Notes

Looping in Java

Looping is a sequence of statements that is continually repeated until a certain condition is reached

Or

Looping is defined as a repetitive structure in which a statement or a set of statements are repeated until the given condition is false.

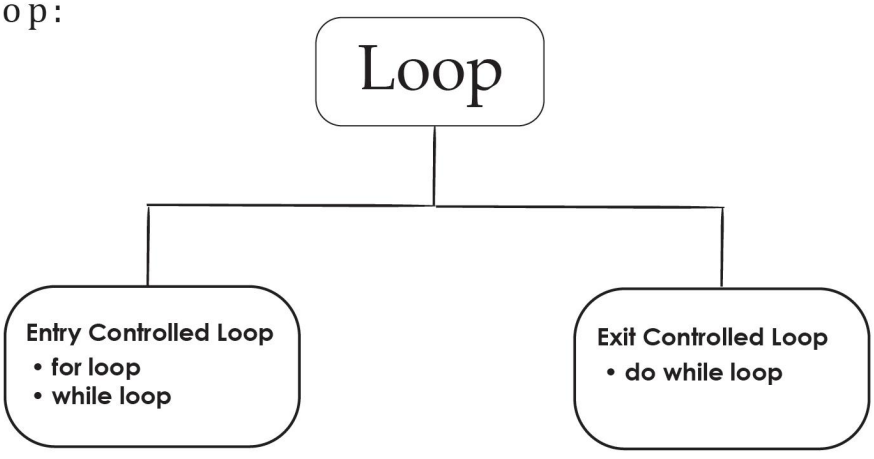
Or

The process of executing a statement or a set of statements a finite number of time.

There are 3 types of loops which we use in Java:

- for loop
- while loop
- do while loop

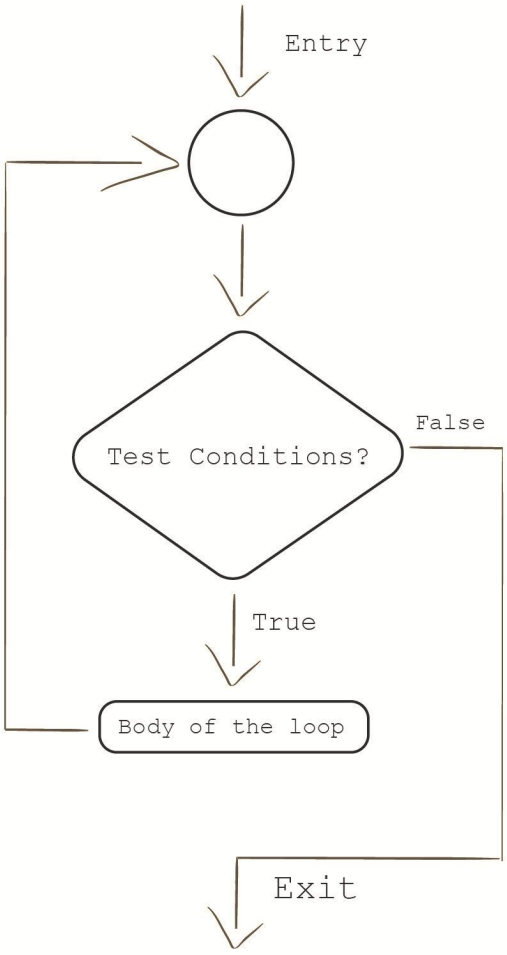
These loops are categorised into Entry Controlled loop and Exit Controlled loop:



