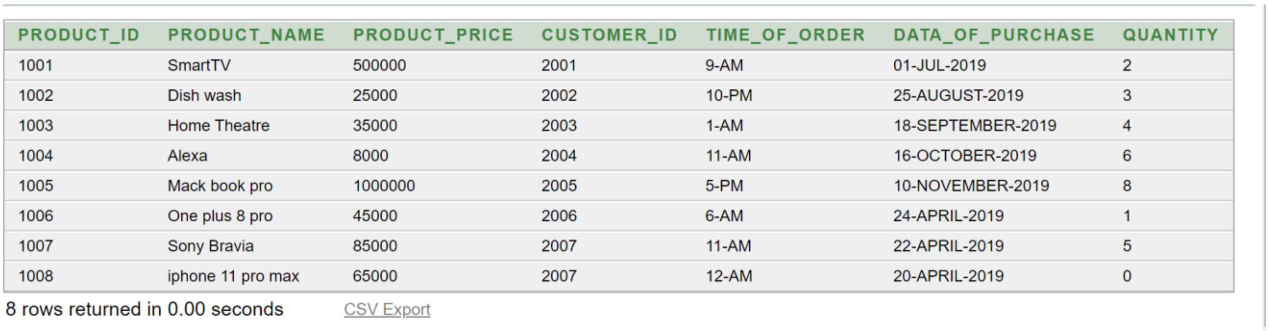
Principles of Data-Base Management System

Prashanth.S

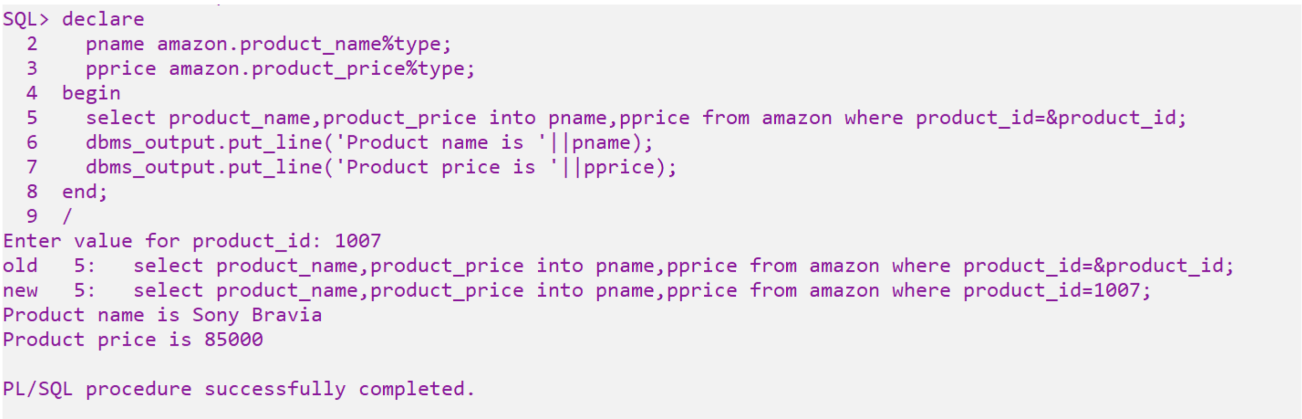
19MID0020



PRODUCT_ID	PRODUCT_NAME	PRODUCT_PRICE	CUSTOMER_ID	TIME_OF_ORDER	DATA_OF_PURCHASE	QUANTITY
1001	SmartTV	500000	2001	9-AM	01-JUL-2019	2
1002	Dish wash	25000	2002	10-PM	25-AUGUST-2019	3
1003	Home Theatre	35000	2003	1-AM	18-SEPTEMBER-2019	4
1004	Alexa	8000	2004	11-AM	16-OCTOBER-2019	6
1005	Mack book pro	1000000	2005	5-PM	10-NOVEMBER-2019	8
1006	One plus 8 pro	45000	2006	6-AM	24-APRIL-2019	1
1007	Sony Bravia	85000	2007	11-AM	22-APRIL-2019	5
1008	iphone 11 pro max	65000	2007	12-AM	20-APRIL-2019	0

