ANUBHAV SINGH

IC2K1654

PLSQL ASSIGNMENT 1

1. Write a program that declares and assigns values to the variables a, b, and c, and then does the following:-

   Halves the value of a, doubles b, multiplies c by itself. Display the output of the program on the screen using dbms_output.put_line.

```
declare
a int;
b int;
c int;
begin
a:=&a;
b:=&b;
c:=&c;
dbms_output.put_line('Half of a is '||a*0.5);
dbms_output.put_line('Double of b is '||2*b);
dbms_output.put_line('Multiply of c by itself is '||c*c);
end;
/
```

OUTPUT

```
SQL> @assgn1
Enter value for a: 10
old   6: a:=&a;
new   6: a:=10;
Enter value for b: 20
old   7: b:=&b;
new   7: b:=20;
Enter value for c: 20
old   8: c:=&c;
new   8: c:=20;
Half of a is 5
Double of b is 40
Multiply of c by itself is 400

PL/SQL procedure successfully completed.
```

2. Write a program that computes the perimeter and the area of a rectangle.Define your own values for the length and width. (Assuming that L and W are the length and width of the rectangle, Perimeter = 2*(L+W) and Area = L*W. Display the output on the screen using dbms_output.put_line.

```
declare
l int;
w int;
begin
```

```
l:=&l;
w:=&w;
dbms_output.put_line('Perimeter of Rectangle is '||2*(l+w));
dbms_output.put_line('Area of Rectangle is '||l*w);
end;
/
```

OUTPUT

```
SQL> @assgn2
Enter value for l: 10
old   5: l:=&l;
new   5: l:=10;
Enter value for w: 20
old   6: w:=&w;
new   6: w:=20;
Perimeter of Rectangle is 60
Area of Rectangle is 200

PL/SQL procedure successfully completed.
```

3. Suppose you had to write a block to compute the volume of a cube. The values you would need are the three dimensions of the cube. Think up four appropriate variable names to be used in the program – three variables to hold the three dimensions, and one for the result. (Assuming that L, W and H are the dimensions of a cube, Volume = L*W*H). Display the output on the screen using dbms_output.put_line. L, W and H are to be input by the user.

```
declare
l real;
w real;
b real;
volume real;
begin
l:=&l;
w:=&w;
b:=&b;
volume:=l*b*w;
dbms_output.put_line('volume of cube is '||volume);
end;
/
```

OUTPUT

```
SQL> @assgn3
Enter value for l: 10
old   7: l:=&l;
new   7: l:=10;
Enter value for w: 20
old   8: w:=&w;
new   8: w:=20;
Enter value for b: 30
old   9: b:=&b;
new   9: b:=30;
volume of cube is 6000

PL/SQL procedure successfully completed.
```

4. Write a program that declares an integer variable called num, assigns a value to it,and computes and inserts into the tempp table the value of the variable itself, its square and its cube

```
declare
num int:=&num;
temp int;
begin
temp:=num;
dbms_output.put_line('Value of variable '||temp||' Square of variable is
'||temp*temp||' Cube of variable is '||temp*temp*temp);
end;
/
```

OUTPUT

```
SQL> @assgn4
Enter value for num: 20
old   2: num int:=&num;
new   2: num int:=20;
Value of variable 20 Square of variable is 400 Cube of variable is 8000

PL/SQL procedure successfully completed.
```

5. Convert a temperature in Fahrenheit (F) to its equivalent in Celsius (C) and vice versa.

   The required formulae are:-

   C= (F-32)*5/9

   F= 9/5*C + 32

   Display the output on the screen using dbms_output.put_line. Data has to be input by the user.

```
declare
f number:=&f;
c number:=&c;
```

```
total number;
begin
total:=(f-32)*5/9;
dbms_output.put_line('Fahrenheit entered '||f||' equals to celsius '||total);
total:=9/5*c+32;
dbms_output.put_line('Celsius entered '||f||' equals to Fahrenheit '||total);
end;
/
```

OUTPUT

```
SQL> @assgn5
Enter value for f: 98.6
old   2: f number:=&f;
new   2: f number:=98.6;
Enter value for c: 37
old   3: c number:=&c;
new   3: c number:=37;
Fahrenheit entered 98.6 equals to celsius 37
Celsius entered 98.6 equals to Fahrenheit 98.6

PL/SQL procedure successfully completed.
```

