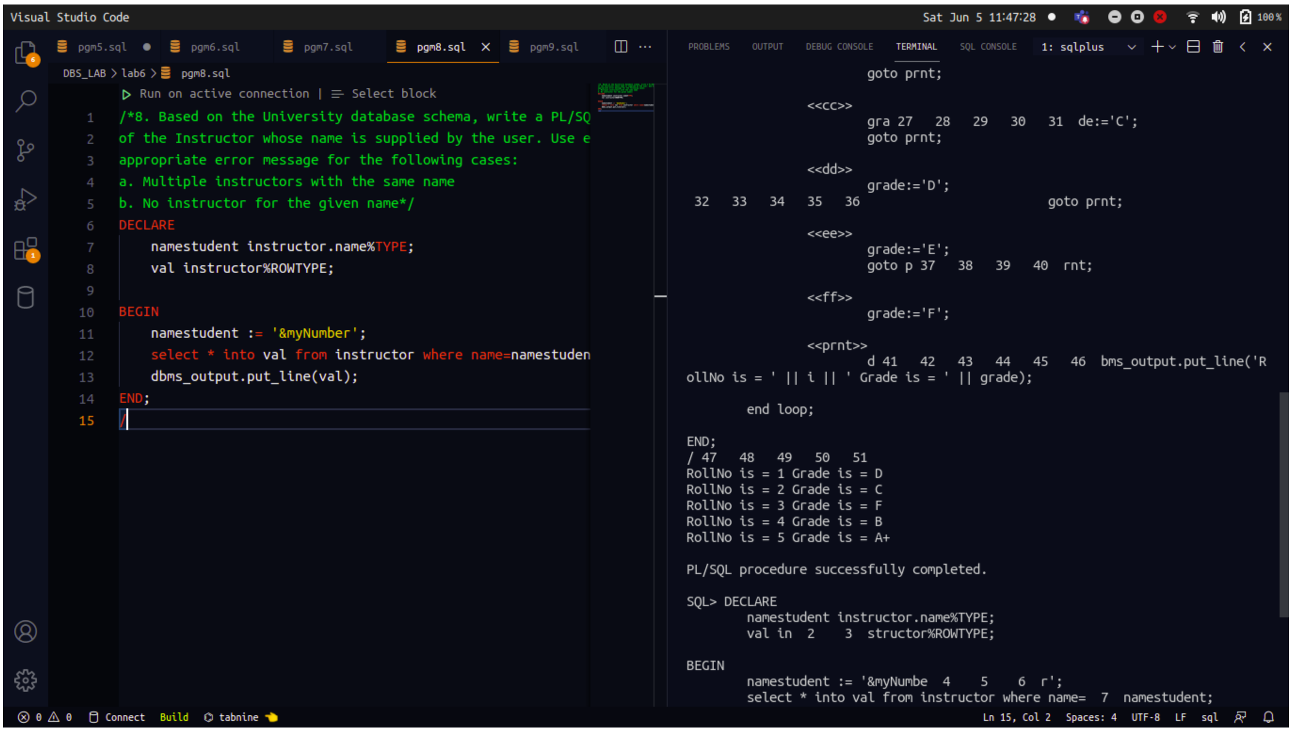
select * into val from instructor where name=namestudent;

dbms_output.put_line(val);

END;

/

Output:



```
/*8. Based on the University database schema, write a PL/SQ
of the Instructor whose name is supplied by the user. Use e
appropriate error message for the following cases:
a. Multiple instructors with the same name
b. No instructor for the given name*/
DECLARE
    namestudent instructor.name%TYPE;
    val instructor%ROWTYPE;

BEGIN
    namestudent := '&myNumber';
    select * into val from instructor where name=namestuden
    dbms_output.put_line(val);
END;
/
```

```
                        goto prnt;

                <<cc>>
                        gra 27   28   29   30   31  de:='C';
                        goto prnt;

                <<dd>>
                        grade:='D';
 32   33   34   35   36                       goto prnt;

                <<ee>>
                        grade:='E';
                        goto p 37   38   39   40  rnt;

                <<ff>>
                        grade:='F';

                <<prnt>>
                        d 41   42   43   44   45   46  bms_output.put_line('R
ollNo is = ' || i || ' Grade is = ' || grade);

        end loop;

END;
/ 47   48   49   50   51
RollNo is = 1 Grade is = D
RollNo is = 2 Grade is = C
RollNo is = 3 Grade is = F
RollNo is = 4 Grade is = B
RollNo is = 5 Grade is = A+

PL/SQL procedure successfully completed.

SQL> DECLARE
        namestudent instructor.name%TYPE;
        val in  2    3  structor%ROWTYPE;

BEGIN
        namestudent := '&myNumbe  4    5    6  r';
        select * into val from instructor where name=  7  namestudent;
```

9)

DECLARE

OutOfRangeException Exception;

result StudentTable.gpa%TYPE;

grade StudentTable.lettergrade%TYPE;

BEGIN

for i in 1..5 loop

select gpa into result from StudentTable where RollNo = i;

if (result>=9 and result<=10) then grade:='A+';

elsif (result>=8 and result<9) then grade:='A';

elsif (result>=7 and result<8) then grade:='B';

elsif (result>=6 and result<7) then grade:='C';

elsif (result>=5 and result<6) then grade:='D';

elsif (result>=4 and result<5) then grade:='E';

elsif (result>=0 and result<4) then grade:='F';

else RAISE OutOfRangeException;

end if;

update StudentTable set lettergrade=grade where RollNo=i;

end loop;

EXCEPTION

when OutOfRangeException then

dbms_output.put_line('GPA out of range');