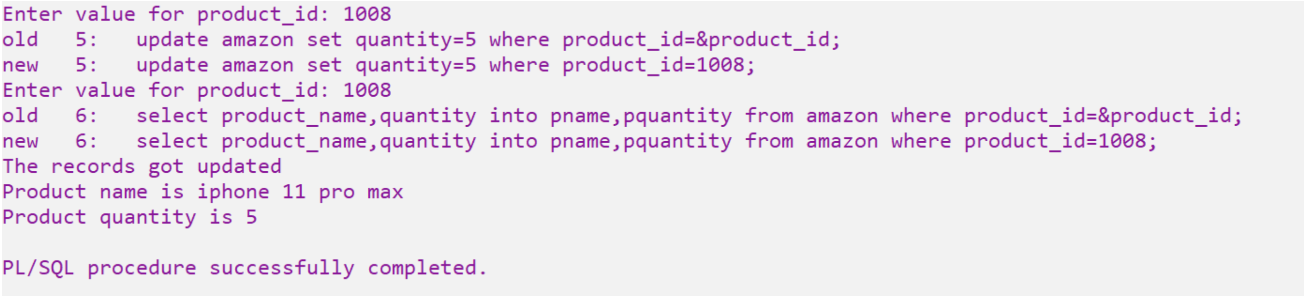
8 rows returned in 0.00 seconds CSV Export

1) Write a PL/SQL block to retrieve the product name, product price of a particular product identified by the product_id from the amazon table by reading product_id value during runtime.



```
SQL> declare
  2    pname amazon.product_name%type;
  3    pprice amazon.product_price%type;
  4  begin
  5    select product_name,product_price into pname,pprice from amazon where product_id=&product_id;
  6    dbms_output.put_line('Product name is '||pname);
  7    dbms_output.put_line('Product price is '||pprice);
  8  end;
  9  /
Enter value for product_id: 1007
old   5:   select product_name,product_price into pname,pprice from amazon where product_id=&product_id;
new   5:   select product_name,product_price into pname,pprice from amazon where product_id=1007;
Product name is Sony Bravia
Product price is 85000

PL/SQL procedure successfully completed.
```

2) Write a PL/SQL block to change the quantity to 5 for product whose product_id is 1008 interactively by reading the product_id during runtime.



```
Enter value for product_id: 1008
old   5:   update amazon set quantity=5 where product_id=&product_id;
new   5:   update amazon set quantity=5 where product_id=1008;
Enter value for product_id: 1008
old   6:   select product_name,quantity into pname,pquantity from amazon where product_id=&product_id;
new   6:   select product_name,quantity into pname,pquantity from amazon where product_id=1008;
The records got updated
Product name is iphone 11 pro max
Product quantity is 5

PL/SQL procedure successfully completed.
```

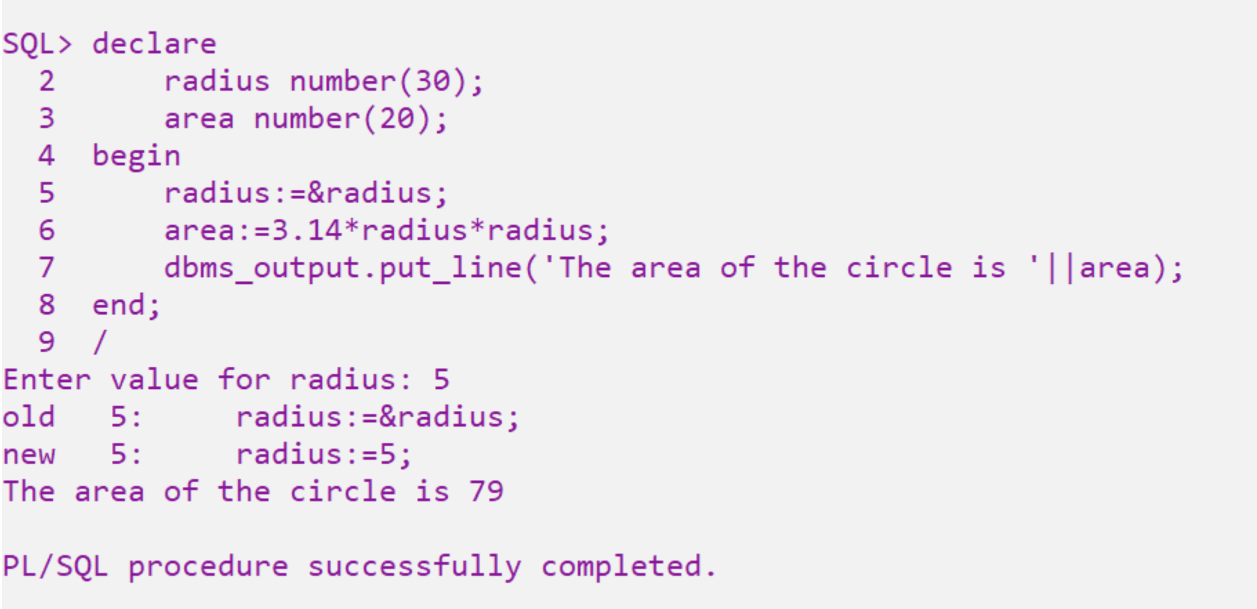
3) Write a PL/SQL block to delete a particular product record by taking its product_id interactively.

```
SQL> declare
  2    pname amazon.product_name%type;
  3  begin
  4    delete from amazon where product_id=&product_id;
  5    dbms_output.put_line('The records got deleted');
  6  end;
  7  /
Enter value for product_id: 1008
old   4:   delete from amazon where product_id=&product_id;
new   4:   delete from amazon where product_id=1008;
The records got deleted

PL/SQL procedure successfully completed.
```

