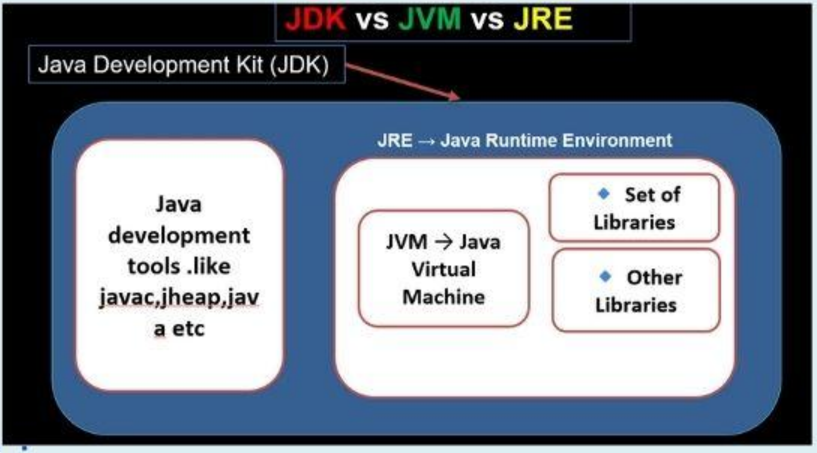
🚀 200 DAYS JAVA DSA SERIES

🎯 DAY 02:HOW TO INSTALL JDK

JVM VS JDK VS JRE IN SIMPLE WORDS!



JDK vs JVM vs JRE

Java Development Kit (JDK)

JRE → Java Runtime Environment

Java development tools .like javac,jheap,java etc

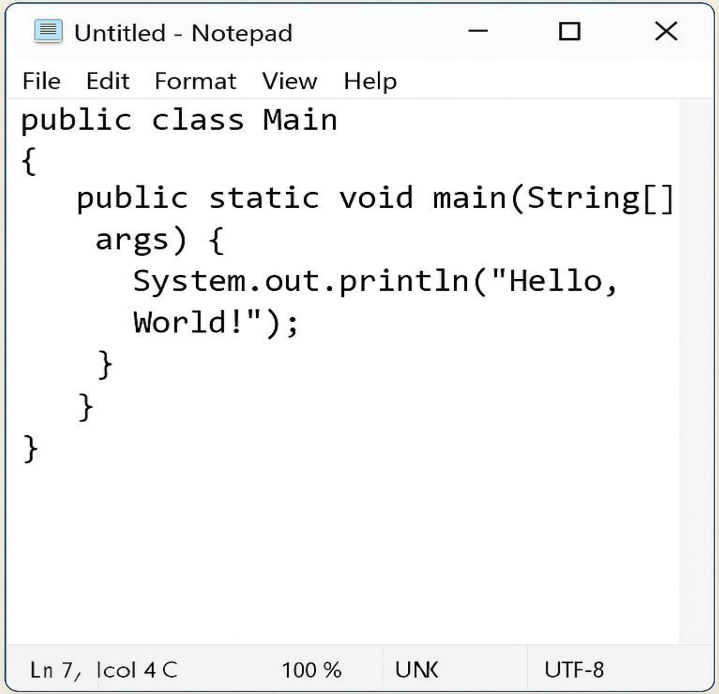
JVM → Java Virtual Machine

◆ Set of Libraries

◆ Other Libraries

Topics Covered

1. Installing JDK
2. First Java Class using Text Editor
3. Compiling and Running Java Code
4. Anatomy of a Java Class
5. File Extensions in Java (.java. class)
6. Difference: JDK vs JVM vs JRE



```
public class Main
{
   public static void main(String[]
    args) {
      System.out.println("Hello,
      World!");
    }
   }
}
```

Lecture -02

🔧 JDK Installation Steps :

1.Official Website Visit karo

➤ JDK download karne ke liye Oracle ya OpenJDK ki website pe jao.

2.Apne OS ke hisaab se Select karo

➤ Windows, macOS, ya Linux — jo bhi aapka system hai, uske according JDK version select karo.

Example: **Windows x64 Installer**

3.Download and Install karo

➤ Installer download hone ke baad, uspe double-click karo aur default settings ke saath install kar do.

