

## Information about JAVA

Java is a **general-purpose, class-based, object-oriented** programming language designed for having lesser implementation dependencies. It is a computing platform for application development. **Java is fast, secure, and reliable**, therefore. It is widely used for developing Java applications in laptops, data centers, game consoles, scientific supercomputers, cell phones, etc.

**Java Platform** is a collection of programs that **help programmers** to **develop and run** Java programming applications efficiently. It includes an execution engine, a compiler, and a set of libraries in it. It is a set of computer software and specifications. **James Gosling developed the Java platform at Sun Microsystems**, and the Oracle Corporation later acquired it.

### What is **use of Java**?

- Here are some important Java applications:
- It is **used for developing Android Apps**
- Helps you to **create Enterprise Software**
- **Wide range of Mobile java Applications**
- **Scientific Computing Applications**
- Use for **Big Data Analytics**
- **Java Programming of Hardware devices**
- **Used for Server-Side Technologies like Apache, JBoss, GlassFish, etc.**

### History Of JAVA:

The history of Java is very interesting. Java was originally designed for interactive television, but it was too advanced technology for the digital cable television industry at the time. The history of java starts from Green Team. Java team members (also known as Green Team), initiated this project to develop a language for digital devices such as set-top boxes, televisions etc. But, it was suited for internet programming. Later, Java technology was incorporated by Netscape. The **principles for** creating Java programming

were "Simple, Robust, Portable, Platform-independent, Secured, High Performance, Multithreaded, Architecture Neutral, Object-Oriented,

Interpreted and Dynamic". Currently, Java is used in internet programming, mobile devices, games, e-business solutions etc. There are given the major points that describes the history of java.

1) James Gosling, Mike Sheridan, and Patrick Naughton initiated the Java language project in June 1991.

2) The small team of sun engineers called Green Team.

3) Originally designed for small, embedded systems in electronic appliances like set-top boxes.

4) Firstly, it was called "Greentalk" by James Gosling and file extension was .gt. After that, it was called Oak and was developed as a part of the Green project.

### Why Java named as "Oak"

5) Why Oak? Oak is a symbol of strength and chosen as a national tree of many countries like U.S.A., France, Germany, Romania etc.

6) In 1995, Oak was renamed as "Java" because it was already a trademark by Oak Technologies. Why Java Programming named as "Java"

7) Why had they chosen java name for java language? The team gathered to choose a new name. The suggested words were "dynamic", "revolutionary", "Silk", "jolt", "DNA" etc. They wanted something that reflected the essence of the technology: revolutionary, dynamic, lively, cool, unique, and easy to spell and fun to say. According to James Gosling "Java was one of the top choices along with Silk". Since java was so unique, most of the team members preferred java.

8) Java is an island of Indonesia where first coffee was produced (called java coffee).

9) Notice that Java is just a name not an acronym.

10) Originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995.

- 11) In 1995, Time magazine called Java one of the Ten Best Products of 1995.
- 12) JDK 1.0 released in (January 23, 1996).

## Features of Java

### Great Performance

The Java compiler is designed for performance. Java code is compiled into bytecode and then compiled by the Java compiler. Post that, it is fed to the JVM (Java Virtual Machine) before it's converted to machine level code.

### Inspired by C and C++

C and C++ are long-tenured programming languages and they are the ancestors of modern programming languages like Java and Python. Java is a bit similar to C and C++ but doesn't have features such as pointers and multiple inheritances. Therefore, having an understanding of C and C++ is useful in learning Java.

### Multi-threaded

Multithreading capabilities come built right into the Java language. This means it is possible to build highly interactive and responsive apps with a number of concurrent threads of activity.

### Platform Independence

Java has a philosophy called WORA (Writing Once, Run Anywhere). Java code is compiled into an intermediate format, called bytecode, which is to be executed in the JVM (Java Virtual Machine). Any system that runs a JVM is able to execute the Java code.

### Truly Object-Oriented

Build upon C++ which is semi object-oriented, Java extends the functionality to become a fully object-oriented programming language. Some of the most important features that make it an object-oriented puritan are:

- Abstraction
- Encapsulation
- Inheritance

- Polymorphism

### Robust

Java guides the programmer to adopt important programming habits required for the creation of highly reliable applications. Unlike C and C++, Java relies on a simple memory management model reinforced by the automatic garbage collection feature.

### Secure

Safety features are built into the language and runtime systems. These include runtime checking and static type-checking at compile time. With such features in place, it becomes a daunting task to invade a Java application from the outside.

### Simple

Ease of reading and writing makes any language simple. This holds true for Java as it has a less ambiguous syntax terminology. Anyone can start right off with Java with an understanding of the basic underlying principles of programming.

