Java Full Course

By TechDive With Gaurav

What is Programming

Just like we use Hindi, English or other language to communicate with each other, we use the programming language to communicate with the computer. A programming language is an artificial language designed to express computations that can be performed by a machine, particularly by a computer. It can be used to create programs that control the behaviour of a machine, to express algorithms precisely, or as a mode of human communication.

The programming language is the way of communication between the user and a computer. Examples of programming language are C, C++, Java, Python etc..

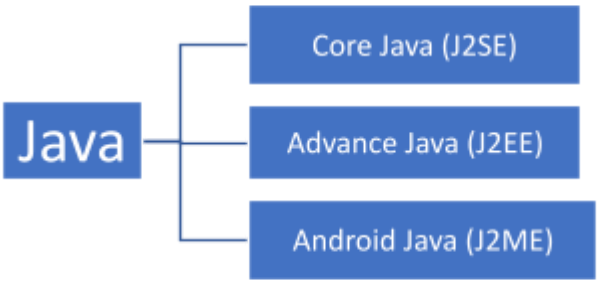
Type of Programming language

There are three types of programming language: -

1. **Low-level Programming Language**
   - Machine Language
   - Assembly Languages or Symbolic Languages
2. **Middle-level Programming Language**
   - C Programming Language
3. **High-level Programming Language**
   - Java
   - Python, etc..

Introduction of Java





Why we learn Java?

- Simple and easy to learn.
- Open Source.
- Embedded
- Compiled and Interpreted
- Robust
- Large library and frameworks
- Versatility: Java is a versatile programming language that can be used for a wide range of applications, including desktop and mobile app development, web development, enterprise software development, game development, and more.
- Popularity: Java is one of the most popular programming languages in the world. It has a large and active community of developers, which means there is a lot of support, resources, and job opportunities available for Java developers.
- Object-Oriented Programming (OOP): Java is an object-oriented programming (OOP) language, which means it is based on the concept of classes and objects. This approach makes it easy to organize and manage code, as well as reuse code across multiple projects.
- Platform-Independent: Java is platform-independent, which means that code written in Java can be run on any operating system, such as Windows, Linux, or MacOS, without needing to be recompiled. This makes Java an ideal language for developing cross-platform applications.
- Security: Java has built-in security features, such as automatic memory management and a sandbox security model, which make it a safe language to use for developing secure applications.

What is Java?

Java is a high-level, class-based, object-oriented programming language. Java is a general-purpose language that is used for developing a wide range of applications, including desktop and mobile applications, web applications, enterprise software, and games. It is also widely used in the development of Android mobile applications.



History of Java.



James Gosling and his friends initiated the Java language project in June 1991. The language was initially called ***Oak*** after an oak tree that stood outside Gosling's office. Later the project went by the name *Green* and was finally renamed *Java*, from Java coffee, a type of coffee from Indonesia. Java was first released in 1995 by Sun Microsystems in USA (JDK Beta – version). In 2010 Java is acquired by Oracle Corporation.

Java SE 10	March 20, 2018
Java SE 11 (LTS)	September 25, 2018
Java SE 12	March 19, 2019
Java SE 13	September 17, 2019
Java SE 14	March 17, 2020
Java SE 15	September 15, 2020
Java SE 16	March 16, 2021
Java SE 17 (LTS)	September 14, 2021
Java SE 18	March 22, 2022
Java SE 19	September 20, 2022
Java SE 20	March 21, 2023

Java Versions

Version	Date
JDK Beta	1995
JDK 1.0	January 23, 1996
JDK 1.1	February 19, 1997
J2SE 1.2	December 8, 1998

J2SE 1.3	May 8, 2000
J2SE 1.4	February 6, 2002
J2SE 5.0	September 30, 2004
Java SE 6	December 11, 2006
Java SE 7	July 28, 2011
Java SE 8 (LTS)	March 18, 2014
Java SE 9	September 21, 2017

JAVA Installation

- Install JDK
- Install IDE / Code Editor
    - Eclipse IDE
    - IntelliJ IDEA
    - Visual Studio code (VS Code)

Que: What is JDK?

