Java Introduction and Basics

ThoughtWorks®

ThoughtWorks®

GLOBAL SOFTWARE CONSULTANCY

Java Introduction

What is Java?

A programming language and a platform

Java is a **programming language** and a **platform**. Java is a high level, robust, object-oriented and secure programming language.

Platform: Any hardware or software environment in which a program runs, is known as a platform. Since Java has a runtime environment (JRE) and API, it is called a platform.

A brief History of Java

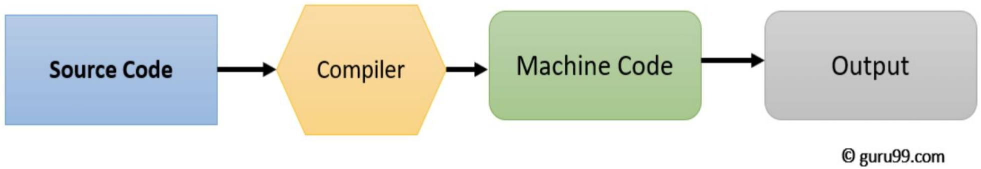
- James Gosling, Mike Sheridan, and Patrick Naughton initiated the Java language project in June 1991
- Originally designed for interactive television, but it was too advanced for digital cable of that time
- Java was developed by Sun Microsystems (which is now the subsidiary of Oracle) in the year 1995. Before Java, its name was Oak. Since Oak was already a registered company, so James Gosling and his team changed the name from Oak to Java.
- First public implementation was Java 1.0 in 1996, that was released by Sun Microsystems (acquired by Oracle in 2010)

Compile & Interpret in java world

What Is Compiler?

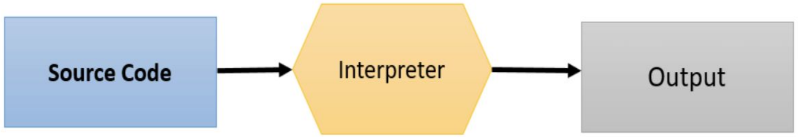
What is Interpreter?

How Compiler Works



© guru99.com

How Interpreter Works



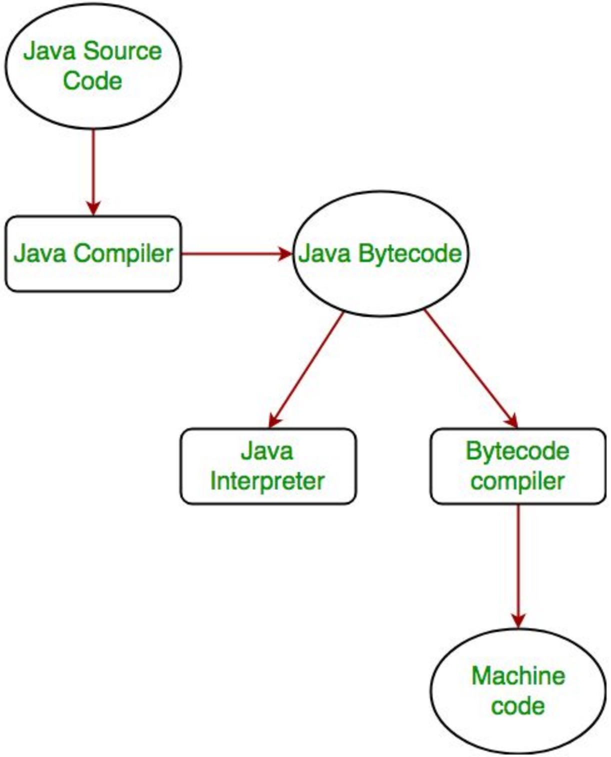
Compiler & Interpreter

Compiler	**Interpreter**
Compiler transforms code written in a high-level programming language into the machine code, at once, before program runs	Interpreter converts each high-level program statement, one by one, into the machine code, during program run
Compiled code runs faster	Interpreted code runs slower
Compiler displays all errors after compilation	Interpreter displays errors of each line one by one

OS Agnostic

The meaning of platform-independent is that the java compiled code(byte code) can run on all operating systems.

How Java Program is executed step by step



Java LTS Version

People often download and install the latest version of Java, rather than the latest LTS version of java. In most cases, especially if it is on a server you probably want to be using the LTS version of java.

