=> Control Statements :-

-> Control Statements are those who can control the flow of the program

-> Types of Control Statements :-

1. Selection Statements:
   if, if else, if-else if, nested if, switch
2. Iteration/Looping Statements:
   for, while, do-while, for-each loop
3. Jump Statements:
   break, continue, return

=> Selection Statements :-

1. if :-

Syntax :

```
if(condition)        //condition -> boolean
{
    statements;
}
```

2. if else :-

Syntax :

```
if(condition)
{
    statements;
}
else
{
    statements
}
```

3. If else-if (ladder if):-

Syntax :

```
if(condition)
{
      statements;
}
else if(condition)
{
      statements;
}
else if(condition)
{
      statements;
}
-
-
-
else
{
     statements;
}
```

4. Nested if :-

Syntax :-

```
if(condition)
{
      if(condition)
      {
            if(condition)
            {
            }
```

```
        }
    }
    else
    {
        if(condition)
        {

        }
    }
```

5. switch :-

Syntax :

```
 switch(variable to be tested)
{
    case value1 : Statements;
                  break;
    case value2 : Statements;
                  break;
    case value3 : Statements
                  break;
    -
    -
    -
   default : Statements;
             break;
}
```

NOTE : variable to be tested must be byte, short, int, char and String but not long, float, double, boolean

```
public class VariableDemo {
    public static void main(String[] args) {
        int x = 9;
        switch (x%7){
            case 1: System.out.println("Monday");
                break;
            case 2: System.out.println("Tuesday");
                break;
            case 3: System.out.println("Wednesday");
                break ;
            case 4: System.out.println("Thursday");
                break;
            case 5: System.out.println("Friday");
                break;
            case 6: System.out.println("Saturday");
                break ;
            default:
                System.out.println("Sunday");
        }

    }
}
```

```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;
        String str = "monday" ;

        switch (str){
            case "monday" : System.out.println("day 1");
                break ;
            case "tuesday" : System.out.println("day 2");
                break ;
            case "wednesday" : System.out.println("day
1");

                break ;
            case "thrusday" : System.out.println("day 2");
                break ;
            default:
                System.out.println("Wrong input");
        }
    }
```

Programs :-

1. WAP to find the greater number between 2 numbers
2. WAP to find the greater number between 3 numbers
3. WAP to check for leap year
4. WAP to create simple calculator using switch statement

5. WAP to find the greater number between 2 numbers using ternary operator
6. WAP to swap the 2 numbers
7. WAP to swap the 2 numbers without using third variable

1.
```
public class VariableDemo {
    public static void main(String[] args) {

        int n1 = 10 ;
        int n2 = 20 ;
        if(n1>n2){
            System.out.println("n1 is greater");
        }
        else{
            System.out.println("n2 is greater");
        }

    }
```

2.
```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;
        int n3 = 30 ;

        if(n1>n2 && n1>n3){
            System.out.println("n1 is Greater");
        }
        else if(n2>n1 && n2>n3){
            System.out.println("n2 is greater");
        }
        else {
            System.out.println("n3 is greater");
        }
    }
}
```

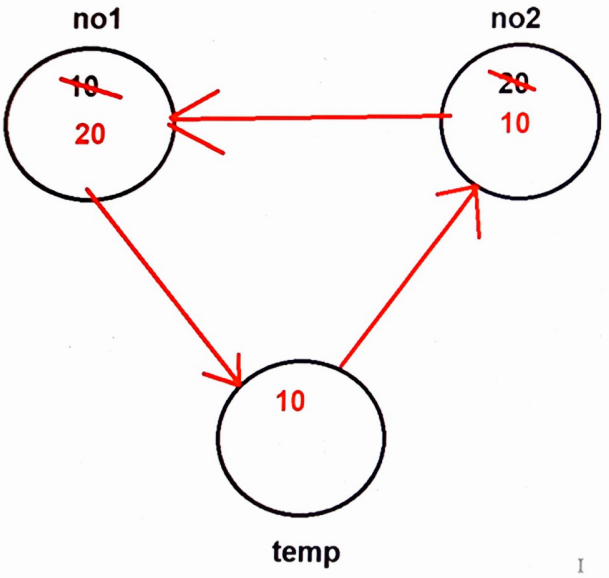
3.
```
public class VariableDemo {
    public static void main(String[] args) {
        int year = 1920 ;
        if(year%4==0){
            System.out.println("Leap year");
        }
        else{
            System.out.println("Not a leap year");
        }
    }
}
```