### Other benefits of Java

Other than the aforementioned features, here are some other great options for choosing Java over other programming languages:

- A multitude of open-source libraries
- Backed by Oracle and high-level support from tech giants like Google and IBM
- Big and mature community support
- Several readily available frameworks for rapidly building reliable apps
- There are a galore of IDEs and tools to help make Java development simpler



## Java Versions

Java Versions	Release Date
JDK Alpha and Beta	1995
JDK 1.0	23rd Jan 1996
JDK 1.1	19th Feb 1997
J2SE 1.2	8th Dec 1998
J2SE 1.3	8th May 2000
J2SE 1.4	6th Feb 2002
J2SE 5.0	30th Sep 2004
Java SE 6	11th Dec 2006
Java SE 7	28th July 2011
Java SE 8	18th Mar 2014
Java SE 9	21st Sep 2017
Java SE 10	20th Mar 2018
JAVA SE 11	25th Sep 2018
JAVA SE 12	19th Mar 2019
JAVA SE 13	17th Sep 2019
JAVA SE 14	17th Mar 2020
JAVA SE 15	15th Sep 2020
JAVA SE 16	20th Mar 2020
JAVA SE 17	14th Sep 2021
JAVA SE 18	22nd Mar 2022
JAVA SE 19	20th Sep 2022 for Open JDK
JAVA SE 20	20th Mar 2023 for Open JDK

## Components Of Java Programming Language

A Java Programmer writes a program in a human-readable language called **Source Code**. Therefore, the **CPU or Chips** never understand the source code written in any programming language.

These **computers or chips** understand only one thing, which is called **machine language or code**. These **machine codes** run at the CPU level. Therefore, it would be different machine codes for other models of CPU.

However, you need to worry about the machine code, as programming is all about the source code. The machine understands this source code and translates them into machine understandable code, which is an executable code.

All these functionalities happen inside the following 3 Java platform components:

### Java Development kit (JDK)

JDK is a software development environment used for making applets and Java applications. The full form of JDK is Java Development Kit. Java developers can use it on Windows, macOS, Solaris, and Linux. JDK helps them to code and run Java programs. It is possible to install more than one JDK version on the same computer.

### Why use JDK?

Here are the main reasons for using JDK:

- JDK contains tools required to write Java programs and JRE to execute them.
- It includes a compiler, Java application launcher, Appletviewer, etc.
- Compiler converts code written in Java into byte code.
- Java application launcher opens a JRE, loads the necessary class, and executes its main method.

### Java Virtual Machine (JVM):

Java Virtual Machine (JVM) is an engine that provides a runtime environment to drive the Java Code or applications. It converts Java bytecode into machine language. JVM is a part of the Java Run Environment (JRE). In other programming languages, the compiler produces machine code for a



particular system. However, the Java compiler produces code for a Virtual Machine known as Java Virtual Machine.

### Why JVM?

Here are the important reasons of using JVM:

- JVM provides a platform-independent way of executing Java source code.
- It has numerous libraries, tools, and frameworks.
- Once you run a Java program, you can run on any platform and save lots of time.

JVM comes with JIT (Just-in-Time) compiler that converts Java source code into low-level machine language. Hence, it runs faster than a regular application.

### Java Runtime Environment (JRE)

JRE is a piece of software that is designed to run other software. It contains the class libraries, loader class, and JVM. In simple terms, if you want to run a Java program, you need JRE. If you are not a programmer, you don't need to install JDK, but just JRE to run Java programs.

### Why use JRE?

Here are the main reasons of using JRE:

- JRE contains class libraries, JVM, and other supporting files. It does not include any tool for Java development like a debugger, compiler, etc.
- It uses important package classes like math, swing, util, lang, awt, and runtime libraries.
- If you have to run Java applets, then JRE must be installed in your system.

### How is Java platform independent?

The meaning of platform independent is that, the java source code can run on all operating systems. A program is written in a language which is a human readable language. It may contain words, phrases etc which the machine does not understand. For the source code to be understood by the