4) Write a PL/SQL block to display your regno and name

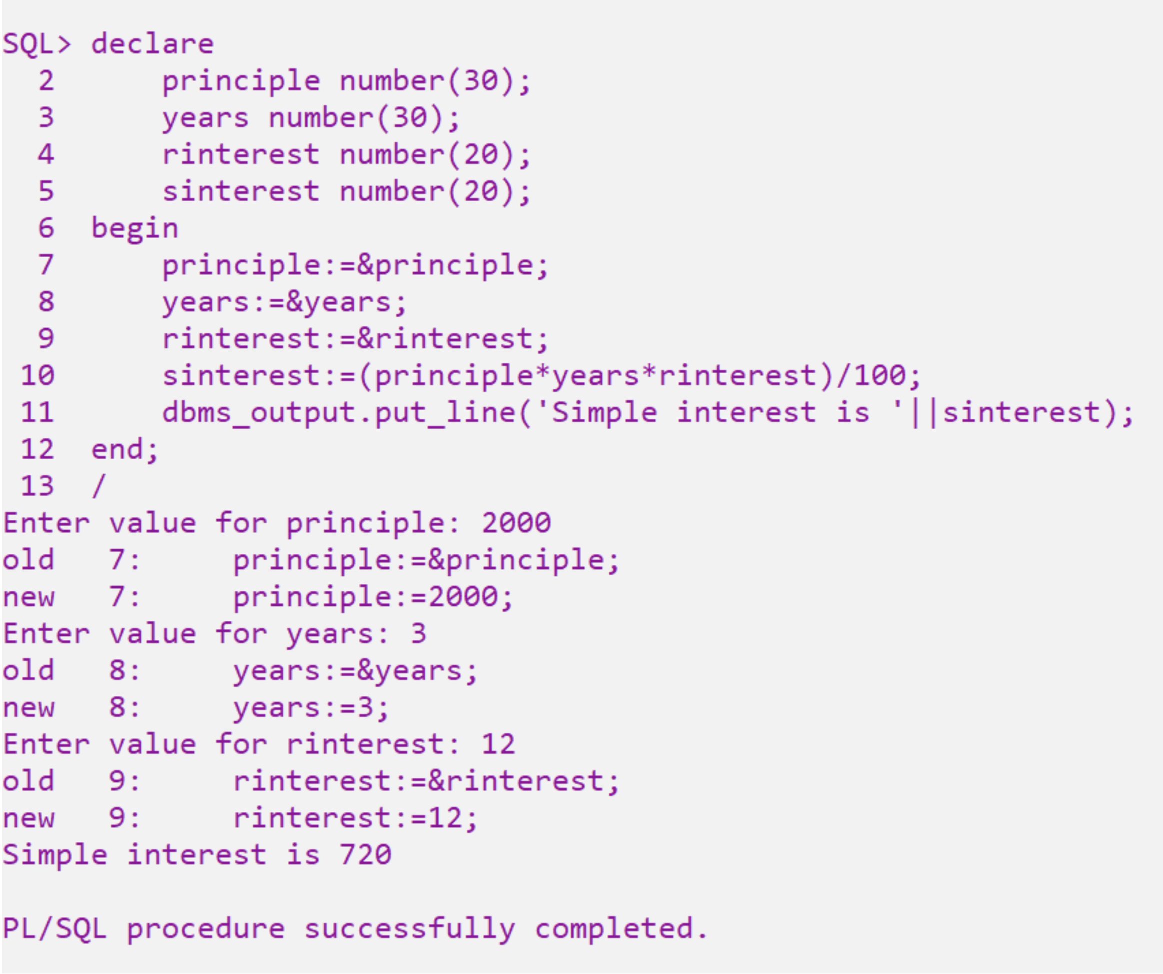
```
SQL> declare
  2      regno number(30);
  3      name varchar2(30);
  4  begin
  5      regno:=&regno;
  6      name :=&name;
  7      dbms_output.put_line('Register number is '||regno);
  8      dbms_output.put_line('Name is '||name);
  9  end;
 10  /
Enter value for regno: 2001
old   5:     regno:=&regno;
new   5:     regno:=2001;
Enter value for name: 'Prashanth'
old   6:     name :=&name;
new   6:     name :='Prashanth';
Register number is 2001
Name is Prashanth

PL/SQL procedure successfully completed.
```