4.Environment Variable Set karo

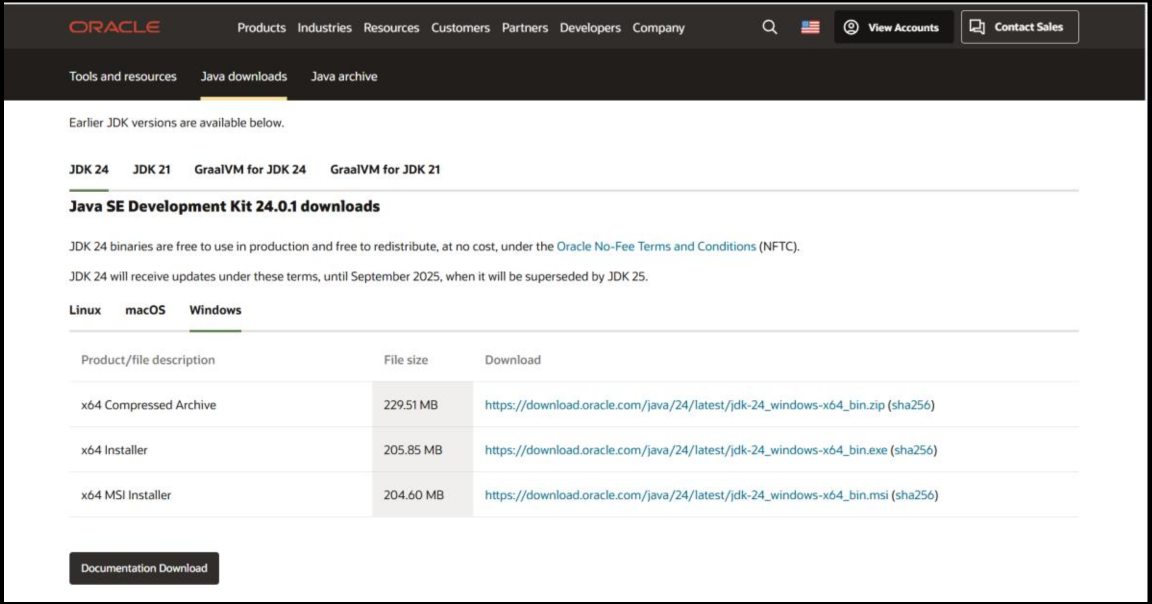
➤ System Properties → Environment Variables → Path mein jaake JDK ka bin path add karo.
➤ Example: C:\Program Files\Java\jdk-21\bin

5.Check Installation using Command Prompt

➤ CMD open karo aur type karo:

```
java -version
javac -version
```

➤ Agar version show ho raha hai, toh JDK successfully install ho gaya hai!



Earlier JDK versions are available below.

Java SE Development Kit 24.0.1 downloads

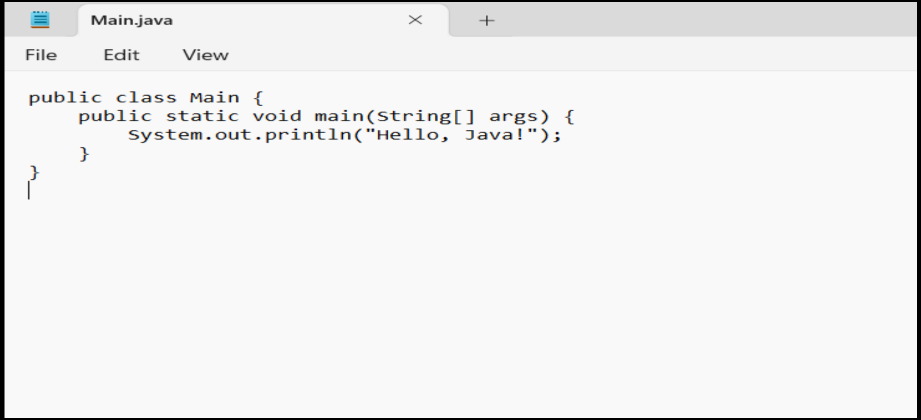
JDK 24 binaries are free to use in production and free to redistribute, at no cost, under the Oracle No-Fee Terms and Conditions (NFTC).

JDK 24 will receive updates under these terms, until September 2025, when it will be superseded by JDK 25.

Product/file description	File size	Download
x64 Compressed Archive	229.51 MB	https://download.oracle.com/java/24/latest/jdk-24_windows-x64_bin.zip (sha256)
x64 Installer	205.85 MB	https://download.oracle.com/java/24/latest/jdk-24_windows-x64_bin.exe (sha256)
x64 MSI Installer	204.60 MB	https://download.oracle.com/java/24/latest/jdk-24_windows-x64_bin.msi (sha256)

First Java Program Using Text Editor

- **Notepad open karo** → Type this code:



```
public class Main {
    public static void main(String[] args) {
        System.out.println("Hello, Java!");
    }
}
```

- **File ko save karo**
  → Main.java naam se save karo
  → **Save as type**: All Files
  → **Encoding**: UTF-8