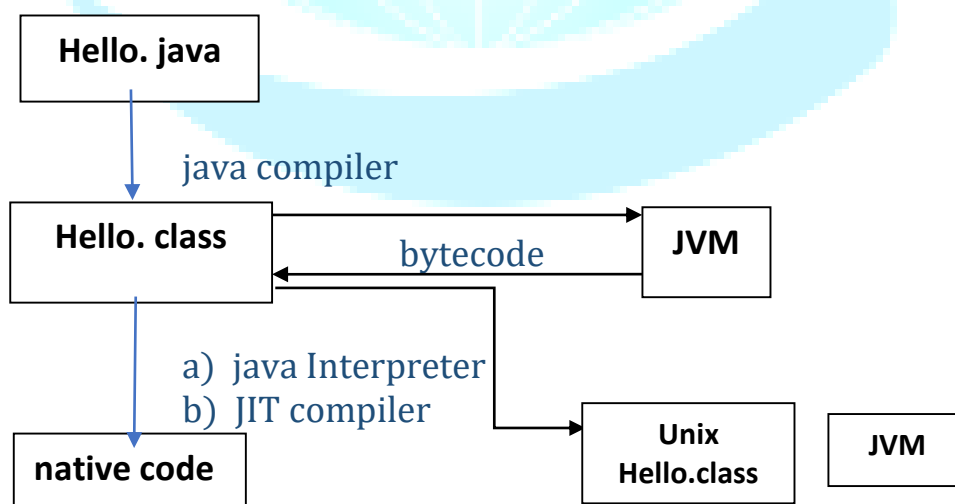
machine, it needs to be in a language understood by machines, typically a machine-level language. So, here comes the role of a compiler. The compiler

converts the high-level language (human language) into a format understood by the machines. Therefore, a compiler is a program that translates the source code for another program from a programming language into executable code. This executable code may be a sequence of machine instructions that can be executed by the CPU directly, or it may be an intermediate representation that is interpreted by a virtual machine. This intermediate representation in Java is the Java Byte Code.

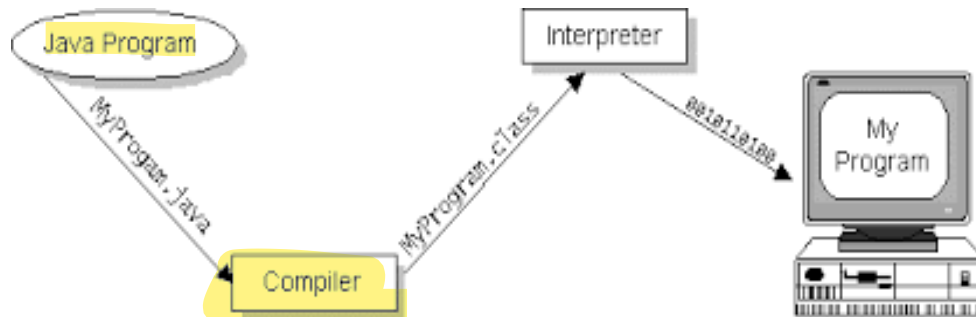
### Step by step Execution of Java Program:

- Whenever, a program is written in JAVA, the javac compiles it.
- The result of the JAVA compiler is the .class file or the bytecode and not the machine native code (unlike C compiler).
- The bytecode generated is a non-executable code and needs an interpreter to execute on a machine. This interpreter is the JVM and thus the Bytecode is executed by the JVM.
- And finally program runs to give the desired output.

In case of C or C++ (language that are not platform independent), the compiler generates an .exe file which is OS dependent. When we try to run this .exe file on another OS it does not run, since it is OS dependent and hence is not compatible with the other OS.







## 1) What is java compiler?

A **compiler** is a program which converts a program from one level of language to another. Example conversion of C++ program into machine code.

The java compiler converts high-level java code into bytecode (which is also a type of machine code).

Compiler is responsible for the following task.

- a. checking the grammatical mistake.
- b. converting source code into byte code with the help of JVM.

## 2) What is java interpreter?

An **interpreter** is a program which converts a program at one level to another programming language at the **same level**. Example conversion of Java program into C++.

Java interpreter is responsible for the following task.

- a. converting the byte code to native code line by line.
- b. executing the native code.
  - i. Because of converting and executing line by line java program executing program is very slower initial versions of java.

**3) What is JIT compiler? (Just-InTime)**

JIT compiler is a single program written in C and C++ with the name java.exe and provided from java2 onward.

JIT compiler task is same as java interpreter. But JIT compiler converts the entire byte code into native code once and then executes.

**4) What is JVM? (Java Virtual machine)**

JVM is a specification provided by sun whose implementation provides environment to run our java applications.

JVM implementation is called as JRE (Java runtime environment).

**5) What is JDK? (Java development kit)**

JDK is a bundle of tools which are required to developed and run your java applications.

**6) State whether following statements are true or false?**

- |   |       |
|---|-------|
| ➤ C program is platform independent.      | False |
| ➤ Java program is platform independent.   | True  |
| ➤ Java compiler is platform independent.  | False |
| ➤ Java Interpreter is platform dependent. | True  |
| ➤ JVM is platform dependent.              | True  |
| ➤ JRE is platform dependent.              | False |

**Stepwise setup of JAVA (i.e., JDK) In Our System:**

JDK is a development environment to develop wide range of applications such as desktop applications, web applications or mobile applications using Java programming language.