5) Write a PL/SQL block to calculate area of a circle given its radius



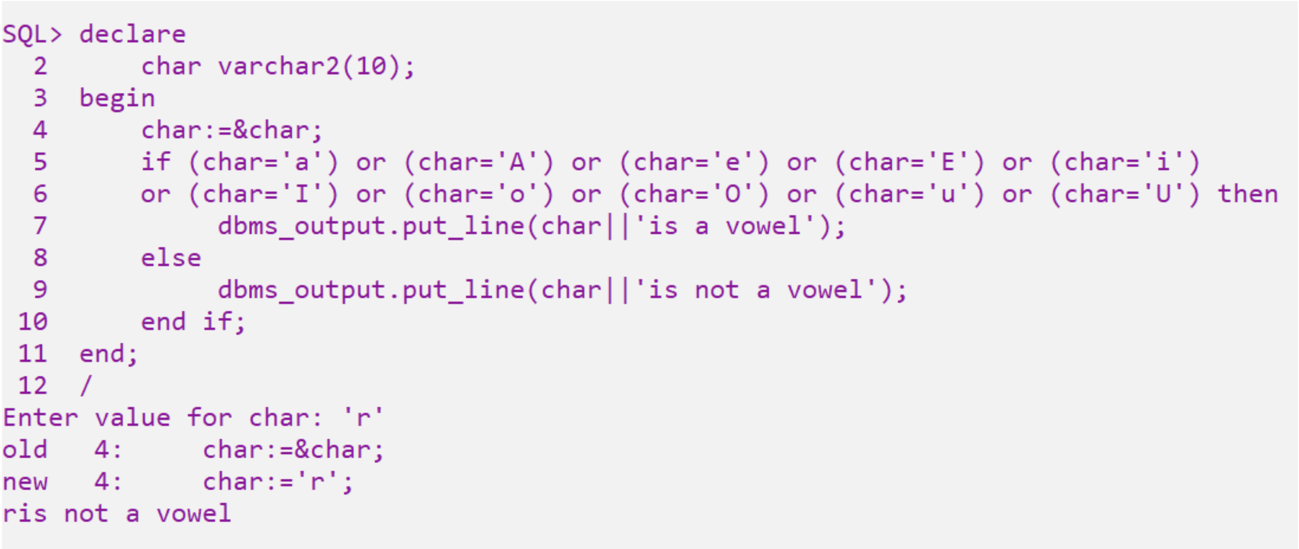
```
SQL> declare
  2      radius number(30);
  3      area number(20);
  4  begin
  5      radius:=&radius;
  6      area:=3.14*radius*radius;
  7      dbms_output.put_line('The area of the circle is '||area);
  8  end;
  9  /
Enter value for radius: 5
old   5:     radius:=&radius;
new   5:     radius:=5;
The area of the circle is 79

PL/SQL procedure successfully completed.
```

6) Write a PL/SQL block to find out Simple Interest given P=10000, N=2 and R=10% (Hint: Simple Interest(SI)=(P*N*R)/100)