So what is a Java LTS Version?

Long Term Support version

What is the latest Java LTS Version?

Java 17 September 2021

How many years is a Java LTS version supported for?

5 years

Basics of Java

Topic - Fundamental

- New project in Intellij
- Create a class
- Main method
- System.out.println
- Write a simple java program to print “Yo! Im gona totally rock!!”

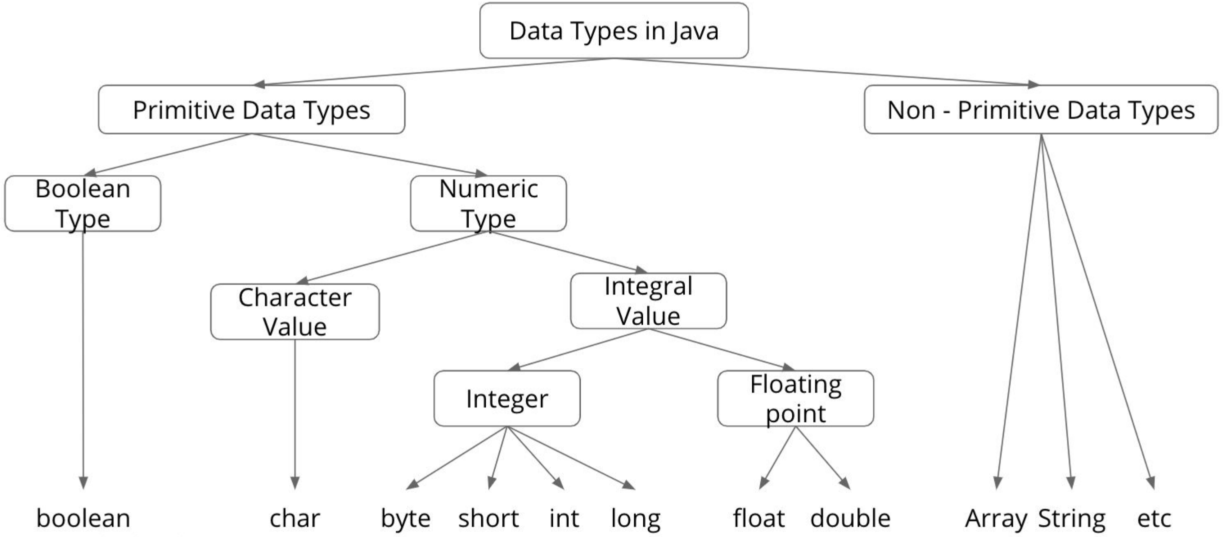
Topic - Fundamental

- Data Types - Data types specify the different sizes and values that can be stored in the variable. There are two types of data types in Java:
  - **Primitive data types:** The primitive data types include boolean, char, byte, short, int, long, float and double. These are the most basic data types available in Java language.
  - **Non-primitive data types:** The non-primitive data types include Classes, Interfaces, and Arrays.

Data Type	Default Value	Default size
boolean	false	1 bit
char	'\u0000'	2 byte
byte	0	1 byte
short	0	2 byte
int	0	4 byte
long	0L	8 byte
float	0.0f	4 byte
double	0.0d	8 byte

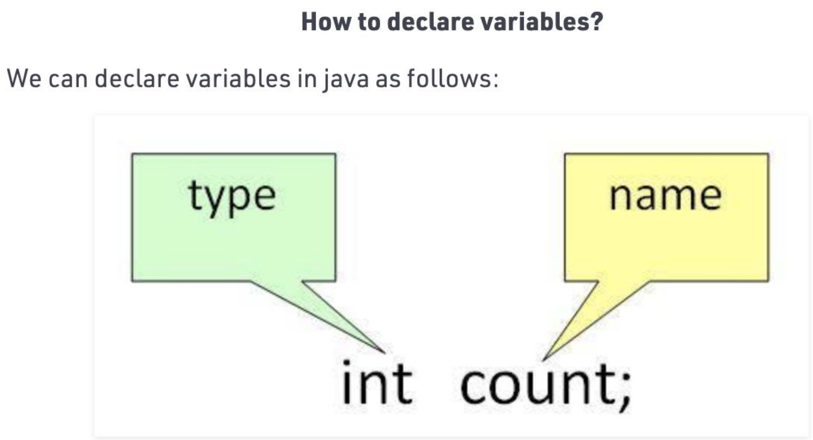
Data Types in Java

Data Types in Java



Variables

A variable is a name given to a memory location. It is the basic unit of storage in a program.