JRE (Java Runtime Environment), is also part of JDK. JRE provides the minimum runtime requirements for executing a java application. It consists of Java Virtual Machine (JVM) executables, core classes and some supporting files.

**Step 1 :** Download the latest version of JDK from official website of Oracle or from the link

**Step 2 :** After downloading the JDK, double click on the application file and click on Next. And Finish the installation.

**Step 3:** Go to the installation folder location in C:\ drive, like below,

i.e. `C:\Program Files\Java\jdk1.8.0\bin`

Copy this path.

**Step 4:** Right click on “*MyComputer*” and open *properties* settings.

System > About

LAPTOP-32TR3V49  
VivoBook\_ASUSLaptop X515EA\_X515EA Rename this PC

① Device specifications Copy ^

Device name	LAPTOP-32TR3V49
Processor	11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz 3.00 GHz
Installed RAM	8.00 GB (7.70 GB usable)
Device ID	7CA8392E-618D-4DC3-86A3-08BB2FD3F3BB
Product ID	00356-24524-67231-AAOEM
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

Related links [Domain or workgroup](#) [System protection](#) [Advanced system settings](#)

Windows specifications Copy ^

Edition	Windows 11 Home Single Language
Version	21H2
Installed on	09-06-2022
OS build	22000.739
Experience	Windows Feature Experience Pack 1000.22000.739.0

[Microsoft Services Agreement](#)  
[Microsoft Software License Terms](#)

② Support Copy ^

**Step 5:** click on Advance System Settings

System > About

LAPTOP-32TR3V49  
VivoBook\_ASUSLaptop X515EA\_X515EA Rename this PC

① Device specifications Copy ^

Device name	LAPTOP-32TR3V49
Processor	11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz 3.00 GHz
Installed RAM	8.00 GB (7.70 GB usable)
Device ID	7CA8392E-618D-4DC3-86A3-08BB2FD3F3BB
Product ID	00356-24524-67231-AAOEM
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

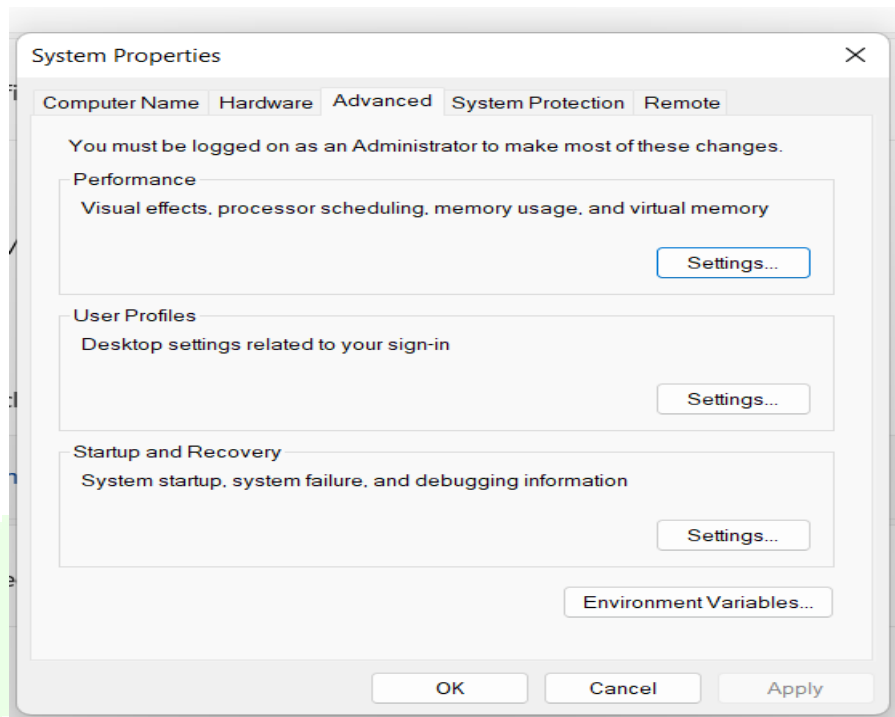
Related links [Domain or workgroup](#) [System protection](#) [Advanced system settings](#)