6. Convert a given number of days to a measure of time given in years, weeks, and days. For example, 375 days equals 1 year, 1 week and 3 days. (Ignore leap year Display the output on the screen using dbms_output.put_line. Data has to be input by the user.

```
declare
num int:=&num;
year int;
week int;
day int;
temp int;
begin
temp:=MOD(num,365);
year:=num/365;
week:=temp/7;
day:=MOD(temp,7);
dbms_output.put_line(num ||' is equivalent to '||year||' year '||week||' week '||day||'
days');
end;
/
```

OUTPUT

```
SQL> @assgn6
Enter value for num: 1000
old   2: num int:=&num;
new   2: num int:=1000;
1000 is equivalent to 3 year 39 week 4 days

PL/SQL procedure successfully completed.
```

7. Convert a number of inches into yards, feet, and inches. For example, 124 inches equals 3 yards, 1 foot, and 4 inches. Display the output on the screen using dbms_output.put_line. Data has to be input by the user.

```
declare
inch number:=&inch;
yard number;
feet number;
begin
yard:=inch/36;
feet:=inch/12;
dbms_output.put_line(inch||' is equivalent to '||yard||' yards '||feet||' feets '||inch||'
inches');
end;
/
```

OUTPUT

```
SQL> @assgn7
Enter value for inch: 3600
old   2: inch number:=&inch;
new   2: inch number:=3600;
3600 is equivalent to 100 yards 300 feets 3600 inches

PL/SQL procedure successfully completed.
```

8. Add up five amounts of money (Rs. and paise) represented as float numbers, and print the result as a truncated integer value. Display the output on the screen using dbms_output.put_line. Data has to be input by the user.

```
declare
n1 number;
n2 number;
n3 number;
n4 number;
n5 number;
total number;
begin
```

```
n1:=&n1;
n2:=&n2;
n3:=&n3;
n4:=&n4;
n5:=&n5;
total:=n1+n2+n3+n4+n5;
dbms_output.put_line('Entered amount is in Rs. and paise');
dbms_output.put_line('Total amount is '||total||' rs.');
dbms_output.put_line('Trucated amount is '||TRUNC(total,0)||' rs.');
end;
/
```

OUTPUT

```
SQL> @assgn8
Enter value for n1: 10
old   9: n1:=&n1;
new   9: n1:=10;
Enter value for n2: 20
old  10: n2:=&n2;
new  10: n2:=20;
Enter value for n3: 30
old  11: n3:=&n3;
new  11: n3:=30;
Enter value for n4: 40
old  12: n4:=&n4;
new  12: n4:=40;
Enter value for n5: 50
old  13: n5:=&n5;
new  13: n5:=50;
Entered amount is in Rs. and paise
Total amount is 150 rs.
Trucated amount is 150 rs.

PL/SQL procedure successfully completed.
```

9. Write a program that enables a user to input an integer. The program should then state whether the integer is evenly divisible by 5. (Use decode instead of IF statement where required). Display the output on the screen using dbms_output.put_line. Data has to be input by the user.

```
declare
num number:=&num;
total varchar2(50);
begin
select decode(MOD (num,5),0,'It is divisible','It is not divisible')
into total
from dual;
dbms_output.put_line(total);
end;
/
```

OUTPUT

```
SQL> @assgn9
Enter value for num: 20
old   2: num number:=&num;
new   2: num number:=20;
It is divisible

PL/SQL procedure successfully completed.
```

10. Your block should read in two real numbers and tell whether the product of the two numbers is equal to or greater than 100. Display the output on the screen using dbms_output.put_line. (Use decode instead of IF statement where required). Data has to be input by the user.

```
declare
firstno number:=&firstno;
secondno number:=&secondno;
total varchar2(50);
begin
select decode(trunc(firstno*secondno/100),0,'less then 100','greater then or equal to 100')
into total from dual;
dbms_output.put_line(total);
end;
/
```

OUTPUT

```
SQL> @assgn10
Enter value for firstno: 50
old   2: firstno number:=&firstno;
new   2: firstno number:=50;
Enter value for secondno: 60
old   3: secondno number:=&secondno;
new   3: secondno number:=60;
greater then or equal to 100

PL/SQL procedure successfully completed.
```