4.
```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;
        char c = '*' ;

        switch (c){
            case '+' :
                System.out.println(n1+n2);
                break ;
            case '-' :
                System.out.println(n1-n2);
                break;
            case '*' :
                System.out.println(n1*n2);
                break ;
            case '/' :
                System.out.println(n1/n2);
                break;
            default:
                System.out.println("Wrong input");
        }
    }
}
```

5.
```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;
        int res = (n1>n2) ? n1:n2 ;
        System.out.println(res + " is
greater");
    }
```

6.




```
int no1=10, no2=20;
int temp=no1;
no1=no2;
no2=temp;
```

```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;

        int temp = n1 ;
        n1 = n2 ;
        n2 = temp ;

        System.out.println("n1 : " +  n1);
        System.out.println("n2 : " + n2);

    }
}
```

7.

```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;

        n1 = n1+n2 ;  //30
        n2 = n1-n2 ;  //10
        n1 = n1-n2 ;  //20

        System.out.println("n1 : " +  n1);
        System.out.println("n2 : " + n2);

    }
}
```

=> More concepts for if part

-> Case 1 : If there is only single statement in if or else then its not compulsory to use curly braces {}

```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;

        if(n1>n2)
            System.out.println("n1 is greater");
        else
            System.out.println("n2 is greater");
    }
}
```

-> Case 2 : If there is no statement in if or else part then we have to use either curly braces - {} or semi-colon - ;

```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;

        if(n1>n2) ;
    }
}
```

OR

```
public class VariableDemo {
    public static void main(String[] args) {
        int n1 = 10 ;
        int n2 = 20 ;

        if(n1>n2){

        }
    }
}
```

=> Iteration/Looping Statements :-

=> for :

Syntax :

```
for(expression1; expression2; expression3)
{
        statements;
}
```

where:

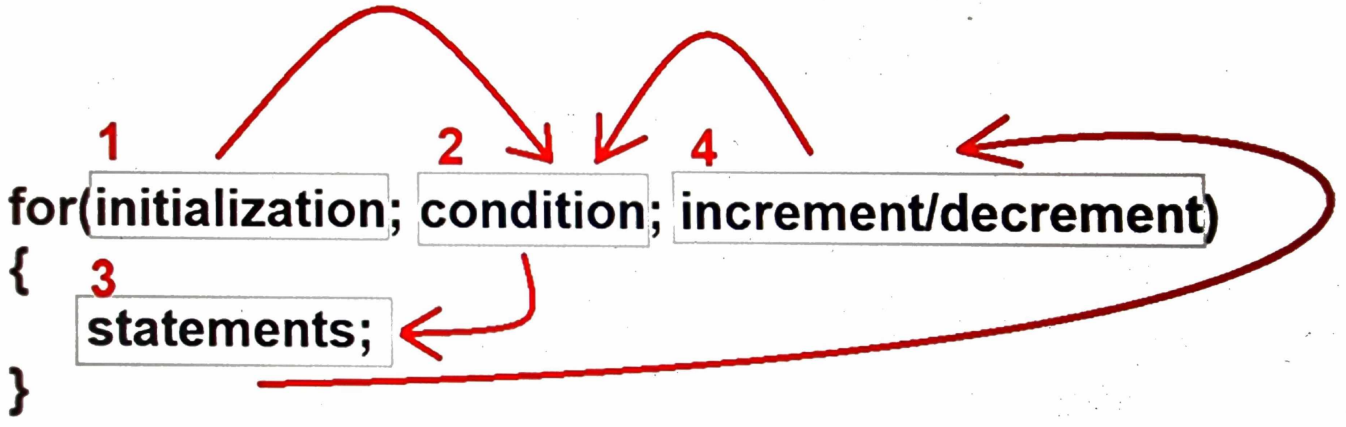
-> expression1 should be any java valid statement but prefered is declaration and initialization

