ASSIGNMENT - 01

CSE 464(ADVANCED DATABASE)

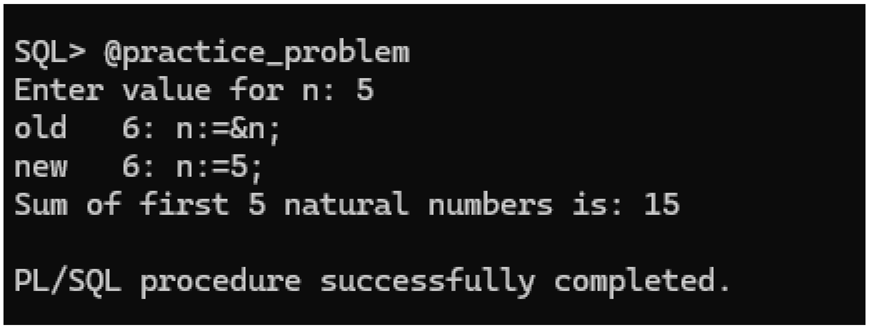
FROM

JANNATUL FERDAUS OISHI
ID-2022-3-60-216

TO
Antu Chowdhury
Lecturer
Department of Computer Science & Engineering

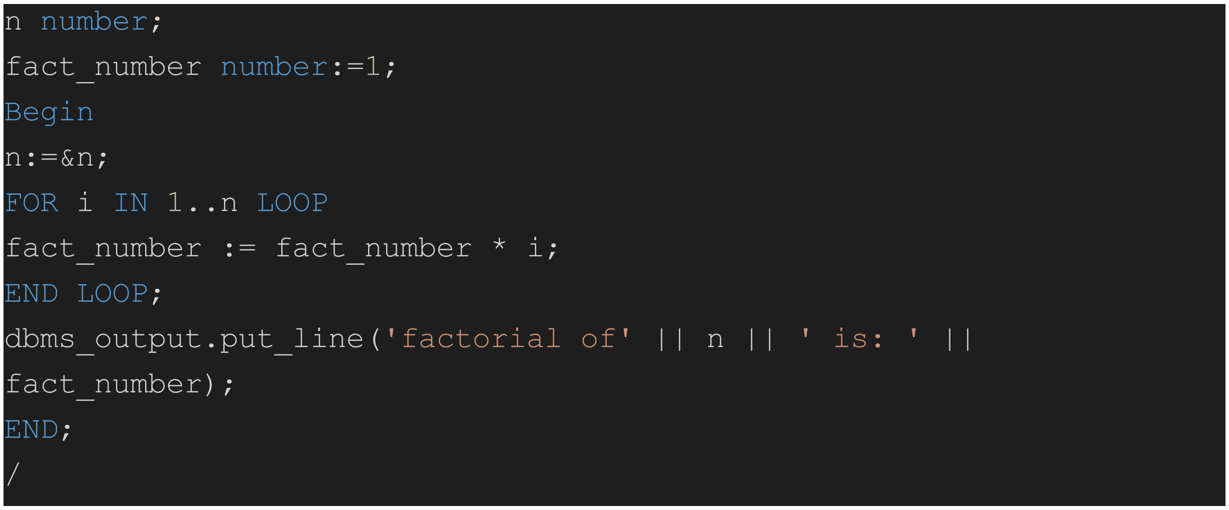
DATE:28 JANUARY, 2026

```
--1.Sum of First N Natural Numbers

Declare
n number;
i NUMBER := 1;
sum_number NUMBER := 0;
Begin
n:=&n;
LOOP
Exit when i >n ;
sum_number := sum_number + i;
i:=i+1;
END LOOP;
DBMS_OUTPUT.PUT_LINE('Sum of first ' || n || ' natural numbers is: ' || sum_number);
END;
/
```