- **CMD open karo**
  → Jahan file save ki hai, us folder mein jao using cd

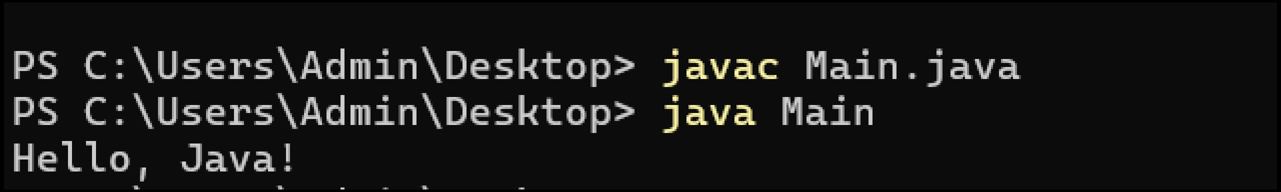
- Compile karo

```
javac Main.java
```

- Run karo

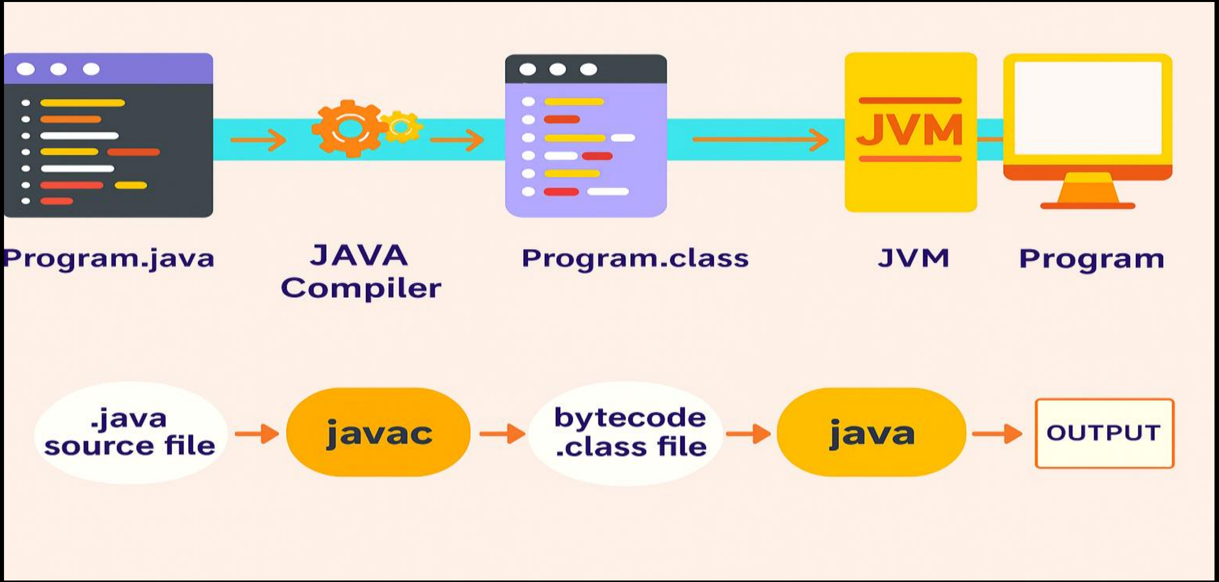
```
java Main
```

- Output: Hello, Java!

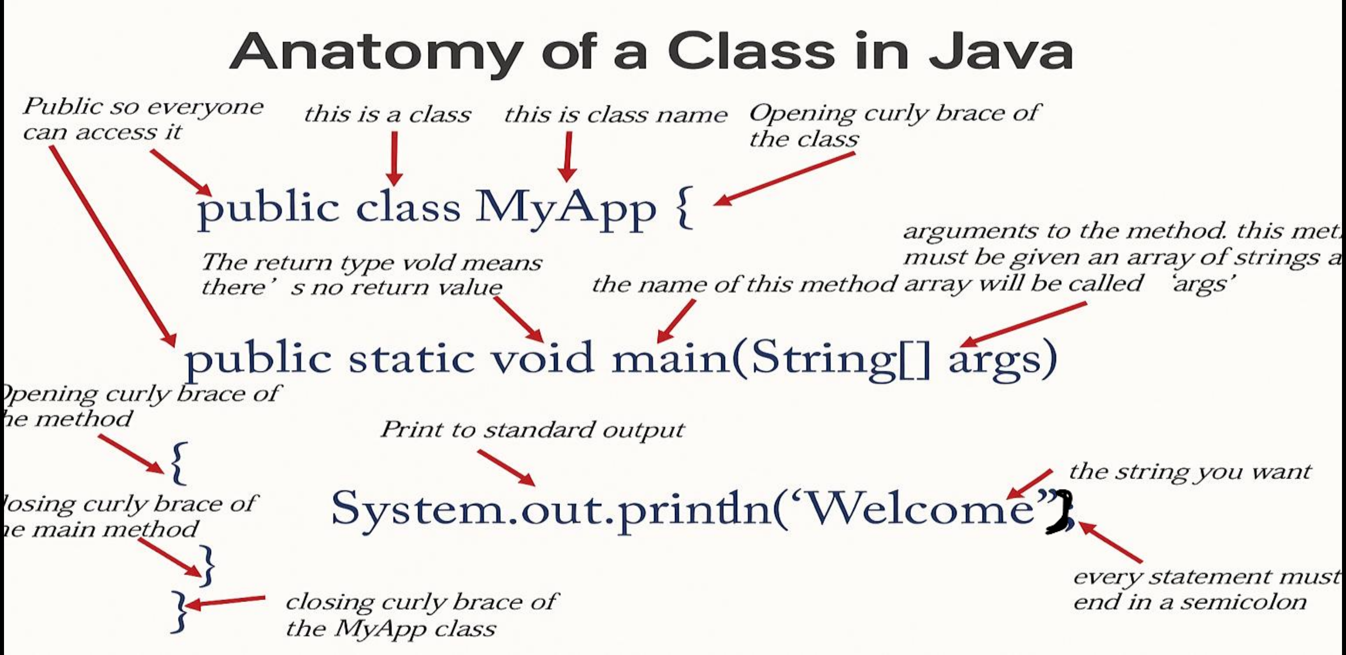


```
PS C:\Users\Admin\Desktop> javac Main.java
PS C:\Users\Admin\Desktop> java Main
Hello, Java!
```

❑ Compiling and Running



❑ Anatomy of a Class



Anatomy of a Class in Java

```
public class MyApp {
public static void main(String[] args)
{
System.out.println(‘Welcome”);
}
}
```