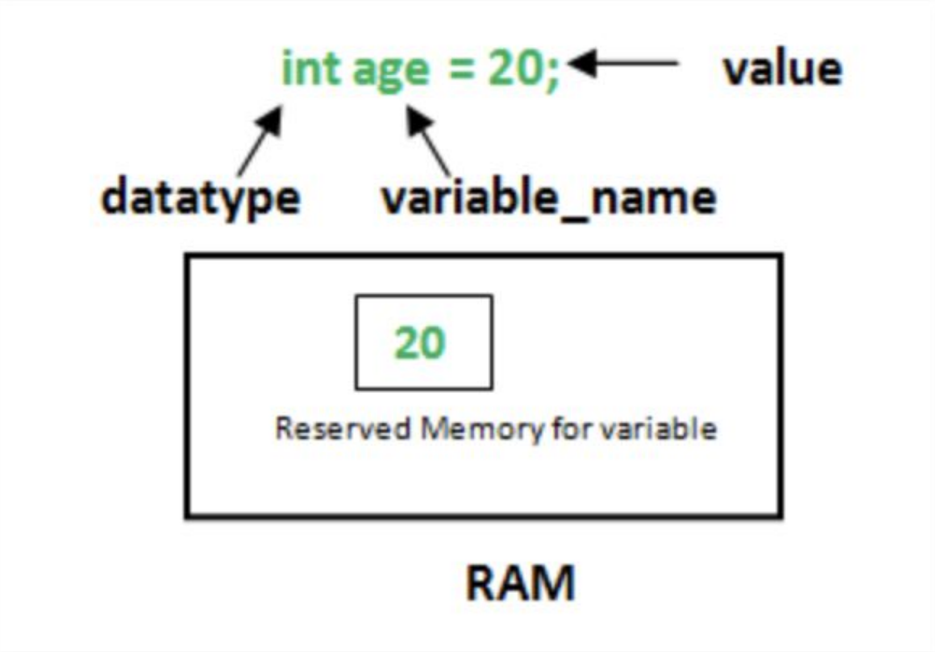
How to declare variables?

We can declare variables in java as follows:

```
int  count;
```

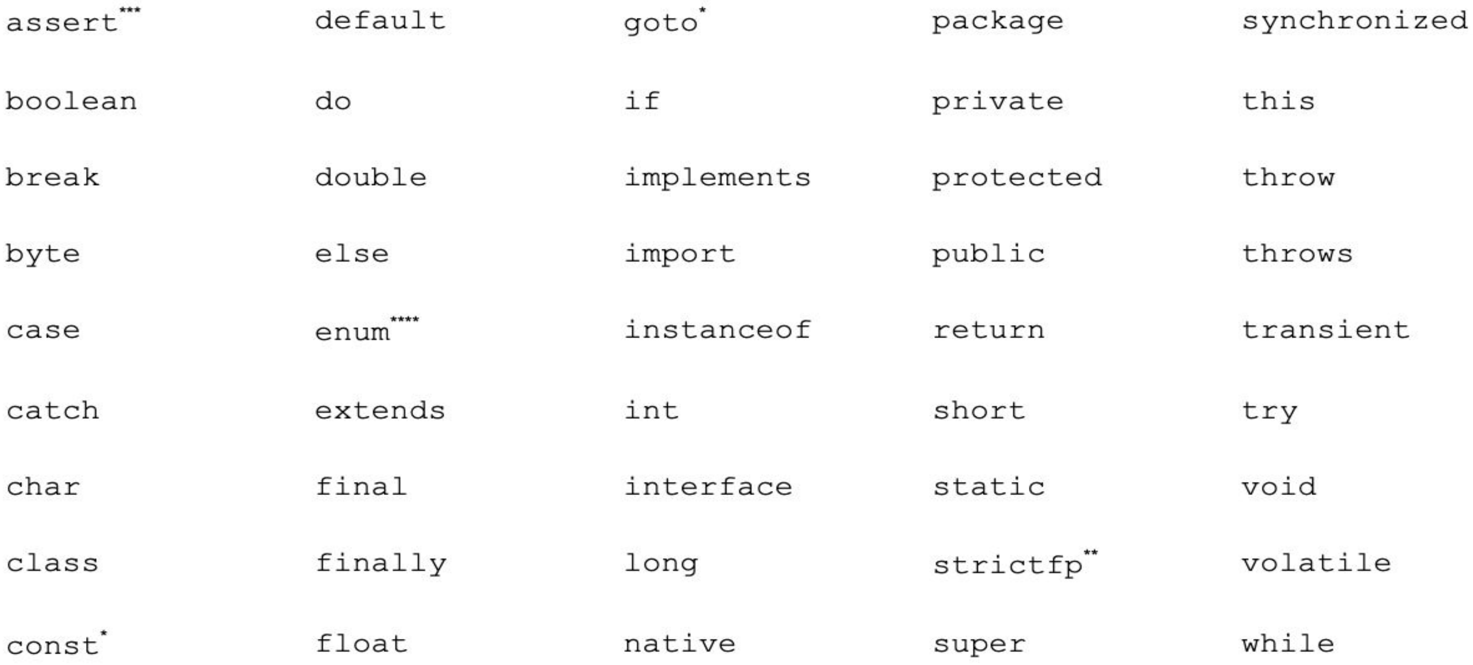
Variables contd..