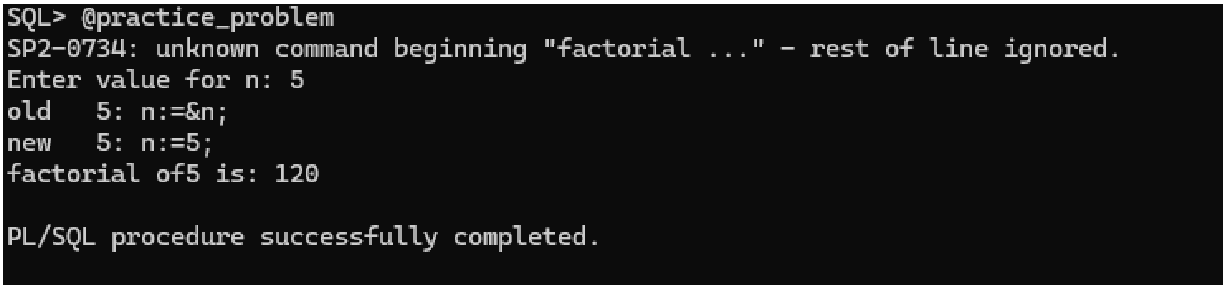


```
SQL> @practice_problem
Enter value for n: 5
old   6: n:=&n;
new   6: n:=5;
Sum of first 5 natural numbers is: 15

PL/SQL procedure successfully completed.
```

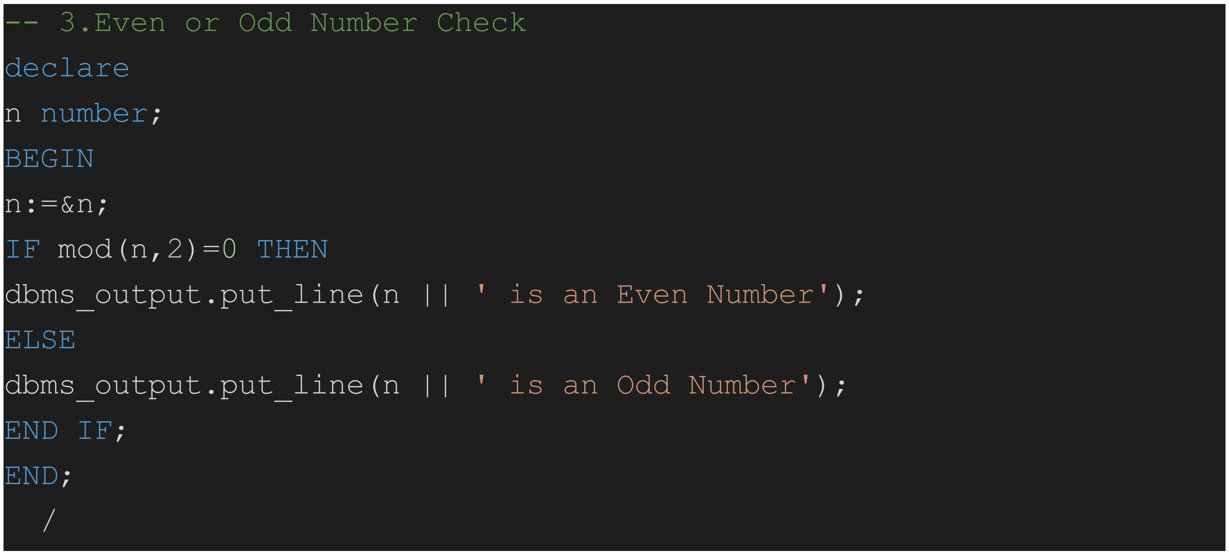
```
-- 2.factorial calculation
DECLARE
```



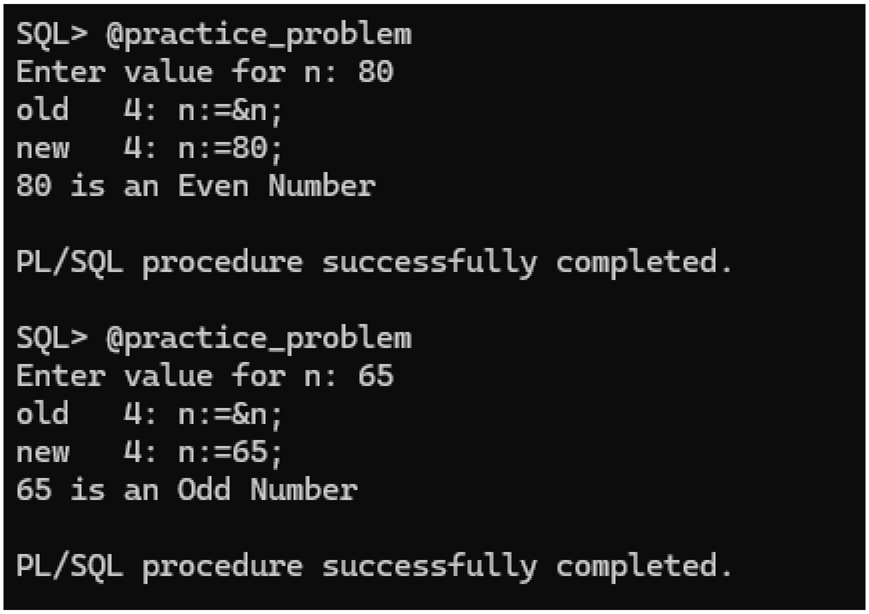
```
n number;
fact_number number:=1;
Begin
n:=&n;
FOR i IN 1..n LOOP
fact_number := fact_number * i;
END LOOP;
dbms_output.put_line('factorial of' || n || ' is: ' ||
fact_number);
END;
/
```