-> expression2 should always be conditional statement which should return boolean value

-> expression3 should be any java valid statement but prefered is increment or decrement

simple form of for loop :

```
for(initialization; condition; increment/decrement)
{
      statements;
}
```



Programs :-

1. WAP to print any table
2. What are unrechable statements ?

1.
```
public class Table {
    public static void main(String[] args) {
        int n = 6 ;
        for(int i=1 ; i<=10 ; i++){
            System.out.println(n + " * " + i + " = " + n*i);
        }
    }
}
```

Case 1:

```
public class Demo {
    public static void main(String[] args) {
        for(int i=1, j=1 ; i<=5 && j<=5 ; i++, j++){
            System.out.println("i: " + i);
            System.out.println("j: " + j);
        }
    }
}
```

case 2:

```
public class Demo {
    public static void main(String[] args) {
        int i=1 ;
        for( ; i<=5 ; i++){
            System.out.println("i: " + i);
        }
    }
}
```

case 3:

```
public class Demo {
    public static void main(String[] args) {
        int i=1 ;
        for(System.out.println("hi"); i<=5 ; i++){
            System.out.println("i: " + i);
        }
    }
}
```

==========

2.

```
public class Demo {
    public static void main(String[] args) {
        for(int i=1; true ; i++){
            System.out.println("i: " + i);
        }
        System.out.println("hi");      // unreachable statement
    }
}
```

=> while :-

Syntax :

```
while(condition)
 {
    statements;
 }
```

When we should use while loop : When we dont know how many time we have to execute the loop then we have to use while loop

Cases :

1. Its compulsory to provide the condition in while loop
2. If while loop is infinite and we provide any expression after while loop then it will provide compile time error saying "Unreachable statement"

=> do-while :-

Syntax :

```
do
{
        //statements
}while(condition);
```

Programs :

1. WAP to get the number of digits in integer value
(163 - 3; 5673 - 4)
2. WAP to reverse a number
(163 -> 361; 5673 -> 3765)
3. WAP to create calculator using do-while loop

=> Interview Question :-

```
1. public class Demo {
     public static void main(String[] args) {
        // finding no of digit in a number
        int n = 12345 ;
        int count=0 ;
        while(n!=0){
            n=n/10;
            count = count+1 ;
        }
        System.out.println(count);
     }
  }
```

==========

```
2. public class Demo {
    public static void main(String[] args) {
        // reverse no of digit in a number
        int n = 12345 ;
        int ans = 0 ;
        while(n>0){
            ans = ans*10 + n%10 ;
            n = n/10 ;
        }
        System.out.println(ans);
    }
}
```

==========

```
import java.util.Scanner;

public class Demo {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in) ;
        String respone = "";
        do{
            System.out.println("Enter the 1st number: ");
            int n1 = sc.nextInt() ;
            System.out.println("Enter the 2nd number: ");
            int n2 = sc.nextInt() ;
            System.out.println("Enter the Symbol +, -, *, / ");
            String symbol = sc.next();

            switch (symbol){
                case "+" :
                    System.out.println(n1+n2);
                    break ;
                case "-" :
                    System.out.println(n1-n2);
                    break;
                case "*" :
                    System.out.println(n1*n2);
                    break ;
                case "/" :
                    System.out.println(n1/n2);
                    break;
                default:
                    System.out.println("Wrong input");
            }
            System.out.println("Do you want to continue, Press y for yes and n for No");
            respone = sc.next() ;
        }
        while(respone.equals("y") || respone.equals("Y")) ;
    }
}
```