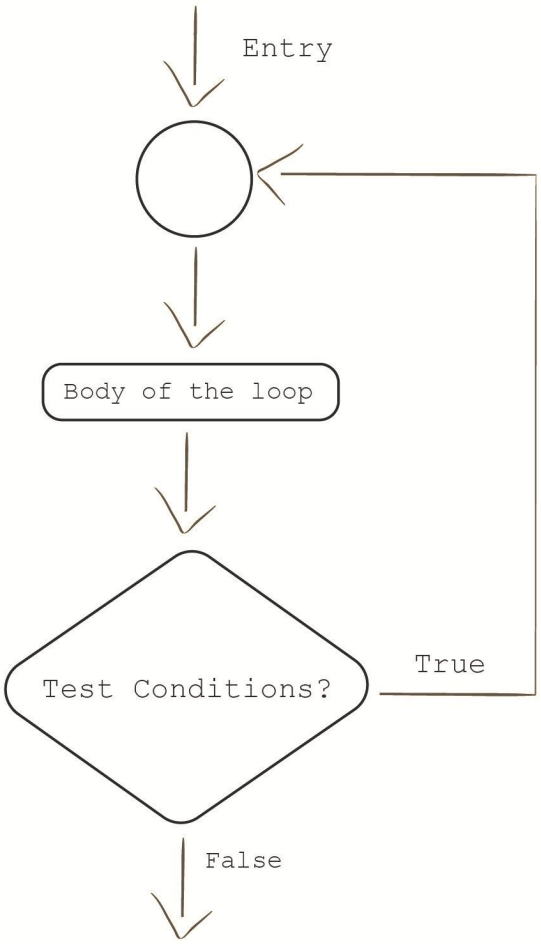
Types of loop:

1. **Entry controlled loop** - Entry controlled loop is a loop in which the test condition is checked first, and then the loop body will be executed

Eg. When you enter a PVR first your ticket is checked and then you are allowed to enter.



These include "For Loop" and "While Loop"



2. **Exit Controlled Loop** - Exit controlled loop is a loop in which the condition is checked after the execution of the body of the loop.

Eg. When you enter a Bus first you are allowed to be seated and then your ticket is checked.

These include "Do while loop"

NOTE: What is a statement? *Java statements are instructions that tell the programming language what to do, like declaration and string statements. Eg – int I = 1, Math.pow(2,1), System.out.print("java"), etc.*

For Loop (Entry Controllled Loop) - A "For" Loop is used to repeat a specific block of code a known number of times.

The loops starts with the given value of a variable. it executes the given statement by updating the values of control variable unless the given condition is false. This looping construct is used only when the user is aware as how many times the process is to be operated

Syntax: for(initialisation; testing; updating){

Statement(s); ← **3**

}

NOTE: What is control variable? *The control variable is initialized, tested, and changed as the loop executes. It is an ordinary int (integer) variable, but it is used in a special way. The way is that of a loop control variable*

For Loop Execution hierarchy-

1. Initialisation - The initialization is a statement that initializes the loop, it is executed only once when the loop starts. Next it goes to Testing(2)

Eg - When a baby is born, his life has been initialised.

2. Testing - It is just like an if statement, it checks if the control variable meets the conditions specified in the testing part. Next it goes to body (3)

Eg - Suppose the value of the control variable is 10 and the condition you wrote is i<=10. So now it checks if "i"(control variable) is smaller than or equal to 10. If condition is "TRUE" the body of the loop executes, if the condition is "FALSE" the control goes outside the loop

3. Body - The statements inside the body of loop gets executed (They are carried out)

Eg - Suppose the test condition comes out to be true, now if we have written System.out.print("Hello World) in the body of the loop then it will print Hello World.

4 . Updating - The control variable gets updated according to the action specified in the update part.

```
for(int i=1;i<=10;i++){
        System.out.println(i);
}
```

Eg. In this case the value of "i" is getting increased by 1.

Examples Programs on For loop:

1. WAP to print Hello World 10 times

```
class helloworld{
    void fg(){
        for(int i = 1; i<=10; i = i+1){
            System.out.println("Hello World");
        }
    }
}
```

Explanation:

• The control variable is "i"

• The test condition is i <= 10

• The statement given is System.out.println("Hello World") this means that it should print Hello World

• The loop begins with initialisation (int i =1), then it goes to testing (i<=10) here it is checked if "i" (Value of i = 1) is smaller than or equal to 10

• Now the loop comes to printing, it prints Hello World for the first time.

• Now the loop comes to updating the value of the control variable ("i") , in this program it is written to increment the value of i by 1.

• The value of "i" becomes 2, now the loop goes to the testing part, now if the condition is true again it loops over the statements again.

• Now the loop continues till the value of "i" reaches 10 and it gets updated and becomes 11. Now, when the value is 11, the condition becomes false and the loop breaks.