Public so everyone can access it

this is a class

this is class name

Opening curly brace of the class

The return type vold means there’ s no return value

the name of this method

arguments to the method. this met must be given an array of strings a array will be called ‘args’

pening curly brace of he method

Print to standard output

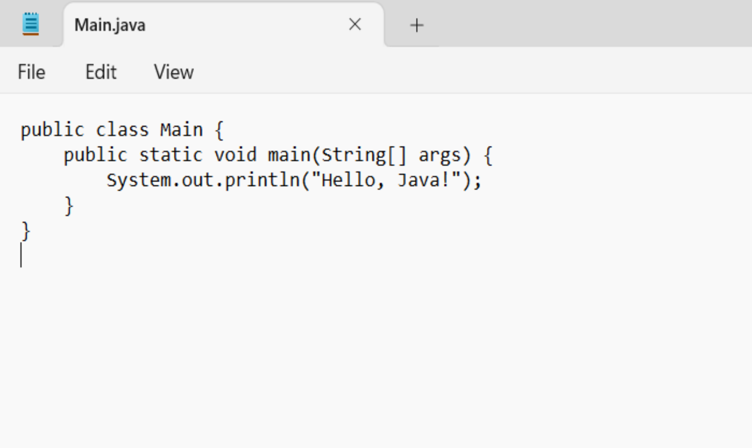
the string you want

losing curly brace of e main method

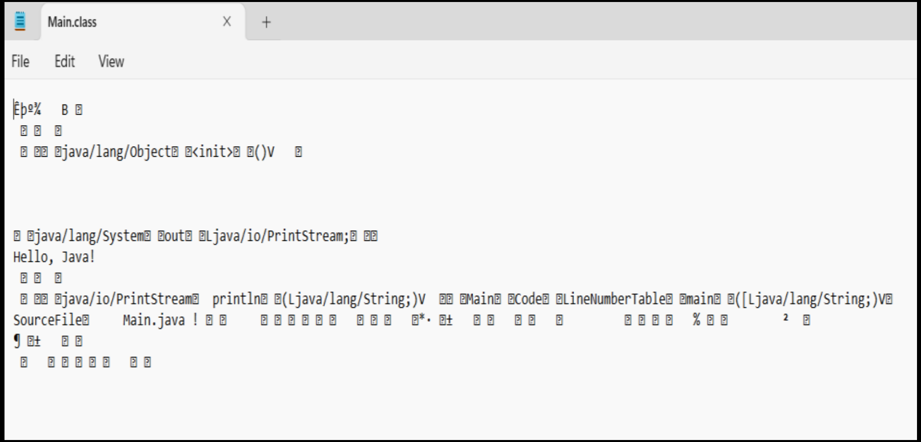
every statement must end in a semicolon

closing curly brace of the MyApp class

File Extension



```java
public class Main {
    public static void main(String[] args) {
        System.out.println("Hello, Java!");
    }
}
```



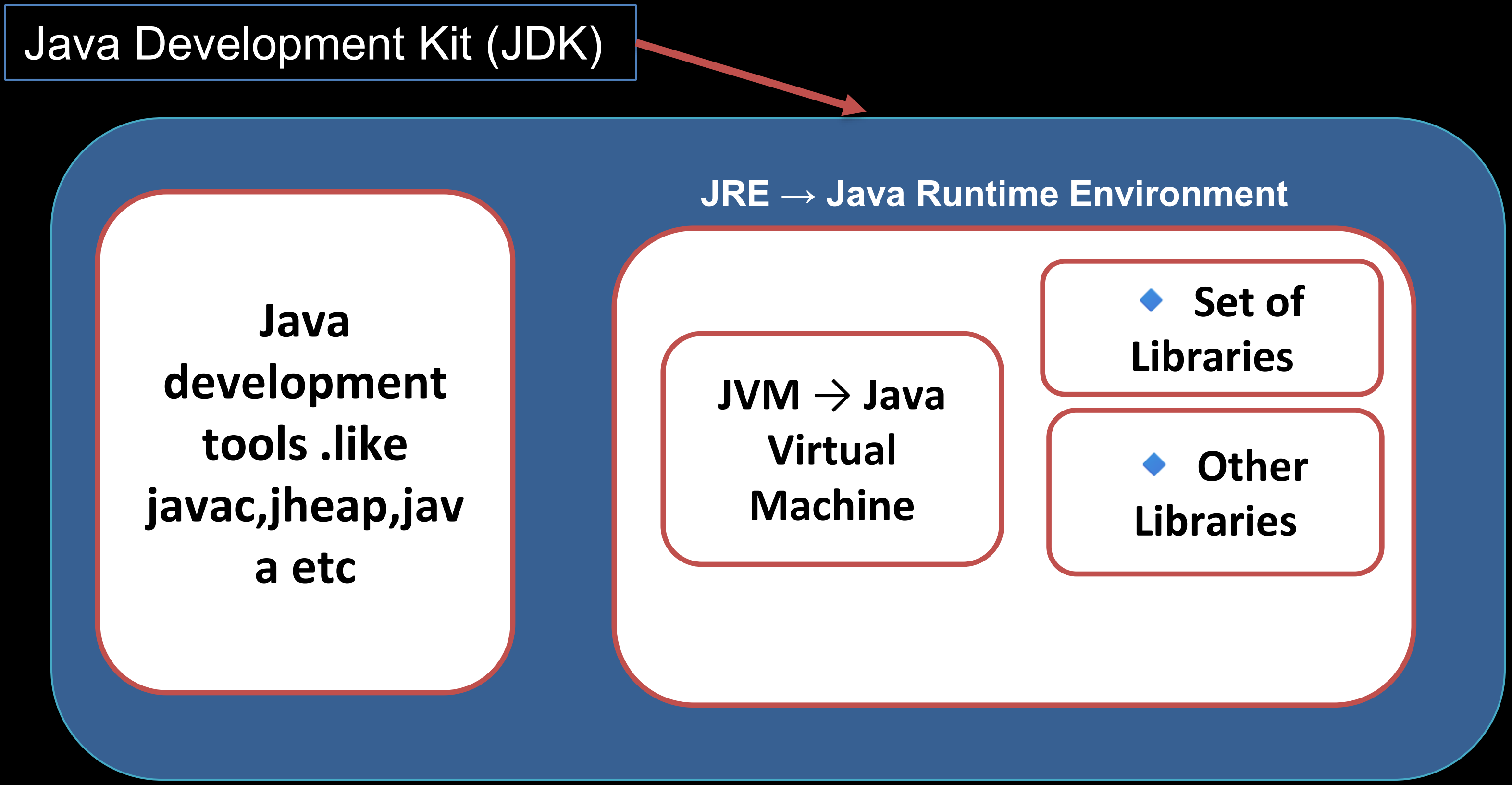
```
Êþº¾   B □
 □ □  □
 □ □□ □java/lang/Object□ □<init>□ □()V   □




□ □java/lang/System□ □out□ □Ljava/io/PrintStream;□ □□
Hello, Java!
 □ □  □
 □ □□ □java/io/PrintStream□  println□ □(Ljava/lang/String;)V  □□ □Main□ □Code□ □LineNumberTable□ □main□ □([Ljava/lang/String;)V□
SourceFile□     Main.java ! □ □    □ □ □ □ □ □   □ □ □   □*· □±   □ □   □ □   □        □ □ □ □   % □ □       ²  □
¶ □±   □ □
 □   □ □ □ □ □   □ □
```