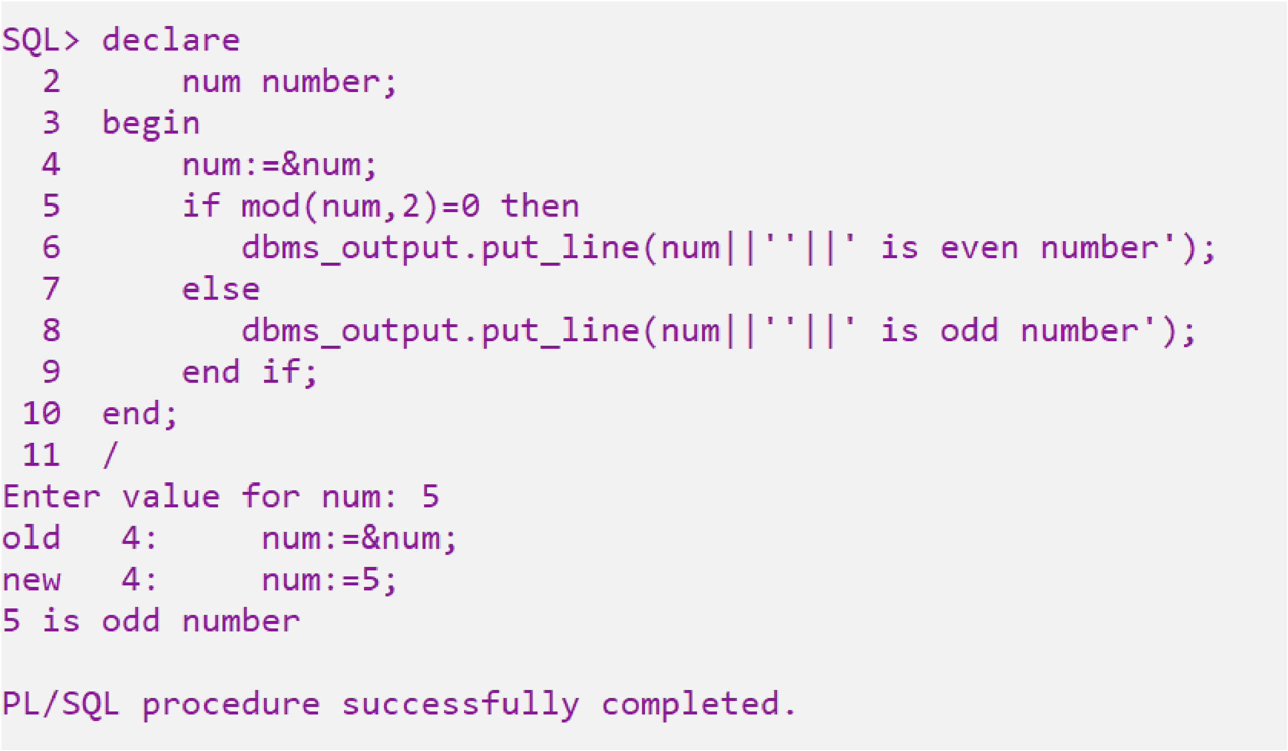
```
SQL> declare
  2      principle number(30);
  3      years number(30);
  4      rinterest number(20);
  5      sinterest number(20);
  6  begin
  7      principle:=&principle;
  8      years:=&years;
  9      rinterest:=&rinterest;
 10      sinterest:=(principle*years*rinterest)/100;
 11      dbms_output.put_line('Simple interest is '||sinterest);
 12  end;
 13  /
Enter value for principle: 2000
old   7:     principle:=&principle;
new   7:     principle:=2000;
Enter value for years: 3
old   8:     years:=&years;
new   8:     years:=3;
Enter value for rinterest: 12
old   9:     rinterest:=&rinterest;
new   9:     rinterest:=12;
Simple interest is 720

PL/SQL procedure successfully completed.
```

7) Write a PL/SQL block to check whether entered character is either vowel or consonant



```
SQL> declare
  2      char varchar2(10);
  3  begin
  4      char:=&char;
  5      if (char='a') or (char='A') or (char='e') or (char='E') or (char='i')
  6      or (char='I') or (char='o') or (char='O') or (char='u') or (char='U') then
  7          dbms_output.put_line(char||'is a vowel');
  8      else
  9          dbms_output.put_line(char||'is not a vowel');
 10      end if;
 11  end;
 12  /
Enter value for char: 'r'
old   4:     char:=&char;
new   4:     char:='r';
ris not a vowel
```

8) Write a PL/SQL block to check whether entered integer number is even or odd?