How to initialize variables?



Java Keywords

Java keywords are also known as reserved words. Keywords are particular words that act as a key to a code. These are predefined words by Java so they cannot be used as a variable or object name or class name.



assert***
boolean
break
byte
case
catch
char
class
const*

default
do
double
else
enum****
extends
final
finally
float

goto*
if
implements
import
instanceof
int
interface
long
native

package
private
protected
public
return
short
static
strictfp**
super

synchronized
this
throw
throws
transient
try
void
volatile
while

Comments

1. Single Line Comment
2. Multiline Comment

Example:

```
// This is a comment

System.out.println("Hello World");

System.out.println("Hello World"); // This is a comment

/* The code below will print the words Hello World
to the screen, and it is amazing */

System.out.println("Hello World");
```

Break Out Activity

Write a simple program to sum 2 decimal point numbers and print output.

E.g. - Input = 2, 5

Output = 7

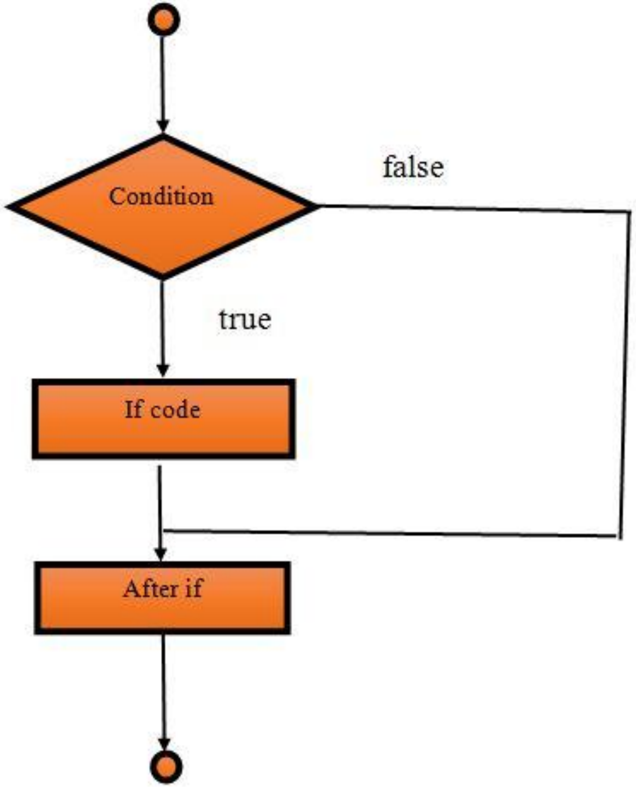
Note: use single line and multiple line comments in your program

Topics - Java Conditions

- If, if-else, else-if, nested ifs
- Switch Statements
- Relational & Logical Operators`