Ans: JDK: JAVA DEVLOPMENT KIT

collection of tools used for developing and running Java program.

Que: What is IDE?

Ans: IDE: integrated development environment

IDE is a software application that helps programmers develop software code efficiently. It increases developer productivity by combining capabilities such as software editing, building, testing, and packaging in an easy-to-use application.

Que: What is JVM?

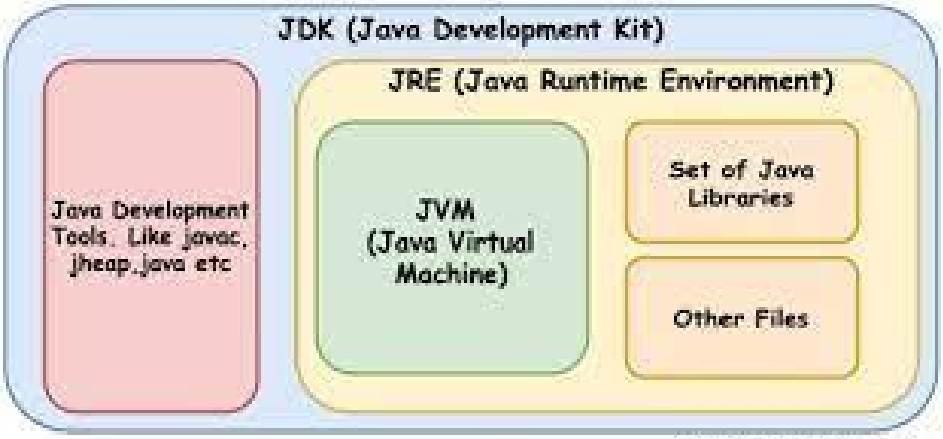
Ans: JVM: JAVA VIRTUAL MECHINE

helps in executing programs developed in java.

Que: What is JRE?

Ans: JRE: JAVA RUNTIME ENVIRONMENT

it provides runtime environment for programs.


JDK (Java Development Kit)
Java Development Tools. Like javac, jheap.java etc
JRE (Java Runtime Environment)
JVM (Java Virtual Machine)
Set of Java Libraries
Other Files

Basic Structure of a Java Program

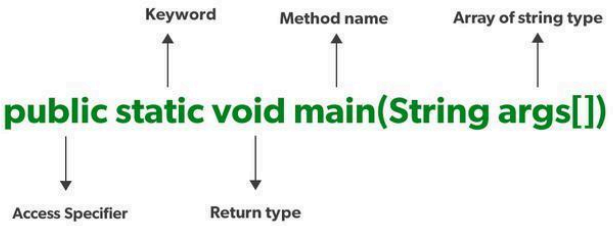
```
package practice;

public class HelloWorld {

     public static void main(String[] args) {


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

     }
}
```

