```
SET SERVEROUTPUT ON

select user from dual;
SELECT NAME FROM v$database;
select table_name from dba_tables;
select owner, table_name from all_tables;
select table_name FROM user_tables;
SELECT table_name FROM user_tables ORDER BY table_name;

create table Stud(rollno int primary key,
                  name char(10),
                  mark1 float,
                  mark2 float,
                  mark3 float);

/* ROLLNO NAME MARK1 MARK2 MARK3
---------- ---------- --------- ---------- ----------
1 aparna 80 90 78
2 amritha 90 92 81
3 binuja 23 18 20
4 cathy 49 50 50
5 danish 60 62 61
6 fayas 76 62 74 */

insert into Stud values(1,'aparna', 80, 90, 78);
insert into Stud values(&rollno, '&name', &mark1, &mark2, &mark3);
select * from Stud;
select name from Stud where rollno=1;


DECLARE
    age integer;
    name VARCHAR(20);
BEGIN
    dbms_output.put_line('Hello world');
    --dbms_output.put_line('age = ' || age);
    --dbms_output.put_line('name = ' || name);
    --insert into Stud values(&rollno, '&name', &mark1, &mark2, &mark3);
END;

select * from Stud;
delete from Stud;
drop table Stud purge;

/* addition of two numbers */
DECLARE
    a integer := &a;
    b integer := &b;
    res int;
BEGIN
    res := a + b;
    dbms_output.put_line('result = ' || res);
END;

/*
IF-THEN, IF-THEN-ELSE, IF-THEN
ELSIF, CASE, WHILE
*/

-- find the largest of two numbers
DECLARE
    a integer := &a;
    b integer := &b;
BEGIN

   if (a > b) then
        dbms_output.put_line(a || ' is the largest number');
   else
        dbms_output.put_line(b || ' is the largest number');
   end if;
END;


/* Use of if elsif ladder */
/* calculate the range of input number */
DECLARE
    c integer := &c;
BEGIN
    if (c >= 0 and c < 10) then
        dbms_output.put_line(' is less than 10');
    elsif (c >= 10 and c < 20) then
        dbms_output.put_line(' is less than 20');
    elsif (c >= 20 and c < 30) then
        dbms_output.put_line(' is less than 30');
    else
        dbms_output.put_line(' is grater than or equal 30');
    end if;
END;


/* Case statement */
/*
    Grading ..!
    A- > print "Excellent"
    B- > print "Very good"
    C- > print "Well done"
    D- > print "You passed"
    F- > print "Better try again"
    others  - "No such grade"

*/
DECLARE
    c char(1) := '&c';
BEGIN
    case c
        when 'A' then dbms_output.put_line('Excellent');
        when 'B' then dbms_output.put_line('Very good');
        when 'C' then dbms_output.put_line('Well done');
        when 'D' then dbms_output.put_line('You passed');
        when 'F' then dbms_output.put_line('Better try again');
        else dbms_output.put_line('No such grade');
    end case;
END;


/* use of array*/
/* index starts with 1 .. not 0 */
DECLARE
    type intArray IS VARRAY(10) OF INTEGER;
    type namesArray IS VARRAY(5) OF VARCHAR2(5);

    arr intArray;
    names namesArray;
    i integer;

BEGIN
    arr :=  intArray(1,5,2,3,6,7,4,8,9,10);
    names := namesArray('Alice', 'Bob', 'Cindy', 'Sam', 'Eric');
    dbms_output.put_line('Integer array');
    --dbms_output.put_line('arr[0]' || arr(0));
    dbms_output.put_line('arr[1]' || arr(1));
    dbms_output.put_line('arr[2]' || arr(2));
    dbms_output.put_line('arr[3]' || arr(3));
END;

/* Use of loops */
DECLARE
    type intArray IS VARRAY(10) OF INTEGER;
    type namesArray IS VARRAY(5) OF VARCHAR2(5);

    arr intArray;
    names namesArray;
    i integer;

BEGIN
    arr :=  intArray(1,5,2,3,6,7,4,8,9,10);
    names := namesArray('Alice', 'Bob', 'Cindy', 'Sam', 'Eric');

    i := 1;
    /*While loop...! */
    while( i <= 10) loop
        dbms_output.put_line('arr[' || i ||'] =' ||arr(i));
        i := i+1;
    end loop;