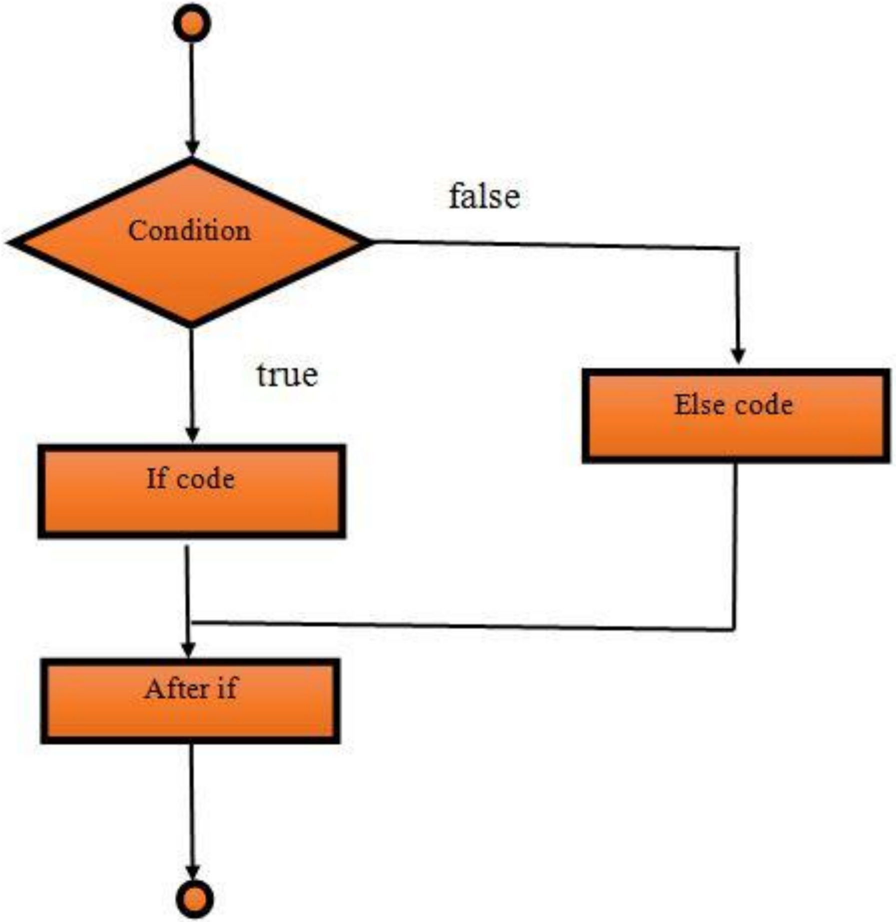
If - Statement

```
public class IfDemo1 {
public static void main(String[] args)
        {
        int marks=70;
        if(marks > 65)
                {
                System.out.print("First division");
                }
        }
}
```