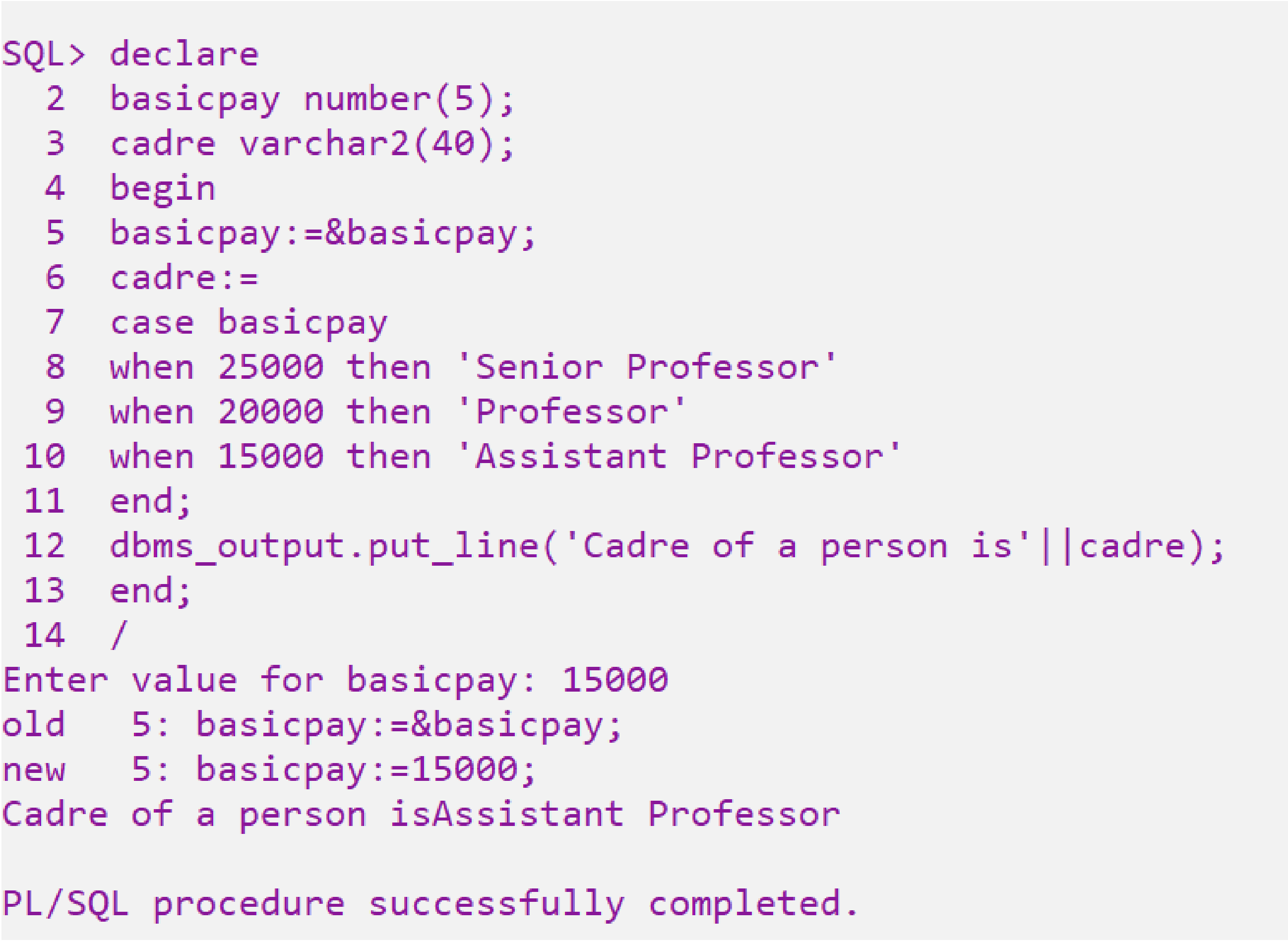


```
SQL> declare
  2      num number;
  3  begin
  4      num:=&num;
  5      if mod(num,2)=0 then
  6         dbms_output.put_line(num||''||' is even number');
  7      else
  8         dbms_output.put_line(num||''||' is odd number');
  9      end if;
 10  end;
 11  /
Enter value for num: 5
old   4:     num:=&num;
new   4:     num:=5;
5 is odd number

PL/SQL procedure successfully completed.
```