Feature	.java File (Source Code)	.class File (Bytecode)
Content	Java source code hota hai	Java bytecode hota hai
Language Level	High-level human readable language hota hai	Low-level JVM readable code hota hai
Use Case	Development ke liye use hota hai	Execution ke liye JVM use karta hai
Editability	Is file ko hum edit kar sakte hain	Ye file editable nahi hoti (normally)
Who Uses It	Java developer use karta hai	Java Virtual Machine (JVM) use karti hai

JDK vs JVM vs JRE



Java Development Kit (JDK)

Java development tools .like javac,jheap,java etc

JRE → Java Runtime Environment

JVM → Java Virtual Machine

◆ Set of Libraries

◆ Other Libraries

💻 JDK (Java Development Kit) Kya Hai?

👉 **JDK ek software toolkit hai jo Java applications banane (develop karne) ke liye use hota hai.**

Agar aap Java mein coding karte ho, toh JDK aapko **code likhne, compile karne, aur run karne** mein madad karta hai.

🔧 JDK Ke Andar Kya-Kya Hota Hai?

Component	Kya Kaam Karta Hai (Hinglish mein)
JRE (Java Runtime Environment)	Java program ko run karne ke liye zaroori cheezein (JVM + Libraries)
JVM (Java Virtual Machine)	.class bytecode ko machine-level instructions mein convert karta hai
javac (Compiler)	.java file ko .class bytecode mein convert karta hai
java (Launcher)	Java program ko start/run karta hai
javadoc, **jdb**, etc.	Documentation aur debugging ke tools
Built-in Libraries	Reusable functions jaise String, ArrayList, Scanner, etc.

🔄 JDK Ka Formula Samjho:

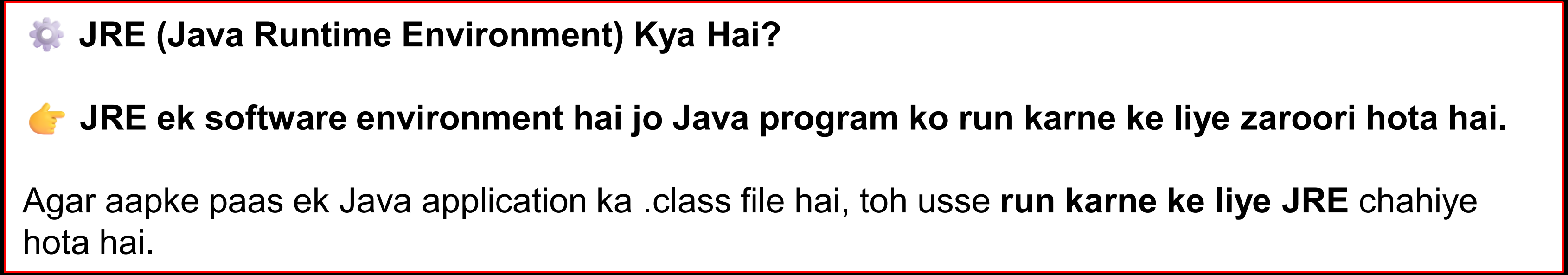
JDK = JRE + Development Tools

❑ Agar sirf Java program **run** karna ho → **JRE** enough hai

❑ Agar Java program **banana, compile karna, aur run** karna ho → **JDK** chahiye

📦 JDK Kisne Banaya?