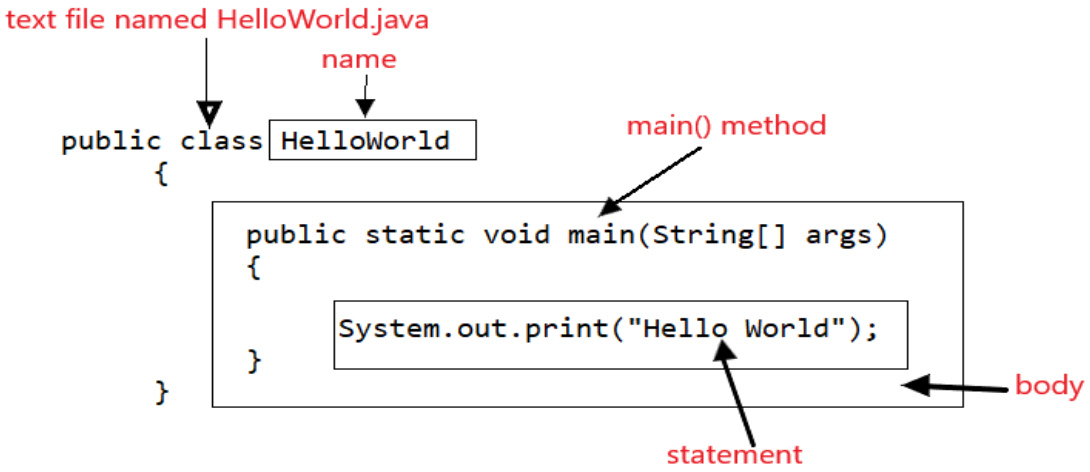


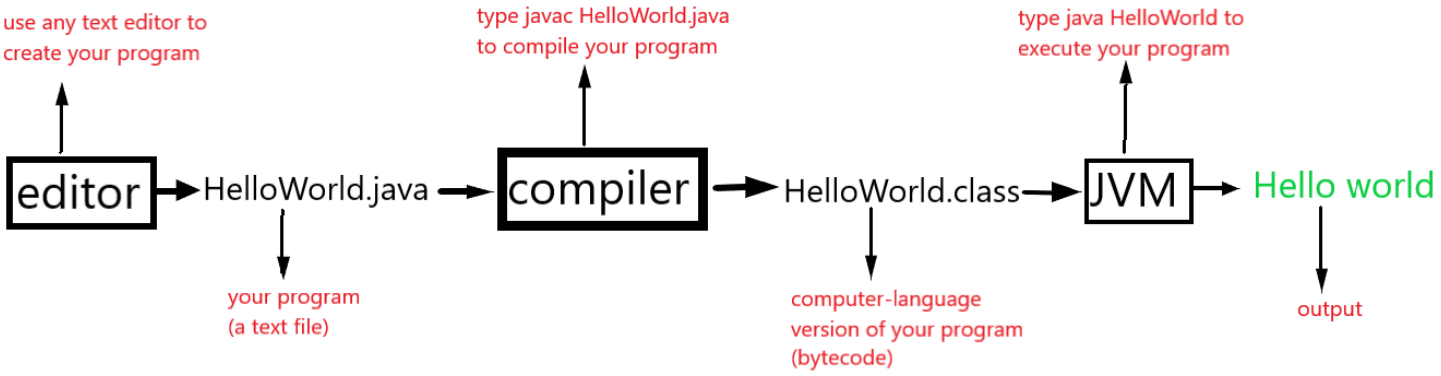
Rule:

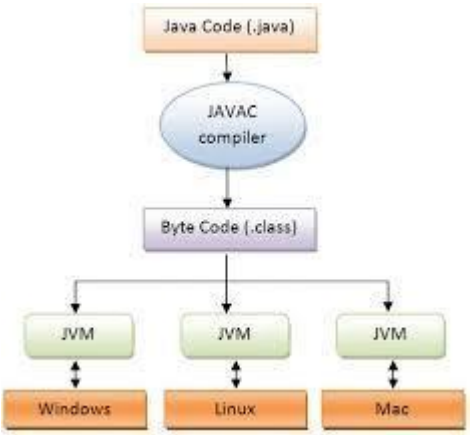
- Save –> Class-Name.Java
- Compilation –> Javac Class-Name.java
- Execution –> Java Class-Name

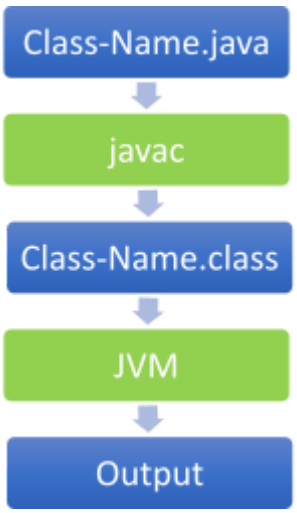
Essential statements of java program: -

- **Class block: -** only class allows us to define the method with logic.
- **The main method: -** it is the initial point of class logic execution.
- **Print statement: -** all kind of data are printed using this statement.


