Principles of Database Management Systems

Pl SQL commands Prashanth.S(19MID0020)

1) To check whether a person is male or female

```
SQL> declare
  2      gender1 varchar2(30);
  3  begin
  4      gender1:=&gender1;
  5
  6      if (gender1='f') or (gender1='F') or (gender1='female')
  7          or (gender1='Female') or (gender1='FEMALE') then
  8          dbms_output.put_line('Gender is Female');
  9      elsif (gender1='m') or (gender1='M') or (gender1='male')
 10          or (gender1='Male') or (gender1='MALE') then
 11          dbms_output.put_line('Gender is Male');
 12      else
 13          dbms_output.put_line('Gender is not matching');
 14      end if;
 15  end;
 16
 17  /
```

```
Enter value for gender1: 'male'
old   4:      gender1:=&gender1;
new   4:      gender1:='male';
Gender is Male

PL/SQL procedure successfully completed.
```

```
Enter value for gender1: 'f'
old   4:      gender1:=&gender1;
new   4:      gender1:='f';
Gender is Female

PL/SQL procedure successfully completed.
```

```
Enter value for gender1: 'none'
old   4:      gender1:=&gender1;
new   4:      gender1:='none';
Gender is not matching

PL/SQL procedure successfully completed.
```

2) To check whether a person is major/minor

```
SQL> declare
  2      age number(10);
  3  begin
  4      age:=&age;
  5      if ((age>=1) and (age<=120)) then
  6        if (age>=18) then
  7          dbms_output.put_line('The person is major');
  8        elsif (age<18) then
  9          dbms_output.put_line('The person is minor');
 10        end if;
 11      else
 12          dbms_output.put_line('Age is not valid');
 13      end if;
 14  end;
 15  /
```

```
Enter value for age: -9
old   4:      age:=&age;
new   4:      age:=-9;
Age is not valid
```

```
Enter value for age: 19
old   4:      age:=&age;
new   4:      age:=19;
The person is major
```

```
Enter value for age: 17
old   4:      age:=&age;
new   4:      age:=17;
The person is minor
```

3) To check whether a student attained ‘S’ grade or not

```
SQL> declare
  2      marks number(10);
  3  begin
  4      marks:=&marks;
  5      if marks>=90 then
  6          dbms_output.put_line('S-grade');
  7      elsif marks>=80 and marks<90 then
  8          dbms_output.put_line('A-grade');
  9      elsif marks>=70 and marks<80 then
 10          dbms_output.put_line('B-grade');
 11      end if;
 12  end;
 13  /
Enter value for marks: 90
old   4:      marks:=&marks;
new   4:      marks:=90;
S-grade

PL/SQL procedure successfully completed.
```

```
Enter value for marks: 89
old   4:      marks:=&marks;
new   4:      marks:=89;
A-grade

PL/SQL procedure successfully completed.
```

4) To check whether a person is senior citizen/not

```
SQL> declare
  2      age number(10);
  3  begin
  4      age:=&age;
  5      if ((age>=1) and (age<=120)) then
  6        if (age>=60) then
  7          dbms_output.put_line('The person comes under Senior Citizen');
  8        elsif (age<60) then
  9          dbms_output.put_line('The person will not come under Senior Citizen');
 10        end if;
 11      else
 12          dbms_output.put_line('Age is not valid');
 13      end if;
 14  end;
 15  /
Enter value for age: 61
old   4:     age:=&age;
new   4:     age:=61;
The person comes under Senior Citizen

PL/SQL procedure successfully completed.
```

```
Enter value for age: 56
old   4:      age:=&age;
new   4:      age:=56;
The person will not come under Senior Citizen

PL/SQL procedure successfully completed.
```

```
Enter value for age: -4
old   4:      age:=&age;
new   4:      age:=-4;
Age is not valid

PL/SQL procedure successfully completed.
```

5) To check whether a student is fail/not

```
SQL> declare
  2      marks number(10);
  3  begin
  4      marks:=&marks;
  5      if marks>=50 then
  6          dbms_output.put_line('The student is pass');
  7      else
  8          dbms_output.put_line('The student is fail');
  9      end if;
 10  end;
 11  /
Enter value for marks: 89
old   4:      marks:=&marks;
new   4:      marks:=89;
The student is pass

PL/SQL procedure successfully completed.
```

```
Enter value for marks: 48
old   4:      marks:=&marks;
new   4:      marks:=48;
The student is fail

PL/SQL procedure successfully completed.
```

6) To display grade of student based on the CGPA

```
SQL> declare
  2       cgpa number(10);
  3  begin
  4       cgpa:=&cgpa;
  5       if (cgpa>=9 and cgpa<=9.5) then
  6           dbms_output.put_line('S-grade');
  7       elsif (cgpa>=8 and cgpa<9) then
  8           dbms_output.put_line('A-grade');
  9       elsif (cgpa>=7 and cgpa<8) then
 10           dbms_output.put_line('B-grade');
 11       end if;
 12  end;
 13  /
Enter value for cgpa: 9.1
old   4:      cgpa:=&cgpa;
new   4:      cgpa:=9.1;
S-grade

PL/SQL procedure successfully completed.
```

```
Enter value for cgpa: 7
old   4:      cgpa:=&cgpa;
new   4:      cgpa:=7;
B-grade

PL/SQL procedure successfully completed.
```