Sun Microsystems ne JDK banaya tha, aur ab **Oracle** usse maintain karta hai.



⚙️ JRE (Java Runtime Environment) Kya Hai?

👉 **JRE ek software environment hai jo Java program ko run karne ke liye zaroori hota hai.**

Agar aapke paas ek Java application ka .class file hai, toh usse **run karne ke liye JRE** chahiye hota hai.

🔧 JRE Ke Andar Kya-Kya Hota Hai?

Component	Kya Kaam Karta Hai (Hinglish mein)
JVM (Java Virtual Machine)	Bytecode ko machine-level instructions mein convert karta hai aur program execute karta hai
Set of Libraries	Java ke predefined packages jaise java.lang, java.util, etc. – jo program ko run karne mein help karte hain
Other Supporting Files	Configuration files, class loaders, etc. jo JVM ko support karte hain execution ke waqt

🔄 JRE Ka Formula Samjho:

JRE = JVM + Set of Libraries

✅ **JRE sirf Java program ko run** karta hai

❌ **Java code ko likhne ya compile karne** ke kaam nahi aata (uske liye JDK chahiye hota hai)

📥 Real Life Example:

- Jab aap **Java-based software** (jaise Minecraft ya Eclipse) install karte ho, toh system mein **JRE** hona chahiye taki wo run ho sake.

- **Developers** JDK use karte hain, aur **users** ko sirf JRE ki zaroorat hoti hai.

🔹 Set of Libraries

Ye woh **core libraries** hoti hain jo har Java program ke execution ke liye zaroori hoti hain.

Examples include:

- java.lang (basic classes like String, Math, etc.)
- java.util (collections, data structures)
- java.io (input/output handling)
- java.net (networking)
- java.sql (database access)

📌 **Part of JRE** – JVM inhe use karta hai execution ke waqt.

🔹 Other Libraries

Ye **extra libraries** hoti hain jo developer ki requirement ke according use ki jaati hain.

Examples:

- Apache Commons
- Google Guava
- Jackson (JSON processing)
- Hibernate (ORM tool)
- JUnit (testing)

📌 **Part of Development** – Ye JDK ke time pe include ki jaati hain project mein.

🧠 JVM (Java Virtual Machine) Kya Hai?

👉 **JVM ek virtual machine hai jo Java program ke bytecode ko aapke system pe run karne layak machine code mein convert karti hai.**

JVM hi Java ko **“Write Once, Run Anywhere”** banata hai — kyunki JVM har OS ke liye alag hota hai, par Java code same hota hai.

🔧 JVM Ka Kya Kaam Hai?

Feature / Role	Explanation (Hinglish mein)
Bytecode Execution	.class file (jo javac se banta hai) ka bytecode JVM execute karta hai
Platform Independence	JVM har OS ke liye alag hota hai, isliye Java code ko har system pe chalaya ja sakta hai
Memory Management	JVM automatically memory allocate aur free karta hai (Garbage Collection)
Security & Exception Handling	JVM secure execution karta hai aur runtime errors handle karta hai
Multithreading Support	JVM threads ko efficiently manage karta hai Java program mein

🔄 JVM Ka Formula Samjho:

JVM = Bytecode Executor + Memory Manager + Security System + OS Adapter

Component Name	Explanation (Hinglish Mein)
Bytecode Executor	.class file ke bytecode ko read karta hai aur usse execute karta hai
Memory Manager	Java program ke liye memory allocate karta hai aur unused memory ko automatically free karta hai (Garbage Collection)
Security System	Program ko secure environment mein run karta hai — unauthorized access se bachata hai
OS Adapter	JVM system ke operating system ke according work karta hai (e.g., Windows, Mac, Linux) — Java code ko machine-specific instructions mein convert karta hai

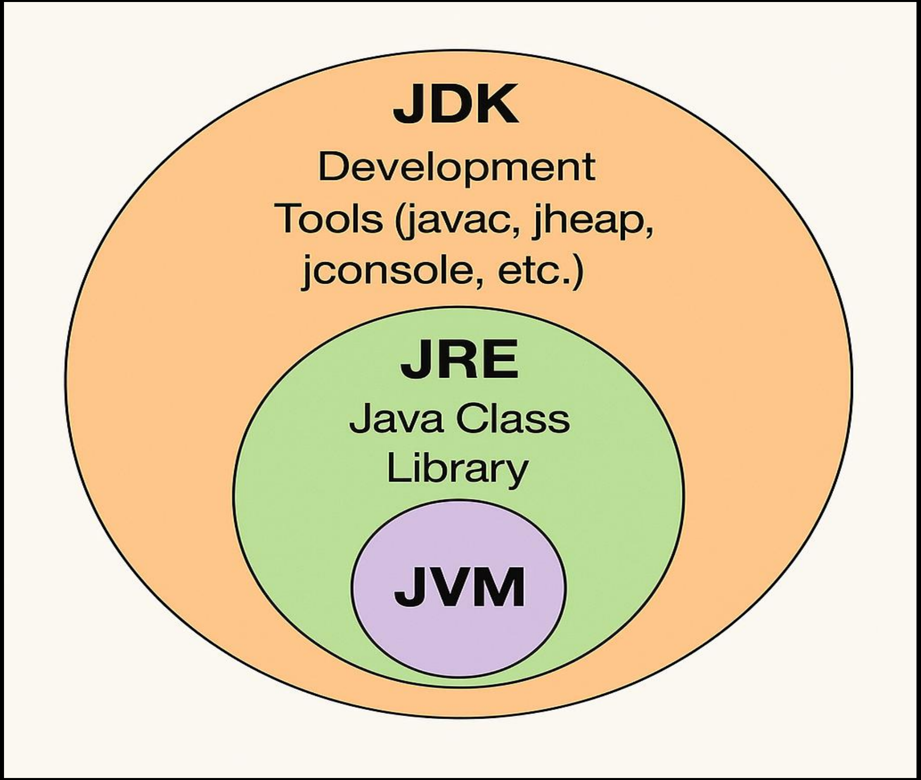
📥 Real Life Example:

- Jab aap terminal/command prompt pe likhte ho:

```
java MyProgram
```

Toh JVM aapke MyProgram.class file ko uthata hai aur system-specific code mein convert karke run karta hai.

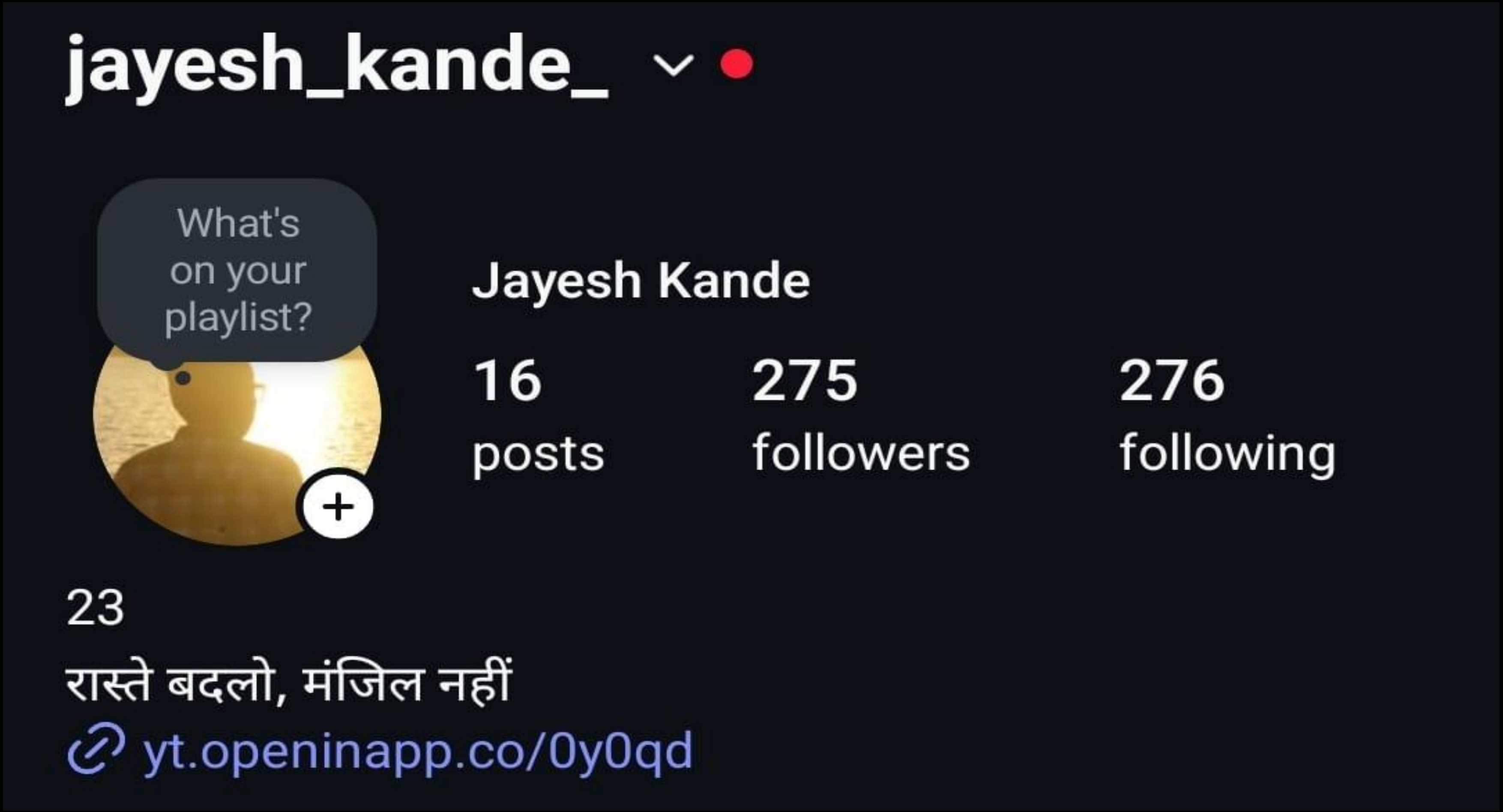
Feature/Aspect	**JDK (Java Development Kit)**	**JRE (Java Runtime Environment)**	**JVM (Java Virtual Machine)**
Full Form	Java Development Kit	Java Runtime Environment	Java Virtual Machine
Use	Java programs **banane, compile karne aur run** karne ke liye	Java programs **sirf run** karne ke liye	Java bytecode **execute** karne ke liye
Includes	JRE + Development Tools (javac, javadoc, etc.)	JVM + Libraries (java.lang, java.util, etc.)	Execution engine only (JVM core)
For Whom?	Java developers	Java application users	Internally used by JRE/JDK
Contains Compiler?	✅ Haan (javac)	❌ Nahi	❌ Nahi
Contains JVM?	✅ Haan	✅ Haan	❌ JVM khud ek component hai
Can Run Java Code?	✅ Haan	✅ Haan	✅ Haan (sirf execution karta hai)
Can Compile Code?	✅ Haan	❌ Nahi	❌ Nahi



1. **JDK** is the **superset** – it includes JRE and development tools.

