5CS019 - Object Oriented programming and design (OODP)

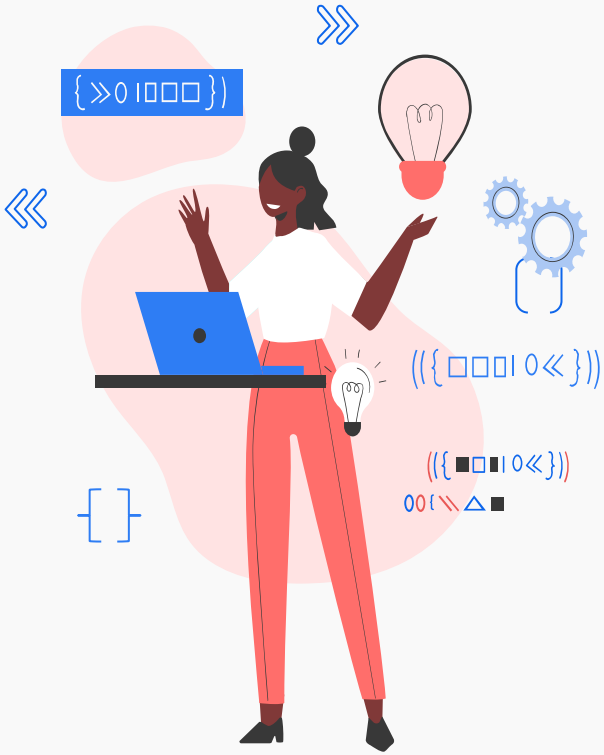
Looping and Arrays

Tutorial 03

Herald College, University of Wolverhampton

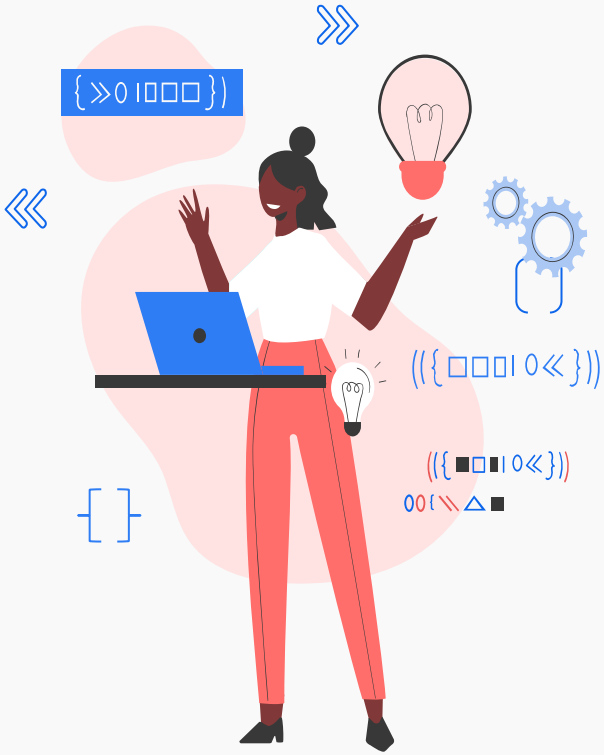
Recap..

- Operators in Java
  - Arithmetic
  - Assignment
  - Logical
  - Comparison
  - Bitwise
- Strings in Java
- Conditional Statements
  - simple if
  - if-else
  - if-else-if
  - nested if
- Type Casting in Java



Topics Covered Today

- Looping in JAVA
  - FOR loop
  - While Loop
  - DO-WHILE loop
- Arrays in Java
- Enhanced FOR loop
- Looping and Arrays combined



Loop in Java

In computer programming, loops are used to repeat a block of code.

Types of Loop in Java

- **For loop**
- **While Loop**
- **Do-while loop**

for loop

The Java for loop is used to iterate a part of the program several times. If the number of iteration is fixed, it is recommended to use for loop.

while loop

The Java while loop is used to iterate a part of the program several times. If the number of iteration is not fixed, it is recommended to use while loop.

do-while loop

The Java do-while loop is used to iterate a part of the program several times. Use it if the number of iteration is not fixed and you must have to execute the loop at least once.

For Loop in Java

It is usually used when the number of iteration is fixed.