Windows specifications Copy ^

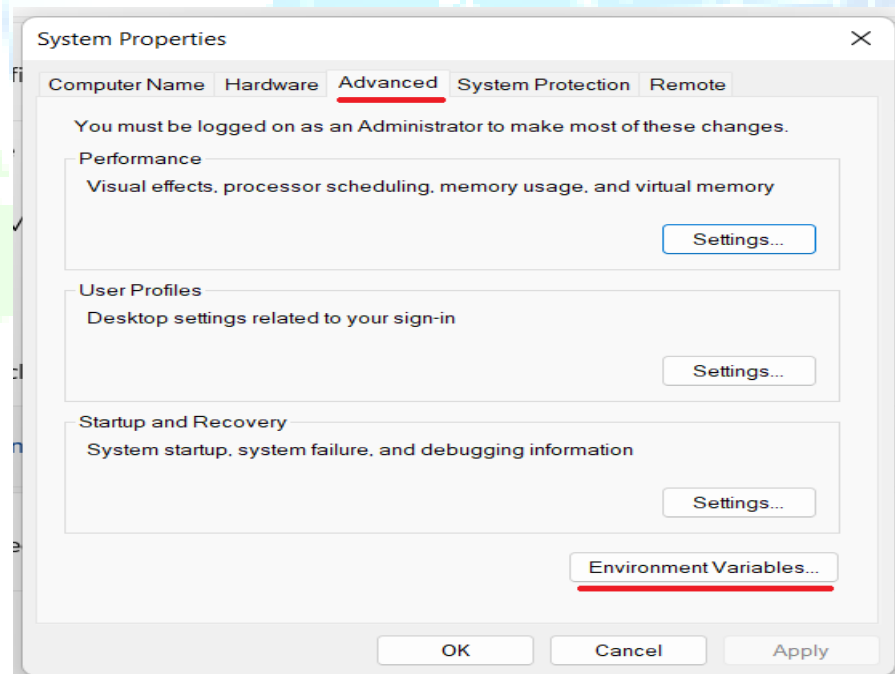
Edition	Windows 11 Home Single Language
Version	21H2
Installed on	09-06-2022
OS build	22000.739
Experience	Windows Feature Experience Pack 1000.22000.739.0

[Microsoft Services Agreement](#)  
[Microsoft Software License Terms](#)