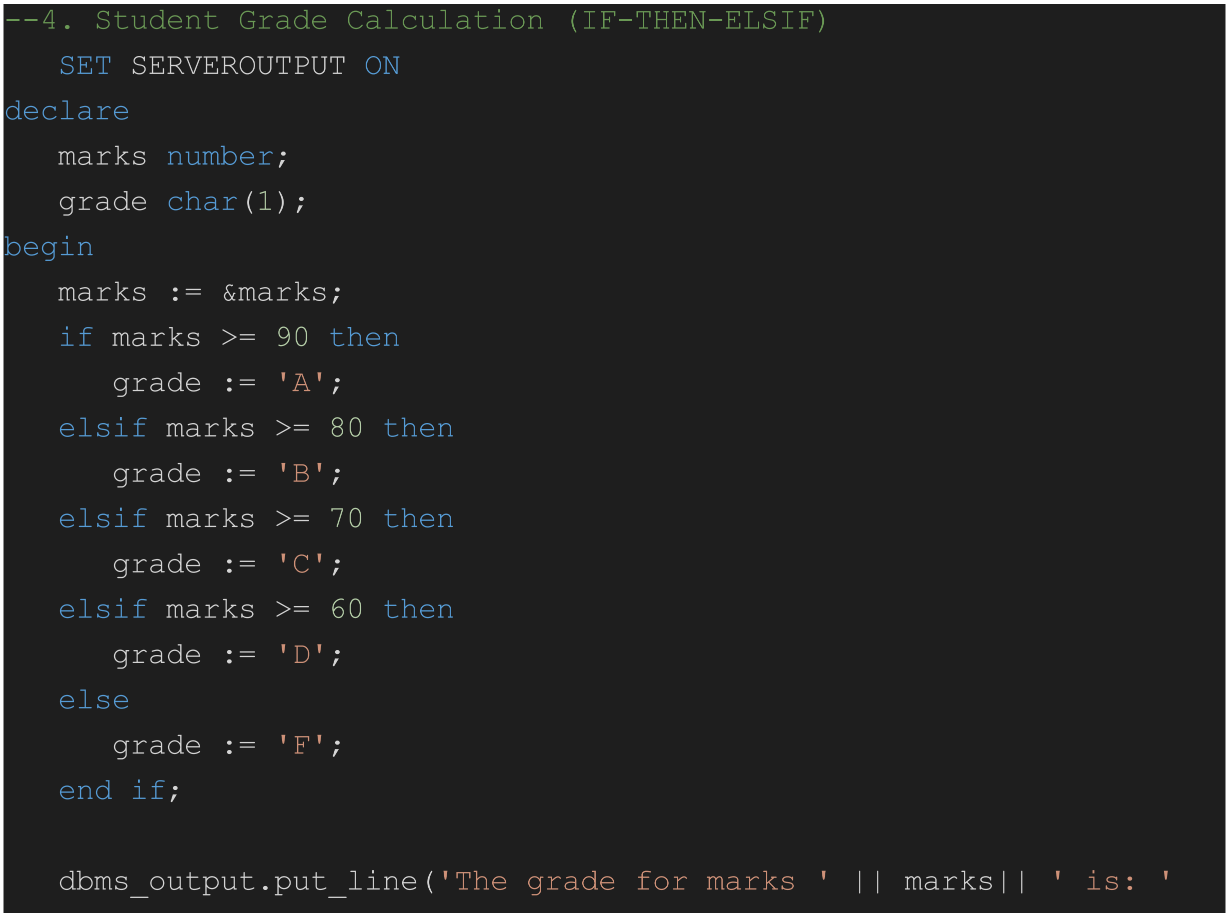
```
SQL> @practice_problem
SP2-0734: unknown command beginning "factorial ..." - rest of line ignored.
Enter value for n: 5
old   5: n:=&n;
new   5: n:=5;
factorial of5 is: 120

PL/SQL procedure successfully completed.
```