It consists of four parts

- Initialization
- Condition
- Increment/Decrement
- Statement

Jump statement

It is a loop control statement used to terminate or skip the loop. The types of jump statements are **break** and **continue**.

```
while (expr):
    <statement>
    <statement>
break
    <statement>
    <statement>
```

```
while (expr):
    <statement>
    <statement>
continue
    <statement>
    <statement>
```

Break in Java

```
while (testExpression) {
    // codes
    if (condition to break) {
        break;
    }
    // codes
}
```

```
do {
    // codes
    if (condition to break) {
        break;
    }
    // codes
}
while (testExpression);
```

```
for (init; testExpression; update) {
    // codes
    if (condition to break) {
        break;
    }
    // codes
}
```

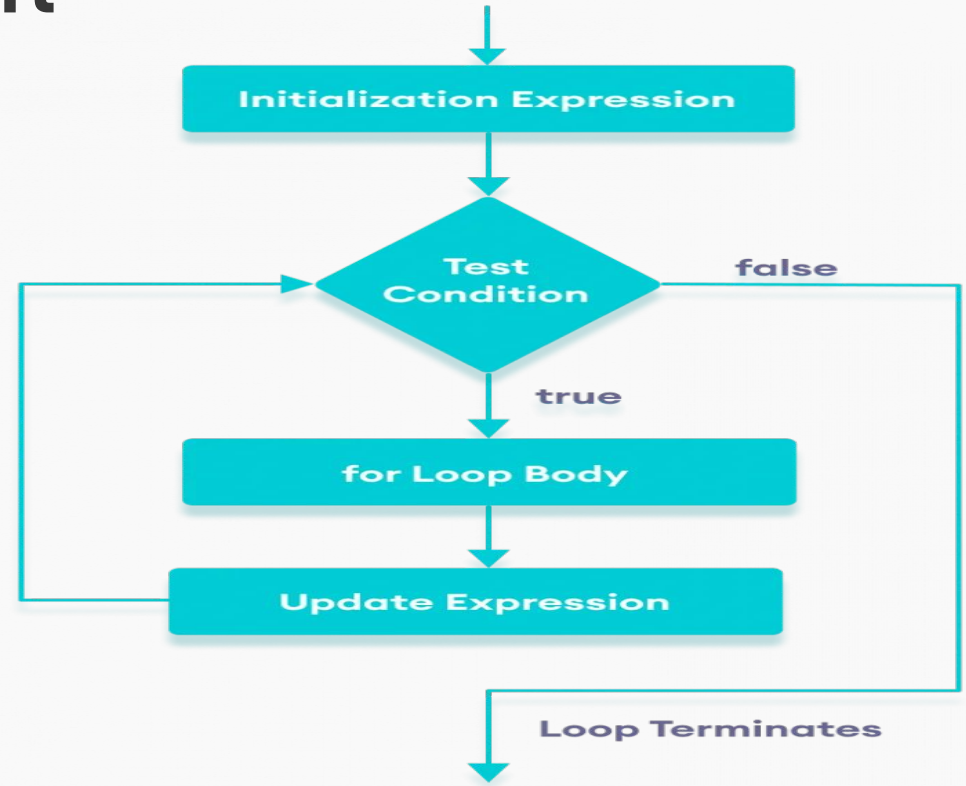
Continue in Java

```java
while (testExpression) {
    // codes
    if (testExpression) {
        continue;
    }
    // codes
}
```

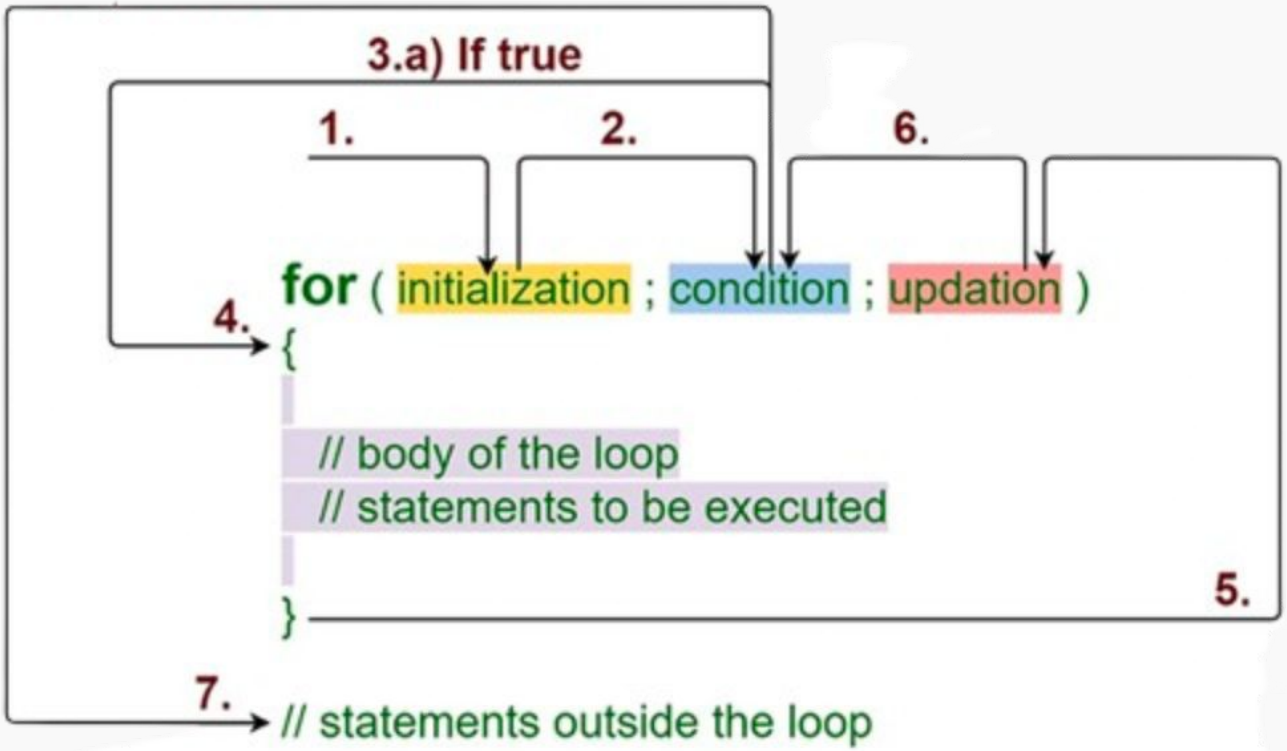
```java
do {
    // codes
    if (testExpression) {
        continue;
    }
    // codes
}
while (testExpression);
```

```java
for (init; testExpression; update) {
    // codes
    if (testExpression) {
        continue;
    }
    // codes
}
```