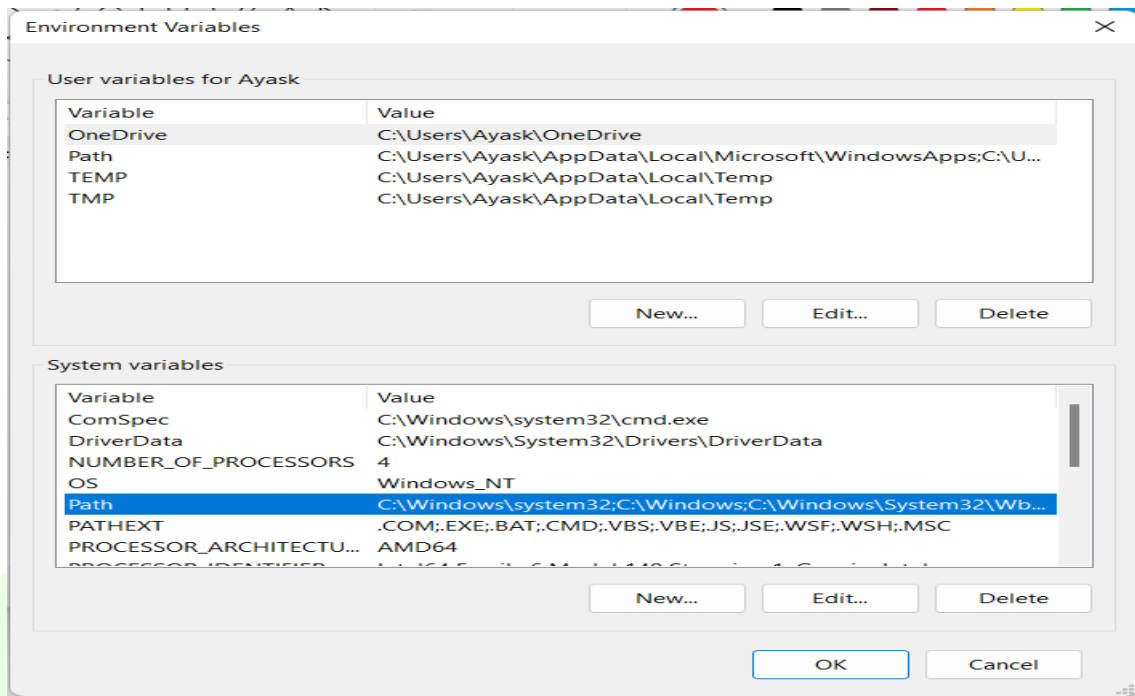
② Support Copy ^



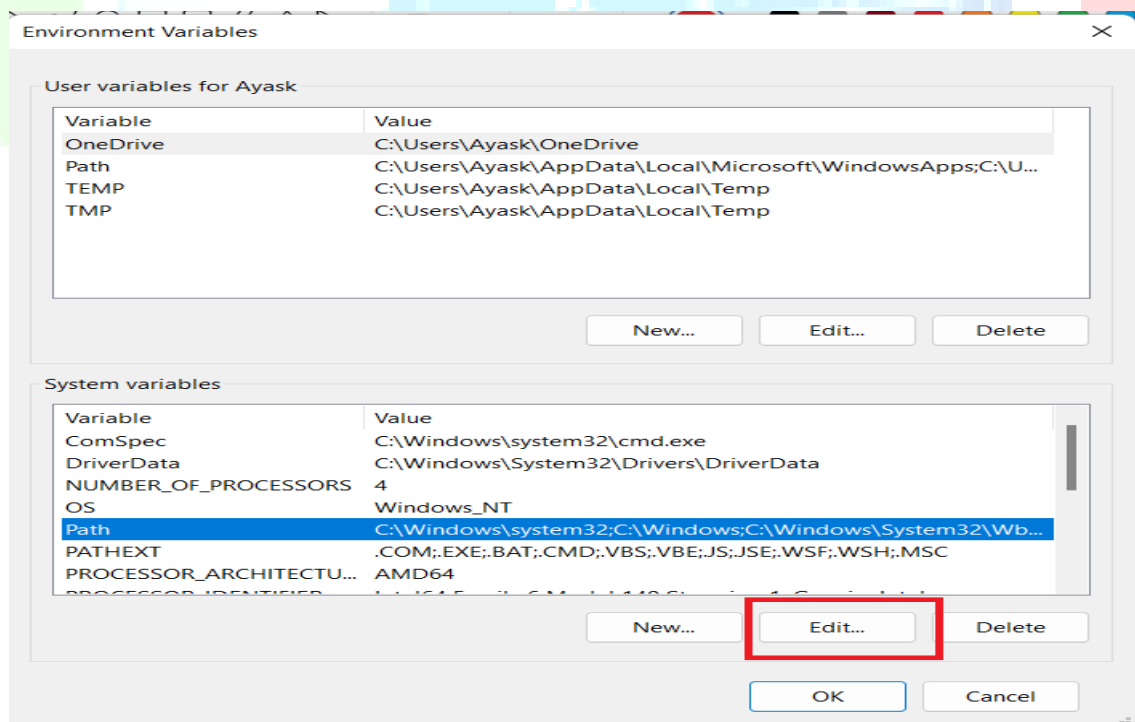
**Step 6:** Click on *Advanced* tabbed pane and after that click on *Environment Variables* button.



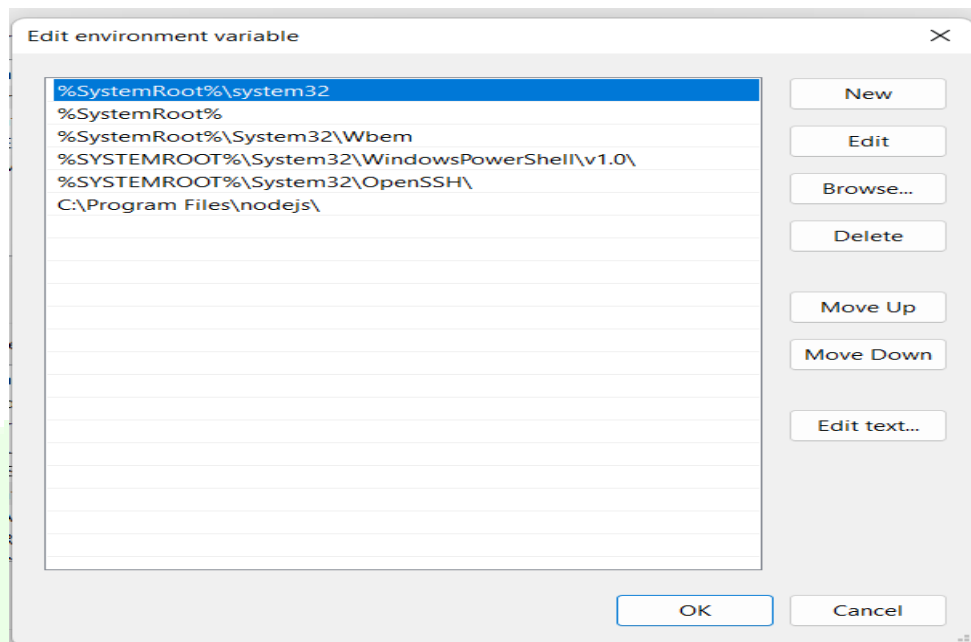
**Step 7:** Select the *Path* variable in *System variables* dialog box,