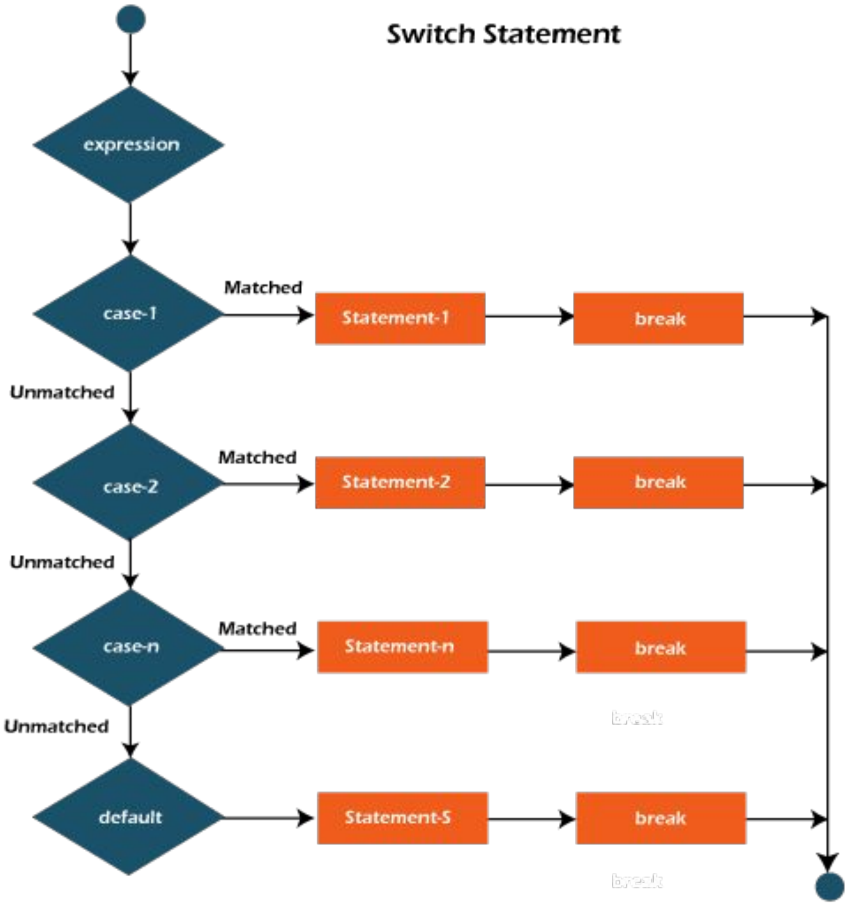
If else- Statement

```
public class IfElseDemo1 {
        public static void main(String[] args)
        {
                int marks=50;
                if(marks > 65)
                {
                        System.out.print("First division");
                }
                else
                {
                        System.out.print("Second division");
                }
        }
}
```



Switch Statement

```
switch (marks) {
        case 75:
           System.out.println("A Division");
           break;
        case 50:
           System.out.println("B- Division");
           break;
        case 34:
           System.out.println("Failed");
           break;
        default:
           System.out.println("Incorrect marks entered");
           break;
    }
```



Relational Operators

Operators	Meaning	Example	Result
<	Less than	5<2	False
>	Greater than	5>2	True
<=	Less than or equal to	5<=2	False
>=	Greater than or equal to	5>=2	True
==	Equal to	5==2	False
! =	Not equal to	5! =2	True
===	Equal value and same type	5 === 5	True
		5 === “5”	False
! ==	Not Equal value or Not same type	5 ! == 5	False
		5 ! == “5”	True

Logical Operator

Logical Operators	**Java Operator**	**True Condition**
AND	&&	Both condition to be true
OR	||	At Least one condition to be true
NOT	!	Condition should be false

Assignment

Write a program to take a number as an input from the user and print which day of week it is as per the input number. Handle incorrect day number scenario as well.

Note: need to expose mod operator

Topic - Loops

- For loop
- While Loop
- Do While loop
- Nested loop

Topic - Loops

Loops

Loop is an important concept of a programming that allows to iterate over the sequence of statements.

Loop is designed to execute particular code block till the specified condition is true or all the elements of a collection(array, list etc) are completely traversed.

The most common use of loop is to perform repetitive tasks. For example if we want to print table of a number then we need to write print statement 10 times. However, we can do the same with a single print statement by using loop.

Loop is designed to execute its block till the specified condition is true.