    /*For loop...! */
    for i in 1 .. 10 loop
        dbms_output.put_line('arr[' || i ||'] =' ||arr(i));
    end loop;

    i := 1;
    while( i <= 5) loop
        dbms_output.put_line('names[' || i ||'] =' ||names(i));
        i := i+1;
    end loop;

    /*For loop...! */
    for i in 1 .. 5 loop
        dbms_output.put_line('names[' || i ||'] =' ||names(i));
    end loop;
END;


/* Use of function in pl/sql */
/* Program - findMax value*/
DECLARE
   a int;
   b int;
   c number;
FUNCTION findMax(x int, y int)
RETURN int
IS
    z int;
BEGIN
   IF x > y THEN
      z:= x;
   ELSE
      Z:= y;
   END IF;
   RETURN z;
END;

BEGIN
    a := &a;
    b := &b;

    c := findMax(a, b);
   dbms_output.put_line(' Maximum of (' || a ||',' ||b || '): ' || c);
END;


/* Sorting algorithm...! */
DECLARE
    type intArray IS VARRAY(10) OF INTEGER;
    arr1 intArray;
    i int;
    j int;
    c int;
    temp INTEGER := 0;
FUNCTION sortArray(arr IN OUT intArray, len integer)
RETURN int
IS
    ret int;
BEGIN
    ret := 1;
    dbms_output.new_line;
    for i in 1 .. len loop
        dbms_output.put_line('arr[' || i ||'] =' ||arr(i));
    end loop;

    for i in 1 .. len loop
        for j in 1+i .. len loop
            if (arr(i) > arr(j)) then
                temp := arr(i);
                arr(i) := arr(j);
                arr(j) := temp;
            end if;
        end loop;
    end loop;

    for i in 1 .. len loop
        dbms_output.put_line('arr[' || i ||'] =' ||arr(i));
    end loop;

    return ret;
END;

BEGIN
    arr1 :=  intArray(1,5,2,3,6,7,4,8,9,10);
    c := sortArray(arr1, 10);
END;

/* SQL Operations */
/* ROLLNO NAME MARK1 MARK2 MARK3
---------- ---------- --------- ---------- ----------
1 aparna 80 90 78
2 amritha 90 92 81
3 binuja 23 18 20
4 cathy 49 50 50
5 danish 60 62 61
6  fayas 76 62 74 */
create table Stud(rollno int primary key,
                  name char(10),
                  mark1 float,
                  mark2 float,
                  mark3 float);
insert into Stud values(1,'aparna', 80, 90, 78);
insert into Stud values(2,'amritha', 90, 92, 81);
insert into Stud values(3,'binuja', 23, 18, 20);
insert into Stud values(4,'cathy', 49, 50, 50);
insert into Stud values(5,'danish', 60, 62, 61);
insert into Stud values(6,'fayas', 76, 62, 74);

select * from Stud;

/*
Write a PL/SQL program to grade the student according to the following rules
Student(name,rollno,mark1,mark2,mark3)
TOTAL MARKS GRADE
    >=250 Distinction
    180-250 First Class
    120-179 Second Class
    80-119 Third Class
    <80 Fail
The result should be in the following Format
    STUDENT NAME:
```

```
    ROLL NO :
    TOTAL MARKS :
    GRADE :
*/
DECLARE
    roleNo integer;
    No integer;
    Name varchar(20);
    total integer;
BEGIN
    -- get role no from user
    roleNo := &roleNo;

    --execute sql query
    select rollno,name,(mark1+mark2+mark3) into No, Name, total from Stud where rollno=roleNo;

    --print output
    dbms_output.put_line('---------------------------');
    dbms_output.put_line('Roll no : ' || No);
    dbms_output.put_line('Name : ' || Name);
    dbms_output.put_line('Total Marks : ' || total);
    dbms_output.put_line('---------------------------');

    --calculate & print grade
    if (total >=250 ) then
        dbms_output.put_line('Grade  = Distinction');
    elsif (total < 250 and  total >= 180) then
        dbms_output.put_line('Grade  = First Class');
    elsif (total < 180 and  total >= 120) then
        dbms_output.put_line('Grade  = Second Class');
    elsif (total < 120 and  total >= 80) then
        dbms_output.put_line('Grade  = Third Class');
    else
        dbms_output.put_line('Grade  = FAIL');
    end if;
END;
/
```