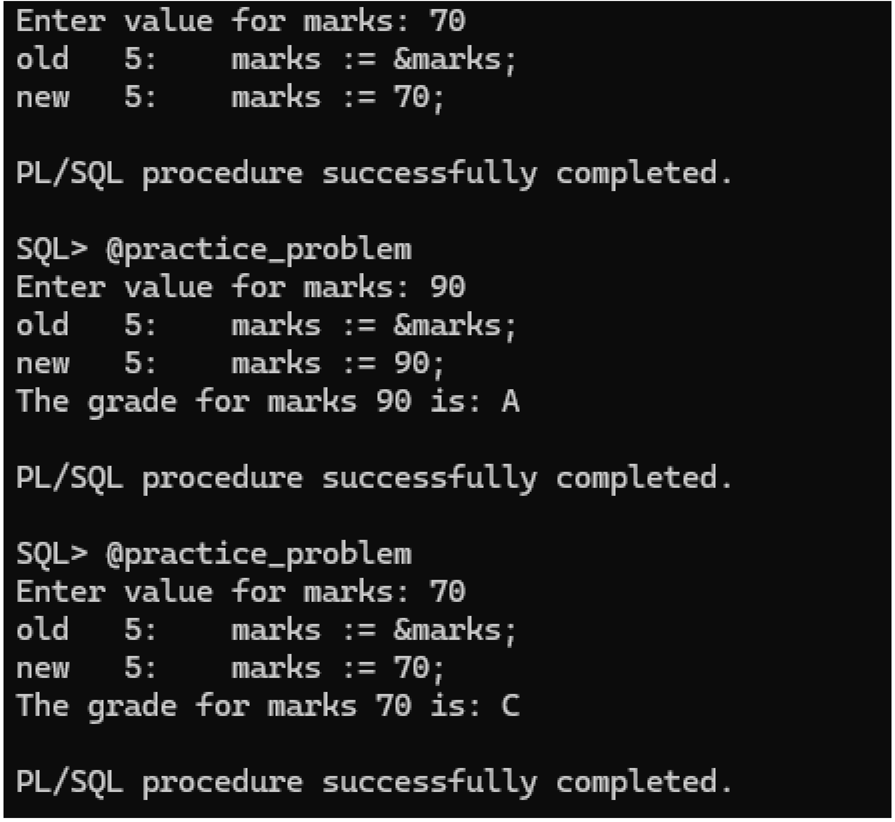
```
-- 3.Even or Odd Number Check
declare
n number;
BEGIN
n:=&n;
IF mod(n,2)=0 THEN
dbms_output.put_line(n || ' is an Even Number');
ELSE
dbms_output.put_line(n || ' is an Odd Number');
END IF;
END;
  /
```



```
SQL> @practice_problem
Enter value for n: 80
old   4: n:=&n;
new   4: n:=80;
80 is an Even Number

PL/SQL procedure successfully completed.

SQL> @practice_problem
Enter value for n: 65
old   4: n:=&n;
new   4: n:=65;
65 is an Odd Number

PL/SQL procedure successfully completed.
```