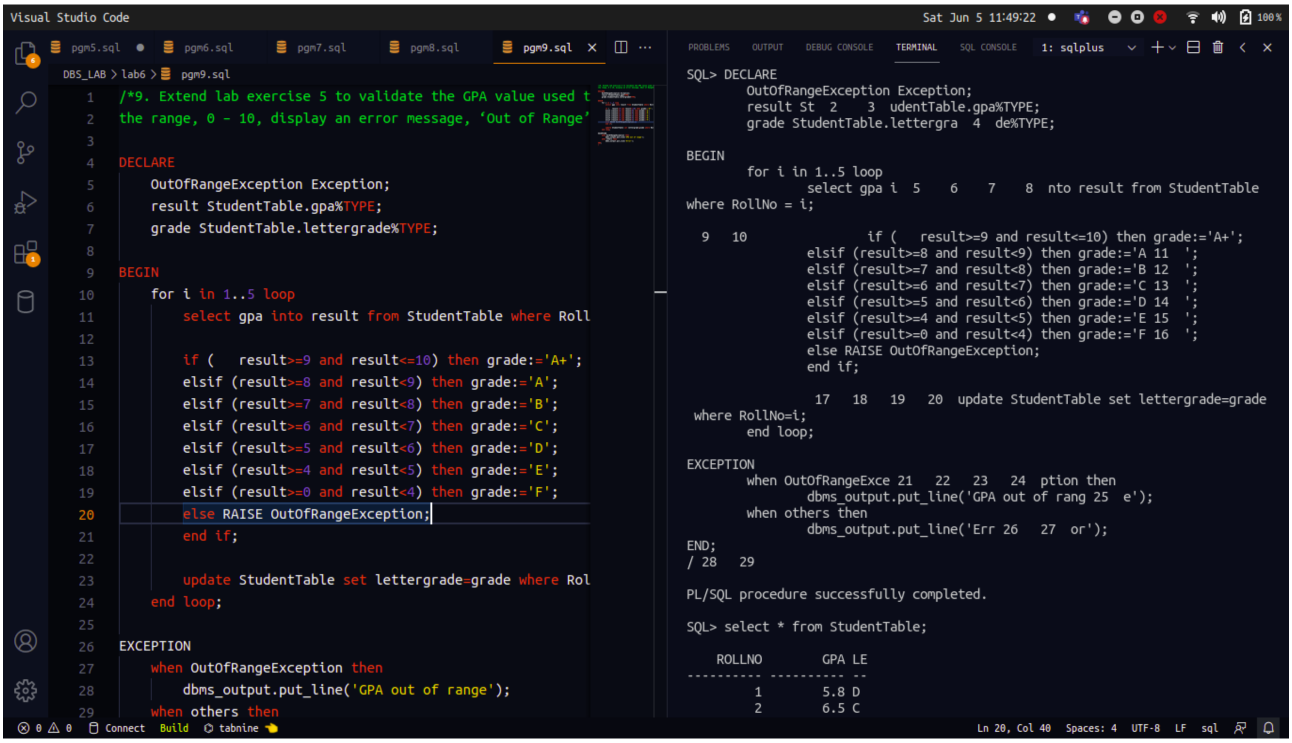
when others then

dbms_output.put_line('Error');

END;

/

Output:



```
/*9. Extend lab exercise 5 to validate the GPA value used t
the range, 0 - 10, display an error message, ‘Out of Range’

DECLARE
    OutOfRangeException Exception;
    result StudentTable.gpa%TYPE;
    grade StudentTable.lettergrade%TYPE;

BEGIN
    for i in 1..5 loop
        select gpa into result from StudentTable where Roll

        if (   result>=9 and result<=10) then grade:='A+';
        elsif (result>=8 and result<9) then grade:='A';
        elsif (result>=7 and result<8) then grade:='B';
        elsif (result>=6 and result<7) then grade:='C';
        elsif (result>=5 and result<6) then grade:='D';
        elsif (result>=4 and result<5) then grade:='E';
        elsif (result>=0 and result<4) then grade:='F';
        else RAISE OutOfRangeException;
        end if;

        update StudentTable set lettergrade=grade where Rol
    end loop;

EXCEPTION
    when OutOfRangeException then
        dbms_output.put_line('GPA out of range');
    when others then
```

```
SQL> DECLARE
        OutOfRangeException Exception;
        result St  2    3  udentTable.gpa%TYPE;
        grade StudentTable.lettergra  4  de%TYPE;

BEGIN
        for i in 1..5 loop
                select gpa i  5    6    7    8  nto result from StudentTable
where RollNo = i;

  9   10                if (   result>=9 and result<=10) then grade:='A+';
                elsif (result>=8 and result<9) then grade:='A 11  ';
                elsif (result>=7 and result<8) then grade:='B 12  ';
                elsif (result>=6 and result<7) then grade:='C 13  ';
                elsif (result>=5 and result<6) then grade:='D 14  ';
                elsif (result>=4 and result<5) then grade:='E 15  ';
                elsif (result>=0 and result<4) then grade:='F 16  ';
                else RAISE OutOfRangeException;
                end if;

                 17   18   19   20  update StudentTable set lettergrade=grade
 where RollNo=i;
        end loop;

EXCEPTION
        when OutOfRangeExce 21   22   23   24  ption then
                dbms_output.put_line('GPA out of rang 25  e');
        when others then
                dbms_output.put_line('Err 26   27  or');
END;
/ 28   29

PL/SQL procedure successfully completed.

SQL> select * from StudentTable;

    ROLLNO        GPA LE
---------- ---------- --
         1        5.8 D
         2        6.5 C
```