Java Loops

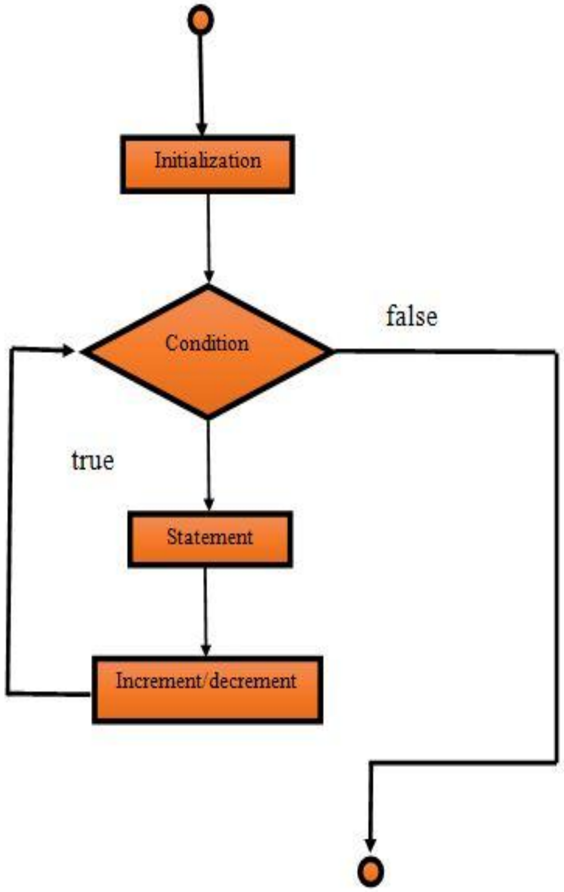
1. for
2. while
3. do while

For Loop

The for loop is used for executing a part of the program **repeatedly**. When the number of execution is fixed then it is suggested to use for loop

Data Flow Diagram of for loop

This flow diagram shows flow of the loop. Here we can understand flow of the loop.



For Example

```
public class ForDemo
{
        public static void main(String[] args)
        {
                int n=2;
                for(int i=1;i<=10;i++)
                {
                        System.out.println(n + "*" + i + "=" + n*i );
                }
         }

}
```

Break and Continue

While Example

```
public class WhileDemo {

      public static void main(String[] args) {

            int i = 0;

            int n = 2;

            while ( i == 10) {

                  System.out.println(i + "*" + n  + "=" + i * n);

            }

      }

}
```

Do While Example

```
public class DoWhile {
  public static void main(String[] args) {
     int i = 0;
     int n = 2;
     do {
        System.out.println(n + "*" + i + "=" + n*i);
        i++;
     } while (i < 10);
  }
}
```

BO

Topic - Variable Scoping & Methods

- Methods
    - Signature
    - Return type
- Variable scoping
- Overloading

Naming convention for variables and methods

	should	Shouldn’t / avoid	example

variables	It should start with a lowercase letter
	If the name contains multiple words, start it with the lowercase letter followed by an uppercase letter
		It should not start with the special characters like & (ampersand), $ (dollar), _ (underscore).
		Avoid using one-character variables such as x, y, z.
			double area;
			int index;
			String lastName;

methods	It should start with lowercase letter.
	It should be a verb such as main(), print(), println().
	If the name contains multiple words, start it with a lowercase letter followed by an uppercase letter
		It should not start with number, %, *, &, @ etc	void area() {}
			int createAccount() {}

BO

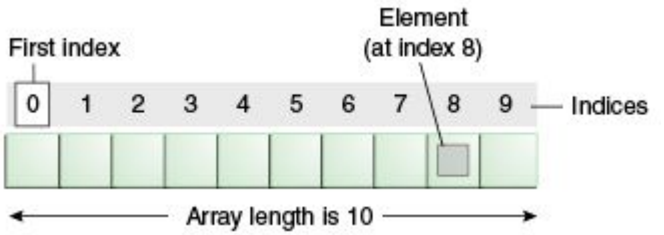
Topic - Arrays

Normally, an array is a collection of similar type of elements which has contiguous memory location.

Java array is an object which contains elements of a similar data type. Additionally, The elements of an array are stored in a contiguous memory location. It is a data structure where we store similar elements. We can store only a fixed set of elements in a Java array.

In Java, there is a class for every array type, so there's a class for int[] and similarly for float, double etc.

For every array type corresponding classes are available and these classes are the part of java language and not available to the programmer level.



- We can get the length of the array using the length member
- **Advantages**
  - Code Optimization: It makes the code optimized, we can retrieve or sort the data efficiently.
  - Random access: We can get any data located at an index position.
- **Disadvantages**
  - Size Limit: We can store only the fixed size of elements in the array. It doesn't grow its size at runtime. To solve this problem, collection framework is used in Java which grows automatically.

Topic - Arrays

Contd…

Syntax:

- dataType[] arr; (or)
- dataType []arr; (or)
- dataType arr[];

Initialization:

- int a[]=new int[5];//declaration and instantiation

  a[0]=10;//initialization

  a[1]=20;
- int a[]={33,3,4,5};//declaration, instantiation and initialization

Printing an Array

Passing Array to a Method

ArrayIndexOutOfBoundsException

Assignment

Given an array of 10 integers, write a program to print the sum of all 10 integers in the given array.

Debugging in IntelliJ

- Toggling breakpoints
- Running and understanding a program in debug mode

THANKYOU