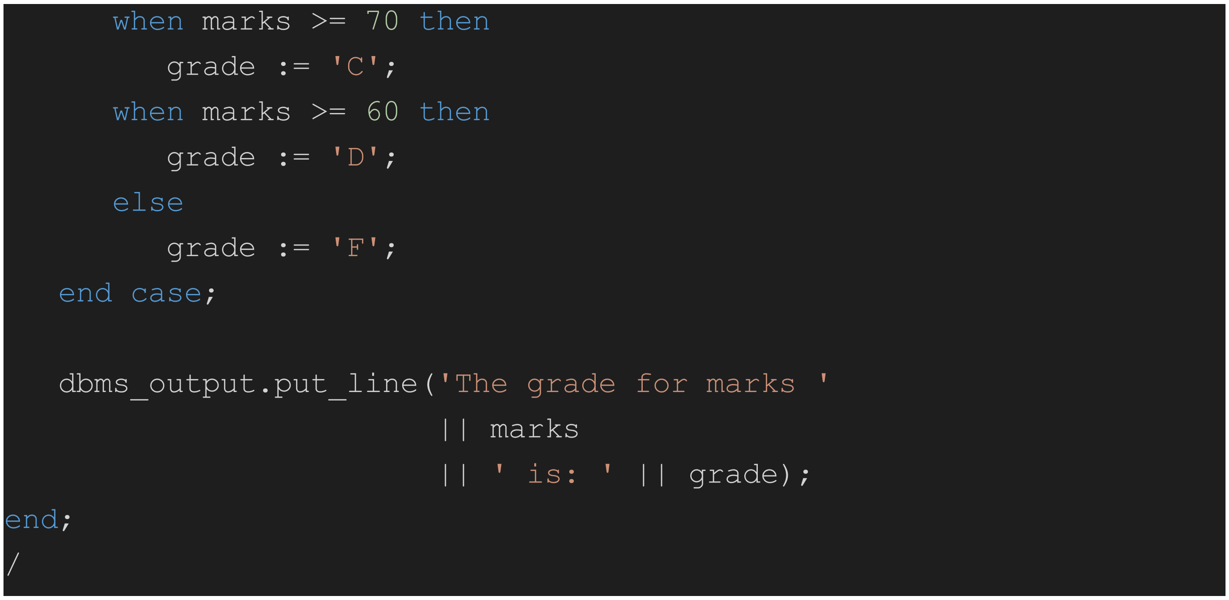


```
--4. Student Grade Calculation (IF-THEN-ELSIF)
   SET SERVEROUTPUT ON
declare
   marks number;
   grade char(1);
begin
   marks := &marks;
   if marks >= 90 then
      grade := 'A';
   elsif marks >= 80 then
      grade := 'B';
   elsif marks >= 70 then
      grade := 'C';
   elsif marks >= 60 then
      grade := 'D';
   else
      grade := 'F';
   end if;

   dbms_output.put_line('The grade for marks ' || marks|| ' is: '
|| grade);
end;
/
```

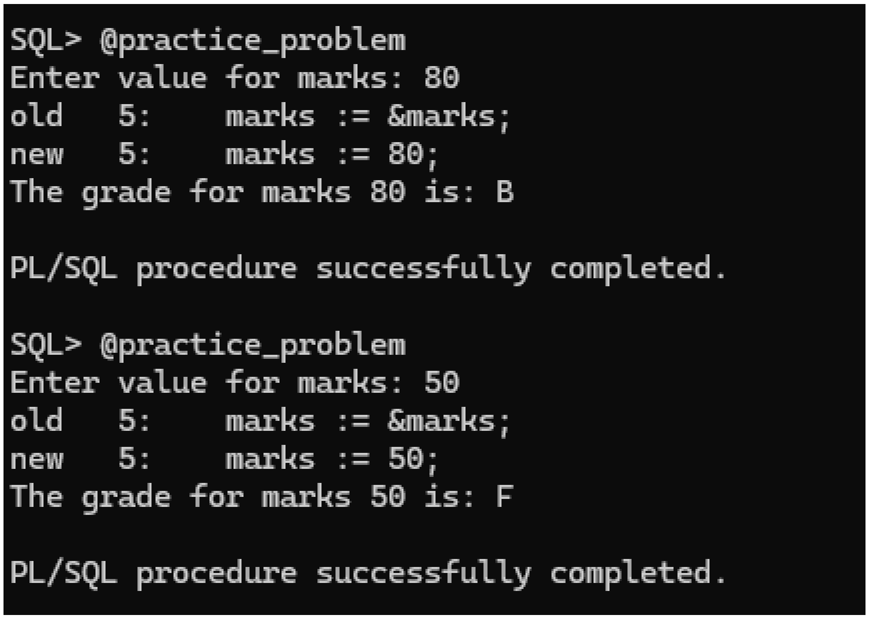


```
Enter value for marks: 70
old   5:     marks := &marks;
new   5:     marks := 70;

PL/SQL procedure successfully completed.

SQL> @practice_problem
Enter value for marks: 90
old   5:     marks := &marks;
new   5:     marks := 90;
The grade for marks 90 is: A

PL/SQL procedure successfully completed.

SQL> @practice_problem
Enter value for marks: 70
old   5:     marks := &marks;
new   5:     marks := 70;
The grade for marks 70 is: C

PL/SQL procedure successfully completed.
```