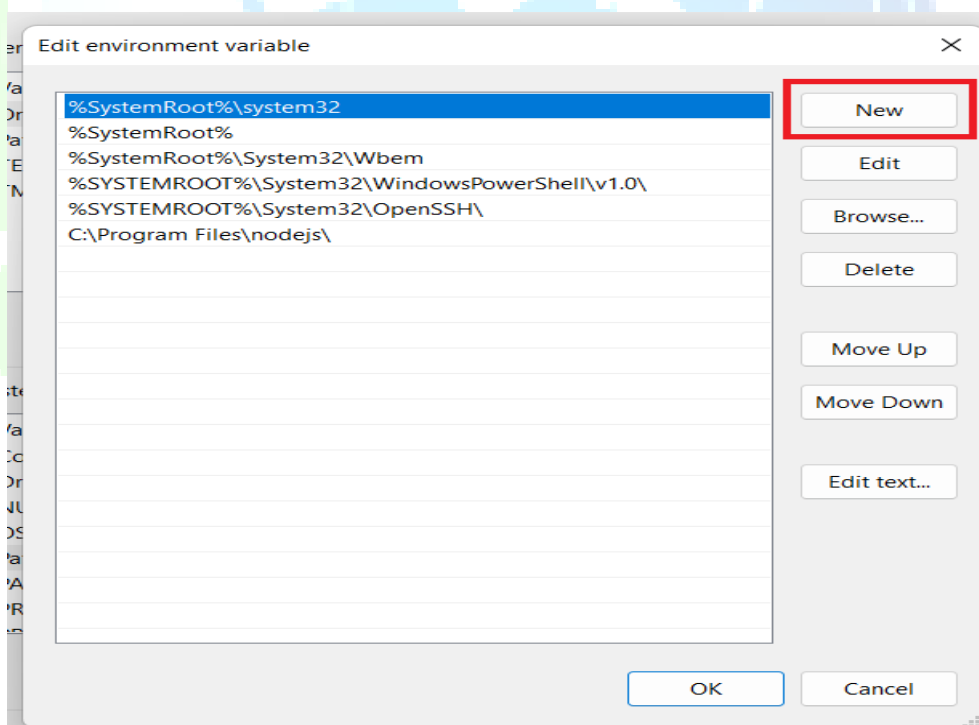
**Step 8:** click on edit button



Step 9: next dialog box will be shown as below,

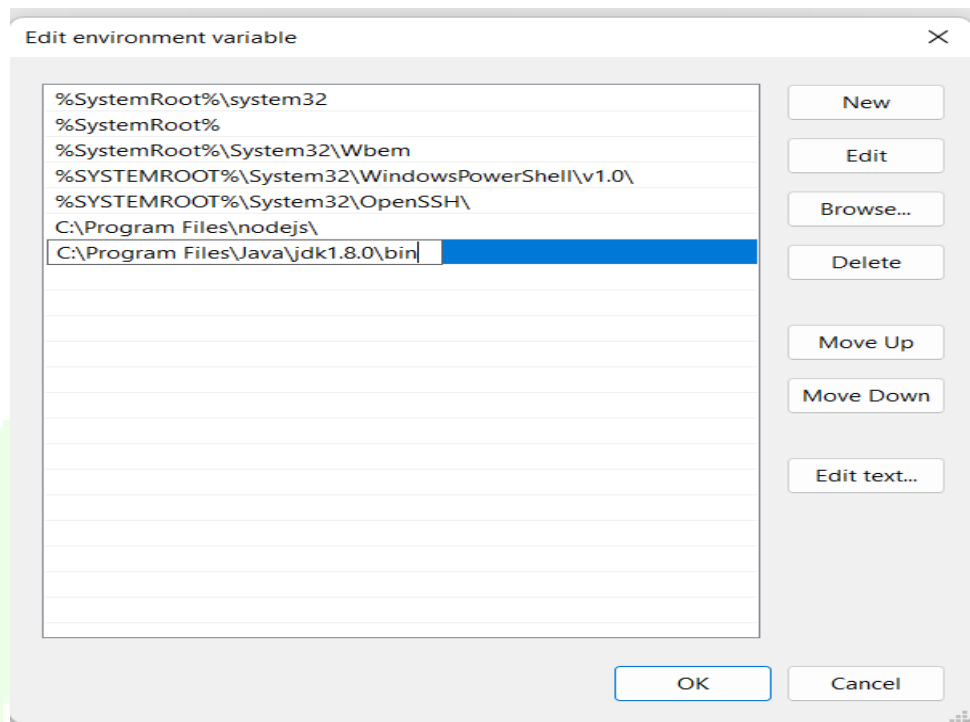


Step 10: Click on New button





**Step 11:** Paste the copied file here



**Step 12:** Click on OK Button, then again OK button, again OK button.

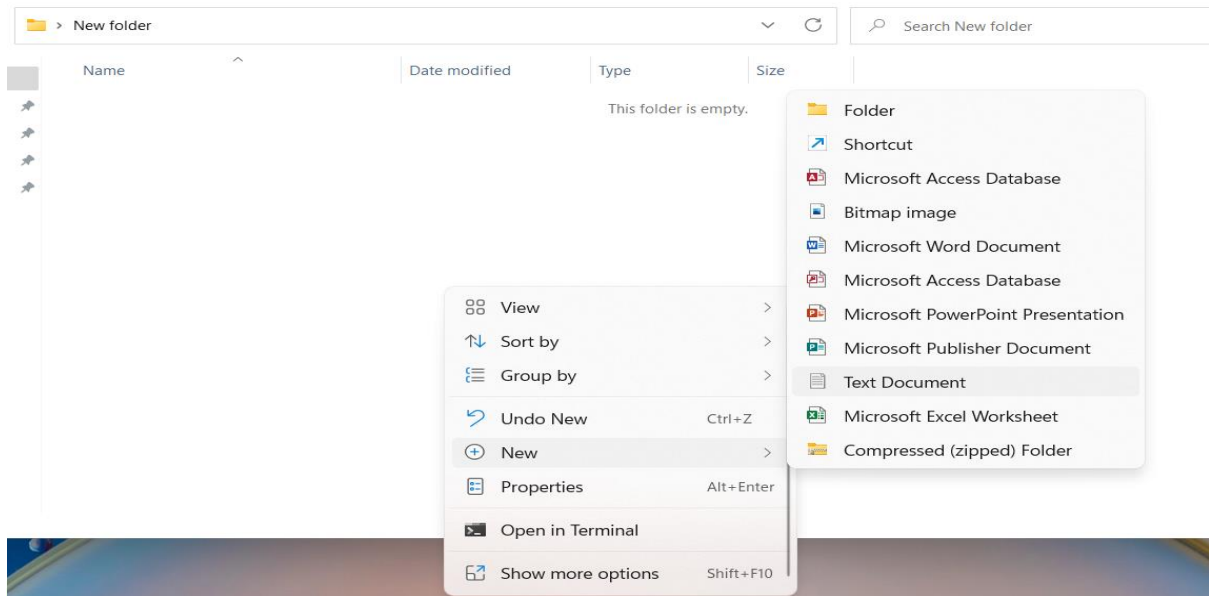
Now, your JDK means java is installed for your system.

**Step 13:** To check JDK is installed in your system or not

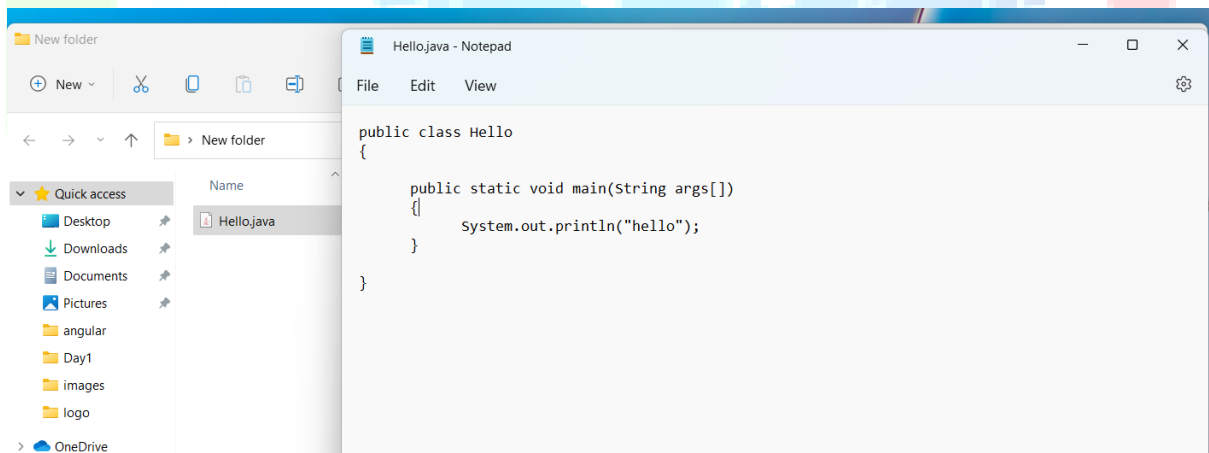
- Open command prompt
- Type **java -version** command to check the version of JDK installed in your system.

## First Program in java

Step 1: create a new text document in a folder with name **Hello.java**



Step 2: Write a class Hello in the file as below,



Step 3: Open the command prompt in same location, then compile a Hello.java file with below command,

➤ javac Hello.java

Step 4: Run the program with the help of below command

➤ java Hello