2. **JRE** is a **subset of JDK** and a **superset of JVM** – it includes the Java Class Library and JVM.

3. **JVM** is the **innermost subset** – it runs the Java bytecode.



jayesh_kande_

Jayesh Kande

16 posts

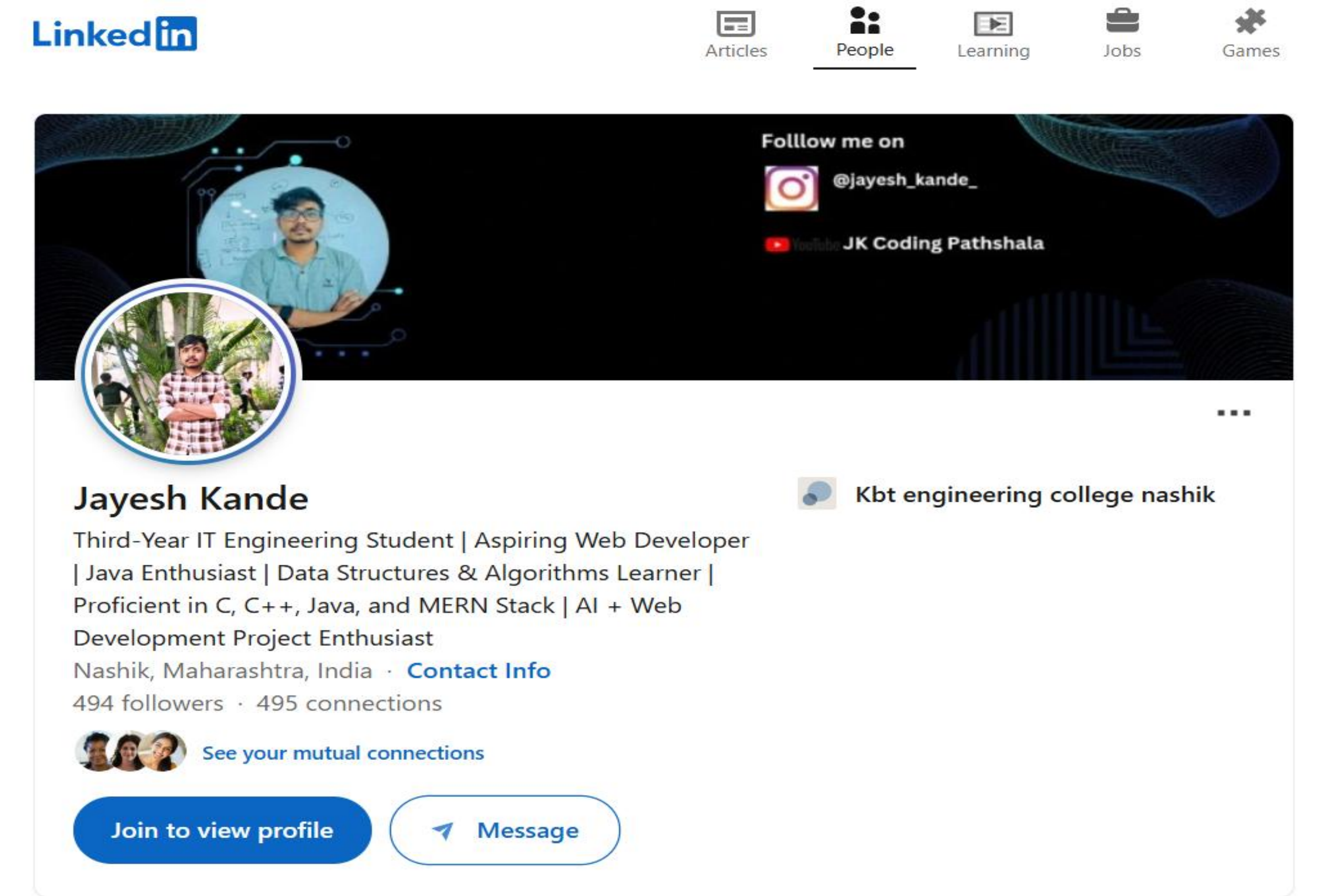
275 followers

276 following

23

रास्ते बदलो, मंजिल नहीं

yt.openinapp.co/0y0qd



Jayesh Kande

Third-Year IT Engineering Student | Aspiring Web Developer | Java Enthusiast | Data Structures & Algorithms Learner | Proficient in C, C++, Java, and MERN Stack | AI + Web Development Project Enthusiast

Nashik, Maharashtra, India · Contact Info

494 followers · 495 connections

See your mutual connections

Kbt engineering college nashik

✨ Thank You for Watching! ✨

📲 Follow us on Instagram:**@jayesh_kande_**
🔗 Connect with us on LinkedIn:[**Jayesh Kande**]