9) Write a PL/SQL program to find the smallest among three integer numbers

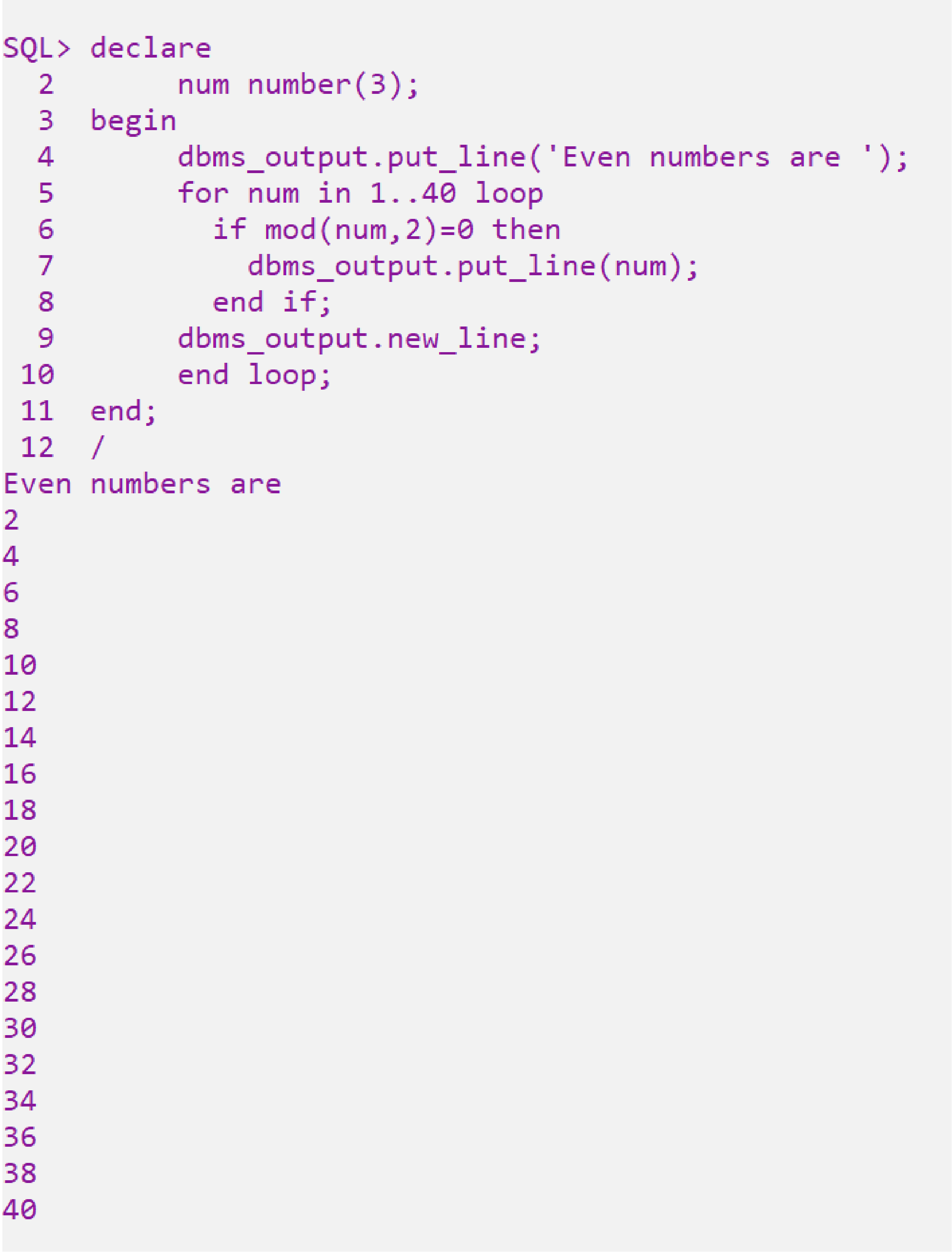
```
SQL> declare
  2       num1 number(10);
  3       num2 number(10);
  4       num3 number(10);
  5  begin
  6       num1:=&num1;
  7       num2:=&num2;
  8       num3:=&num3;
  9
 10       if (num1>num2) and (num1>num3) then
 11         dbms_output.put_line(num1||''||' is greater');
 12       elsif(num2>num3) then
 13         dbms_output.put_line(num2||''||' is greater');
 14       else
 15         dbms_output.put_line(num3||''||' is greater');
 16       end if;
 17  end;
 18  /
Enter value for num1: 21
old   6:      num1:=&num1;
new   6:      num1:=21;
Enter value for num2: 90
old   7:      num2:=&num2;
new   7:      num2:=90;
Enter value for num3: 45
old   8:      num3:=&num3;
new   8:      num3:=45;
90 is greater

PL/SQL procedure successfully completed.
```

10)Write a PL/SQL program to display cadre of an employee based on his basic pay



```
SQL> declare
  2  basicpay number(5);
  3  cadre varchar2(40);
  4  begin
  5  basicpay:=&basicpay;
  6  cadre:=
  7  case basicpay
  8  when 25000 then 'Senior Professor'
  9  when 20000 then 'Professor'
 10  when 15000 then 'Assistant Professor'
 11  end;
 12  dbms_output.put_line('Cadre of a person is'||cadre);
 13  end;
 14  /
Enter value for basicpay: 15000
old   5: basicpay:=&basicpay;
new   5: basicpay:=15000;
Cadre of a person isAssistant Professor

PL/SQL procedure successfully completed.
```

11)Write PL/SQL program to display even numbers between 1 and 40



```
SQL> declare
            num number(3);
     begin
            dbms_output.put_line('Even numbers are ');
            for num in 1..40 loop
              if mod(num,2)=0 then
                dbms_output.put_line(num);
              end if;
            dbms_output.new_line;
            end loop;
     end;
     /
Even numbers are
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
```