7) To find the smallest of the three integer numbers

```
SQL> declare
  2      num1 number(5);
  3      num2 number(5);
  4      num3 number(5);
  5
  6  begin
  7      num1:=&num1;
  8      num2:=&num2;
  9      num3:=&num3;
 10
 11      if (num1<num2) and (num2<num3) then
 12          dbms_output.put_line('The smallest number is : '||num1);
 13      elsif (num2<num1) and (num2<num3) then
 14          dbms_output.put_line('The smallest number is : '||num2);
 15      else
 16          dbms_output.put_line('The smallest number is : '||num3);
 17      end if;
 18  end;
 19  /
Enter value for num1: 89
old   7:      num1:=&num1;
new   7:      num1:=89;
Enter value for num2: 32
old   8:      num2:=&num2;
new   8:      num2:=32;
Enter value for num3: 79
old   9:      num3:=&num3;
new   9:      num3:=79;
The smallest number is : 32

PL/SQL procedure successfully completed.
```

8) To display cadre of an employee based on his basic pay

Basic Pay(Rs.)	Cadre
25000	Senior Prosser
20000	Professor
15000	Assistant Professor

```
SQL> declare
  2      pay  number(10);
  3  begin
  4      pay:=&pay;
  5      if (pay=25000) then
  6          dbms_output.put_line('Senior Professor');
  7      elsif (pay=20000) then
  8          dbms_output.put_line('Professor');
  9      elsif (pay=15000) then
 10          dbms_output.put_line('Assistant Professor');
 11      end if;
 12  end;
 13  /
Enter value for pay: 20000
old   4:      pay:=&pay;
new   4:      pay:=20000;
Professor

PL/SQL procedure successfully completed.
```

```
Enter value for pay: 1500
old   4:      pay:=&pay;
new   4:      pay:=1500;

PL/SQL procedure successfully completed.
```

```
Enter value for pay: 25000
old   4:      pay:=&pay;
new   4:      pay:=25000;
Senior Professor

PL/SQL procedure successfully completed.
```

9) To find the sum of the digits of given ‘n’ digit integer number

```
SQL> declare
  2       num1 number(10);
  3       sum1 number(10):=0;
  4  begin
  5       num1:=&num1;
  6       while(num1>0) loop
  7         sum1:= sum1+mod(num1,10);
  8         num1:= trunc(num1/10);
  9       end loop;
 10       dbms_output.put_line('The sum is : '||sum1);
 11  end;
 12  /
Enter value for num1: 123
old   5:      num1:=&num1;
new   5:      num1:=123;
The sum is : 6

PL/SQL procedure successfully completed.
```

10) To find the factorial of number between certain range

```
SQL> declare
  2      fact number:=1;
  3      str varchar2(100);
  4      lower_range number;
  5      upper_range number;
  6  begin
  7      lower_range:=&lower_range;
  8      upper_range:=&upper_range;
  9      for i in lower_range..upper_range
 10      loop
 11      for j in 1..i
 12      loop
 13      fact:=fact*j;
 14      str:=j||'*'||str;
 15      end loop;
 16      dbms_output.put_line('The factorial of '||i||' : '||rtrim(str,'*')||'='||fact);
 17  fact:=1;
 18  str:=null;
 19  end loop;
 20  end;
 21
 22  /
Enter value for lower_range: 5
old   7:     lower_range:=&lower_range;
new   7:     lower_range:=5;
Enter value for upper_range: 10
old   8:     upper_range:=&upper_range;
new   8:     upper_range:=10;
The factorial of 5 : 5*4*3*2*1=120
The factorial of 6 : 6*5*4*3*2*1=720
The factorial of 7 : 7*6*5*4*3*2*1=5040
The factorial of 8 : 8*7*6*5*4*3*2*1=40320
The factorial of 9 : 9*8*7*6*5*4*3*2*1=362880
The factorial of 10 : 10*9*8*7*6*5*4*3*2*1=3628800
```

11) To check whether a given integer is palindrome/not

```
SQL> declare
  2      num number(5);
  3      old number(5);
  4      r number:=0;
  5  begin
  6      num:=&num;
  7      old:=num;
  8      while (num>0) loop
  9          r:=r*10+mod(num,10);
 10          num:=trunc(num/10);
 11      end loop;
 12      if (old = r) then
 13          dbms_output.put_line('The given number is Palindrome');
 14      else
 15          dbms_output.put_line('The given number is not Palindrome');
 16  end if;
 17  end;
 18  /
Enter value for num: 121
old   6:      num:=&num;
new   6:      num:=121;
The given number is Palindrome

PL/SQL procedure successfully completed.
```