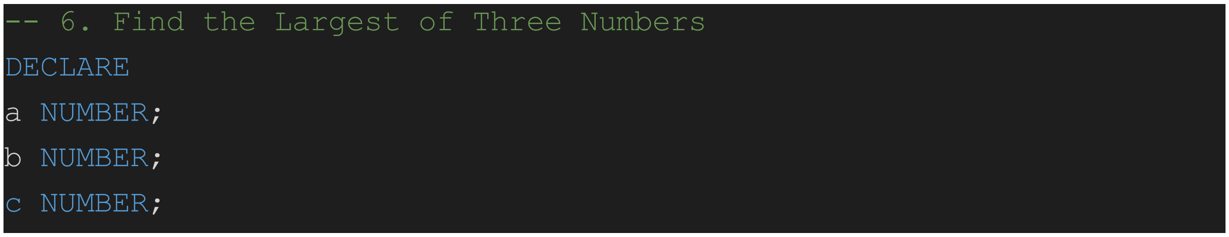
```
-- 5. Student Grade Calculation (CASE Statement)
declare
   marks number;
   grade char(1);
begin
   marks := &marks;
   case
      when marks >= 90 then
         grade := 'A';
      when marks >= 80 then
         grade := 'B';
```



```
      when marks >= 70 then
         grade := 'C';
      when marks >= 60 then
         grade := 'D';
      else
         grade := 'F';
   end case;

   dbms_output.put_line('The grade for marks '
                        || marks
                        || ' is: ' || grade);
end;
/
```

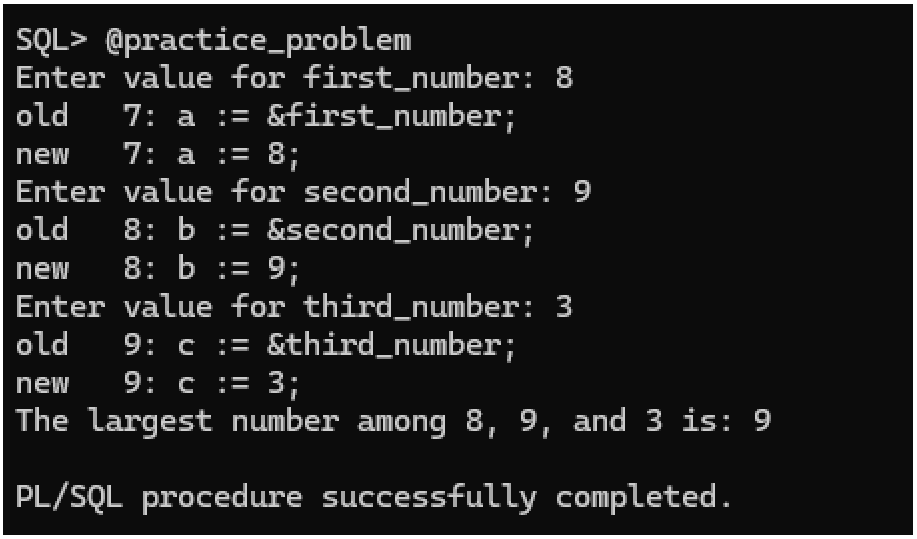


```
SQL> @practice_problem
Enter value for marks: 80
old   5:     marks := &marks;
new   5:     marks := 80;
The grade for marks 80 is: B

PL/SQL procedure successfully completed.

SQL> @practice_problem
Enter value for marks: 50
old   5:     marks := &marks;
new   5:     marks := 50;
The grade for marks 50 is: F

PL/SQL procedure successfully completed.
```