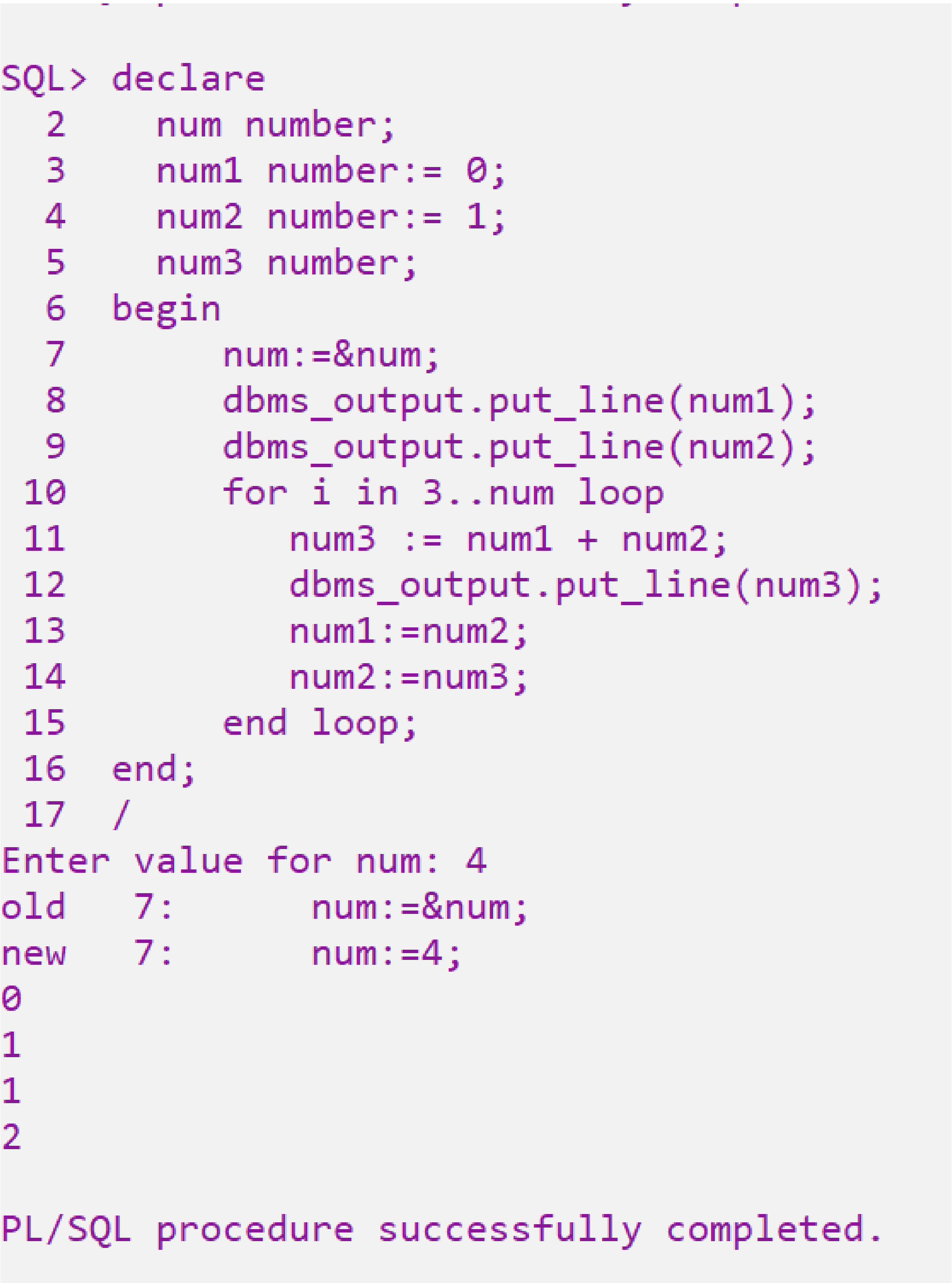
12) Write PL/SQL block to display factorial of given number

```
SQL> declare
  2      num number(3);
  3      i number(20);
  4      fact number:=1;
  5  begin
  6      num:=&num;
  7      for i in 1..num loop
  8          fact:=fact*i;
  9      end loop;
 10      dbms_output.put_line('Factorial is '||fact);
 11  end;
 12  /
Enter value for num: 5
old   6:     num:=&num;
new   6:     num:=5;
Factorial is 120

PL/SQL procedure successfully completed.
```

13) Write PL/SQL block to display reverse number of given number

```
SQL> declare
  2     num number;
  3     reverse number:=0;
  4  begin
  5     num:=&num;
  6     while (num>0) loop
  7        reverse:=reverse*10+mod(num,10);
  8        num:=trunc(num/10);
  9     end loop;
 10     dbms_output.put_line('The Reverse is'||' : '||reverse);
 11  end;
 12  /
Enter value for num: 3871
old   5:    num:=&num;
new   5:    num:=3871;
The Reverse is : 1783

PL/SQL procedure successfully completed.
```

14) Write PL/SQL block to generate Fibonacci Series up to given number



```
SQL> declare
  2    num number;
  3    num1 number:= 0;
  4    num2 number:= 1;
  5    num3 number;
  6  begin
  7       num:=&num;
  8       dbms_output.put_line(num1);
  9       dbms_output.put_line(num2);
 10       for i in 3..num loop
 11          num3 := num1 + num2;
 12          dbms_output.put_line(num3);
 13          num1:=num2;
 14          num2:=num3;
 15       end loop;
 16  end;
 17  /
Enter value for num: 4
old   7:      num:=&num;
new   7:      num:=4;
0
1
1
2

PL/SQL procedure successfully completed.
```