2. WAP to print a series of numbers from 1 to 10

```
class numbers{
    void fg(){
        /*Using for loop to print the numbers from
        1 to 10*/
        for(int i=1; i<=10;i++){
            System.out.print(i);/*Printing the
            counter variable"i"*/
        }
    }
}
```

Explanation:

- The control variable is “i”
- The test condition is i <= 10
- The statement given is System.out.println(i) this means that it should the value stored inside the variable “i”
- The loop begins with initialisation (int i =1), then it goes to testing (i<=10) here it is checked if “i” (Value of i = 1) is smaller than or equal to 10.
- Now the loop comes to printing, it prints the value of “i” for the first time which is “1”
- Now the loop comes to updating the value of the control variable (”i”) , in this program it is written to increment the value of i by 1. For eg - if the value of “i” is 1 it will become i = 1(value of i) + 1.
- The value of “i” becomes 2, now the loop goes to the testing part, now if the condition is true again it loops over the statements again.
- Now the loop continues till the value of “i” reaches 10 and it gets updated and becomes 11. Now, when the value is 11, the test condition becomes false and the loop breaks

Extra Knowledge about update part - *You can use either i++ or ++i or i = i+1 or i+=1 to update the value of i by 1. If you have to increase the value of “i” by 2 or greater than 2 then we either use i = i+2(or any number according to the question) or we use i+=2.*

3. WAP to print a series of even numbers from 1 to 10

```
class evennumbers{
    void fg(){
        /*Using for loop to print even numbers from
        1 to 10*/
        for(int i=1; i<=10;i++){
            //Checking if "i" is a even number or not
            if(i%2 == 0){
                System.out.print(i);/*Printing the
            counter variable"i"*/
            }
        }
    }
}
```

Explanation:

- The control variable is “i”
- The test condition is i <= 10
- The statement given is System.out.println(i) this means that it should the value stored inside the variable “i”
- The loop begins with initialisation (int i =1), then it goes to testing (i<=10) here it is checked if “i” (Value of i = 1) is smaller than or equal to 10.
- Now the loop comes to the loop body, we have written an if statement with the condition - i%2 == 0 which means that it will check if the remainder after dividing “i” by 2 is 0 or not. If it is 0 then it prints “i” other wise it moves on to the update part
- Now the loop comes to updating the value of the control variable (”i”) , in this program it is written to increment the value of i by 1. For eg - if the value of “i” is 1 it will become i = 1(value of i) + 1.
- The value of “i” becomes 2, now the loop goes to the testing part, now if the condition is true again it loops over the statements again.
- Now the loop continues till the value of “i” reaches 10 and it gets updated and becomes 11. Now, when the value is 11, the test condition becomes false and the loop breaks

Testing your ability - *Find another logic to print the even digits.*
HINT: DONT USE IF STATEMENT, MAKE CHANGES TO THE LOOP

Practice Program: *WAP to print a series of odd numbers from 1 to 10. Use both the logics to solve this program*

PRACTICE PROGRAMS:

1. Write a program to print first 10 even numbers
2. Write a program to print first 15 odd numbers

(There is a difference in printing number from 1 to 10 and printing first 10 numbers)

3. Write a program to print squares of numbers from 1 to 20
(HINT: Assuming that the control variable is "i". Put i*i or Math.pow(i,2) inside the loop)

4. Write a program to print cubes of numbers from 1 to 10
5. Write a program to print squares and cubes of numbers from 1 to 10

Sequence and Series:

1. Sequence: A list of numbers of objects in a special order

Example: 3,5,7,9 - (Odd numbers)

2. Series: The cumulative sum of values in a sequence is called a series. We have both finite and infinite series.

Example: 3 + 5 + 7 + 9 +.... is a series

Now we will be studying about how do we conquer series and sequences using loops.

Steps of analysing and making a logic for a series or sequence-

STEP 1- Try to find a pattern in the sequence/series

STEP 2- Check if the pattern applies to any scenario.

STEP 3- Think how can you write that logic in program.

STEP 4- Check if the logic you want to write is a universal pattern set for the given series/sequence.

STEP 5- Write the program.

Can you find a pattern in nature?
Write about that pattern -

COMING SOON.....