text file named HelloWorld.java
name
public class HelloWorld
{
main() method
public static void main(String[] args)
{
System.out.print("Hello World");
}
}
body
statement


use any text editor to create your program
editor
HelloWorld.java
your program (a text file)
type javac HelloWorld.java to compile your program
compiler
HelloWorld.class
computer-language version of your program (bytecode)
type java HelloWorld to execute your program
JVM
Hello world
output


Java Code (.java)
JAVAC compiler
Byte Code (.class)
JVM
JVM
JVM
Windows
Linux
Mac


Class-Name.java
javac
Class-Name.class
JVM
Output

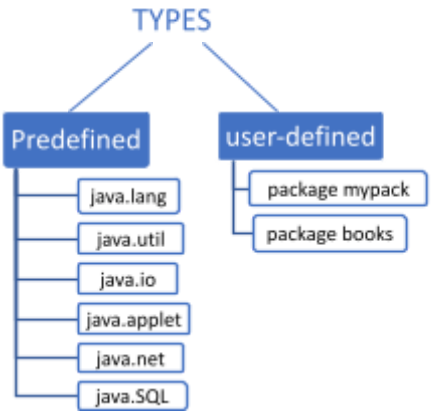
Package

- A package is just like a container that contains collections classes.
- Java uses package to group related classes, interface and sub-packages in any Java project.
- We can assume package is used to store similar files.
- Example: -

```
package mypack;

public class employee
{
        String empId;
        String name;
}
```

- In previous statement will create a package worth name mypack in the project directory.



Naming Conventions

- For Classes, we use PascalConvention. First and Subsequent characters from a word are Capital Letters (Uppercase)

  Example: - HelloWorld, MyScanner, Main, MyCode, PrimeNumber.

- For Functions and Variable, we use camelCaseConvention. Here First Characters is lowercase and the Subsequent Characters are uppercase.

  Example: - main, myCode, myProgram.

Java Comments

- **Single-line comments**: - Single-line comments start with two forward slashes (//).

  Any text between // and the end of the line is ignored by Java (will not be executed).

  Example: -

```
// this is a comment

System.out.print("hello");
```

- **Multi-line Comments: -** Multi-line comments start with /* and ends with */.

  Any text between /* and */ will be ignored by Java.

  Example: -

```
/* this is multi-line comment
In this java program*/

System.out.print("hello");
```

Java language Fundamentals

Tokens in Java

Java Supports five different tokens: -

1. Identifiers
2. Keywords
3. Literals
4. Operators
5. Special Symbols

Every word or symbol written in the below Java program will come under any one of the above five tokes.

```
class test{
        public static void main(String[] args){
                int number = 9;
                double variable = 1.5;
                System.out.println(number+variable);
                System.out.println(variable==number);
                System.out.println("Hello");
        }
}
```

Identifiers

A name in the program is an identifier it may be class name or method name, variable name or label name.

Example: -

```
class Test
{
    public static void main(String[] args){
        int x = 10;
        System.out.println(x);
    }
}
```

Rules for defining Identifiers

1. A Java Identifier is a sequence of characters, where each character may be a letter from a-z or A-Z or a digit from 0-9 or currency symbol $ or connecting punctuation- , if we are using any other symbol we will get compile time error – “IllegalCharacter”.
2. Identifier should not be start with digit.
3. There is no length limit for Java Identifiers but it is not recommended to take more than 15 length.
4. Java Identifiers are case sensitive

   Example: -

```
class FundaDemo
{
    public static void main(String[] args){
        int String = 10;
        System.out.println(String)
    }
}
```

O/P: - 10