Flow chart


Initialization Expression
Test Condition
false
true
for Loop Body
Update Expression
Loop Terminates

Syntax


3.a) If true
1.
2.
6.
for (initialization ; condition ; updation)
4.
{
// body of the loop
// statements to be executed
}
5.
7.
// statements outside the loop

Task 1

Write a Java program to Display a Text, “Herald College Kathmandu” Five Times.

Task 2

Write a Java program to Display Sum of n even Natural Numbers

For n = 5, sum should be 6 (2+4).

Infinite Loop

If the test expression never evaluates to false, then the loop will run forever. This is called infinite loop.

Example of infinite loop:

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

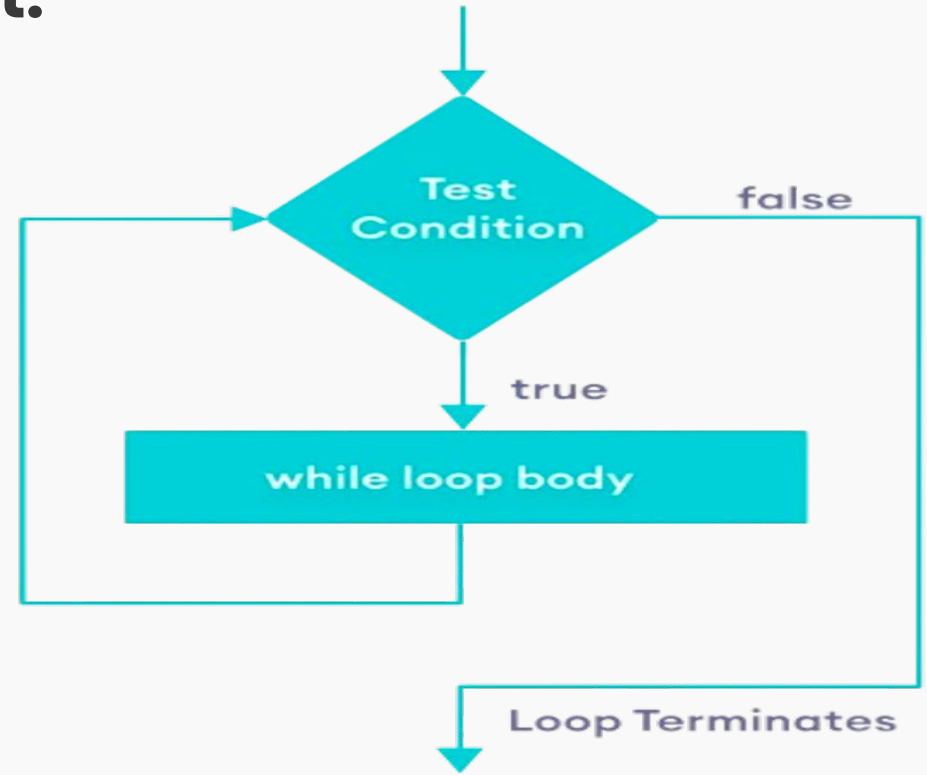
While Loop in Java

It is usually used when the number of iteration is not fixed.