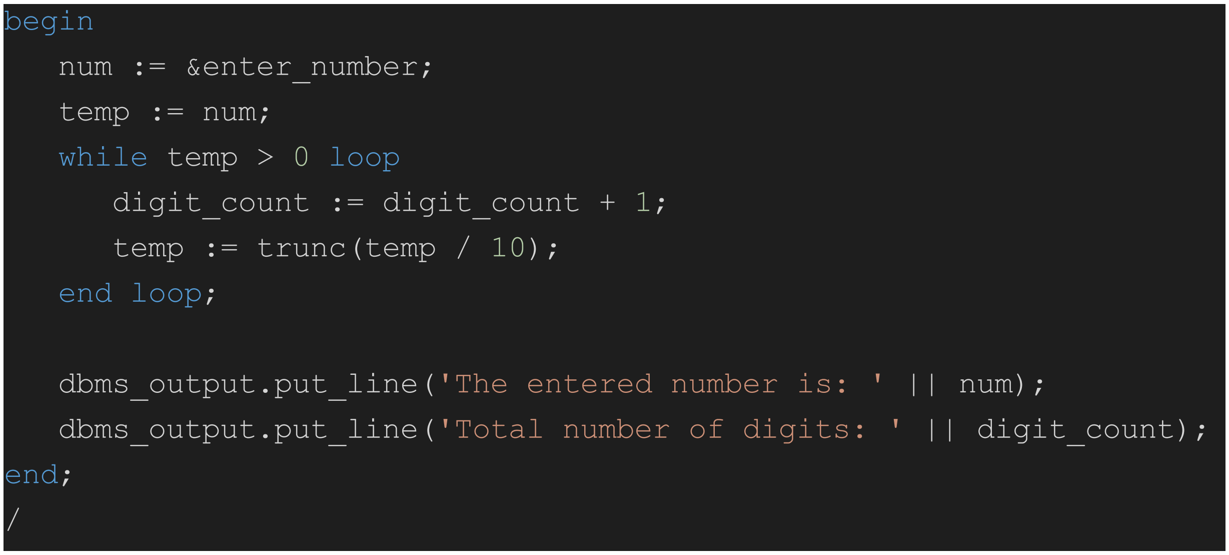
```
-- 6. Find the Largest of Three Numbers
DECLARE
a NUMBER;
b NUMBER;
c NUMBER;
```

```
largest NUMBER;
BEGIN
a := &first_number;
b := &second_number;
c := &third_number;
IF a >= b AND a >= c THEN
largest := a;
ELSIF b >= a AND b >= c THEN
largest := b;
ELSE
largest := c;
END IF;
DBMS_OUTPUT.PUT_LINE('The largest number among '|| a || ', ' || b
|| ', and ' || c || ' is: ' || largest);
END;
/
```

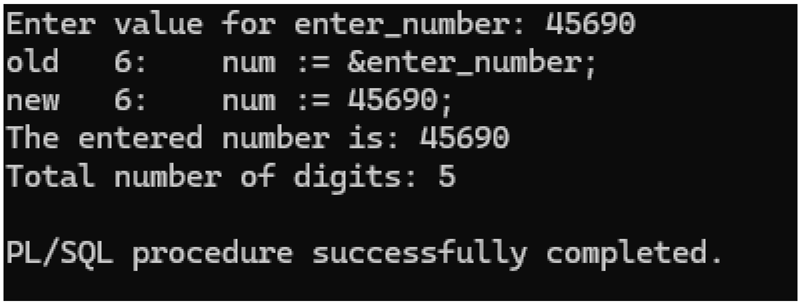


```
SQL> @practice_problem
Enter value for first_number: 8
old   7: a := &first_number;
new   7: a := 8;
Enter value for second_number: 9
old   8: b := &second_number;
new   8: b := 9;
Enter value for third_number: 3
old   9: c := &third_number;
new   9: c := 3;
The largest number among 8, 9, and 3 is: 9

PL/SQL procedure successfully completed.
```

```
-- 7. Count Digits in a Number
declare
   num         number;
   temp        number;
   digit_count number := 0;
```



```
begin
   num := &enter_number;
   temp := num;
   while temp > 0 loop
      digit_count := digit_count + 1;
      temp := trunc(temp / 10);
   end loop;

   dbms_output.put_line('The entered number is: ' || num);
   dbms_output.put_line('Total number of digits: ' || digit_count);
end;
/
```



```
Enter value for enter_number: 45690
old   6:    num := &enter_number;
new   6:    num := 45690;
The entered number is: 45690
Total number of digits: 5

PL/SQL procedure successfully completed.
```