While loop syntax:

```
while (testExpression) {
  // body of loop
}
```

Flowchart:



Do while Loop in Java

Do..while loop is quite similar to while loop. But the body is executed **at least once**.

It is usually used when the number of iteration is not fixed.

Do while loop syntax:

```
do{
  // body of loop
} while (testExpression)
```

Task 4

Write a Java program to display multiplication of a number from 1 to 10. (ask user to input the number).

Task 5

Write a program to print the following pattern in Java.

```
*
* *
* * *
* * * *
* * * * *
```

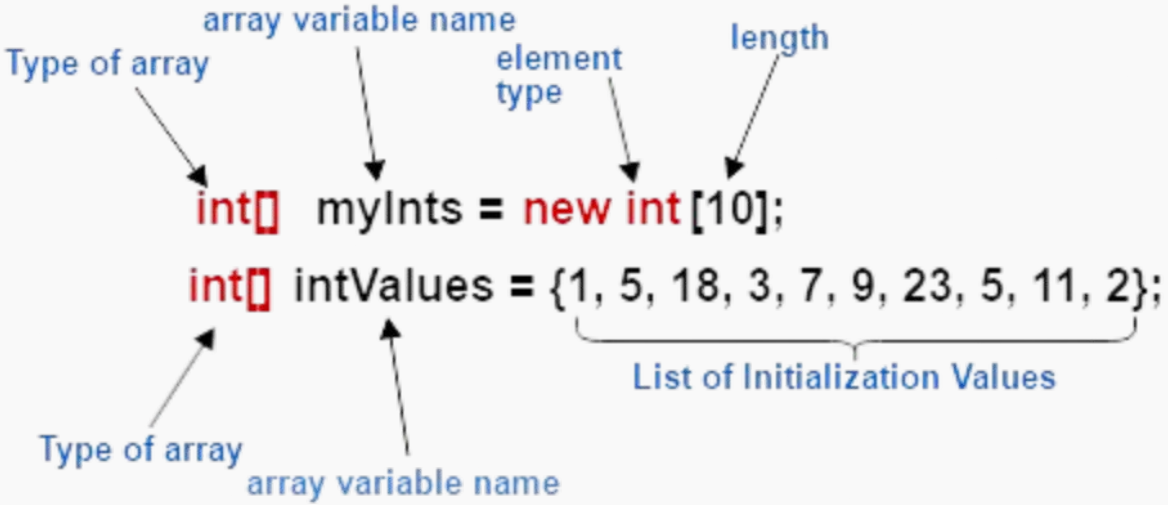
Task 6

Write a program to print the following pattern in Java. (Use for loop)

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

Java Array

An array is a collection of similar types of data.
Java array is an object which contains elements of a similar data type.



Initializing an Array

1. int[] arr = {1,2,3,4,5};

2. int[] arr;
arr = new int[10];

or, int[] arr = new int[10];

Types of Array

Single dimensional array:

This is the most basic type of array where elements are stored in a linear sequence.

Example:

Initializing an array:

```
int[] integerArray= new int[3];
```

Declaring a one dimensional array

```
int[] integerArray = {1, 2, 3}
```

Multi-dimensional array:

This is the most basic type of array where elements are stored in a multiple dimensions, like row and column..

Example:

Initializing a multi dimensional array:

```
int[][] integerArray= new int[3][3];
```

Declaring a multidimensional array

```
int[][] integerArray = {{1,2,3}, {4,5,6}}
```

Task 7

Write a Java program to initialize a one dimensional integer array **integerArray** that can store five integers.

Task 8

Write a java program to initialize a 2D array with the given values:

5	12	17	9	3
13	4	8	14	1
9	6	3	7	21

Iterating through array elements

Using For-loop:

```
class Tutorial2 {

    Run | Debug
    public static void main(String[] args) {

        int[] myArray = { 1, 2, 3, 4, 5 };

        for (int i = 0; i < myArray.length; i++) {
            System.out.print(myArray[i] + " ");
        }
    }
}
```

```
[Running] cd "/Users/sajagsilwal
1 2 3 4 5
```

Iterating through array elements

Using For-each loop:

```
class Tutorial2 {

    Run | Debug
    public static void main(String[] args) {

        int[] myArray = { 1, 2, 3, 4, 5 };

        for (int i : myArray) {
            System.out.print(i + " ");
        }
    }
}
```

```
[Running] cd "/Users/sajagsilwal
1 2 3 4 5
```

Task 9

Iterate and print the elements of the array declared in Task 7 and 8.

Task 10

Implement Linear search to find a user input integer in an array of integer. Print "**Matched!**" if the string has been found, else print "**Not matched!**"


Any
Questions?