

Core - Java

Module – Introduction to Java



Key Features of Java

- **Platform Independence (Write Once, Run Anywhere - WORA)**
 - Java programs are compiled into **bytecode**, which can run on any system with a JVM.
- **Object-Oriented**
 - Java follows **OOP principles** (Encapsulation, Inheritance, Polymorphism, Abstraction).
- **Automatic Memory Management (Garbage Collection)**
 - The **JVM handles memory allocation** and garbage collection automatically.
- **Multi-threading**
 - Java supports **concurrent execution** of multiple threads for better performance.
- **Robust & Secure**
 - Features like **exception handling**, strong memory management, and runtime security make Java reliable.
- **Rich API & Large Ecosystem**
 - Java offers APIs for networking, data structures, I/O operations, databases, and more.

Java Architecture

- **1. Java Development Kit (JDK)**
 - Contains the **compiler (javac)**, **debugger**, and development tools.
- **2. Java Runtime Environment (JRE)**
 - Includes libraries and JVM, required for running Java applications.
- **3. Java Virtual Machine (JVM)**
 - Converts Java **bytecode into machine code** and executes it on the host system.

Java version and Feature

Java Version	Year	Key Features	Example Code
Java 1.0	1996	- First stable release - AWT (Abstract Window Toolkit) - Applets - Basic OOP support	<pre>public class HelloWorld { public static void main(String[] args) {System.out.println("Hello, World!"); } }</pre>
Java 1.1	1997	- JDBC - Inner classes - RMI - Reflection API	<pre>Class Outer { class Inner { void display() { System.out.println("Inner Class !") } } }</pre>
Java 1.2	1998	- SwingGUI - Collections -Java 2 Platform	<pre>List<String> list = new ArrayList<>(); List.add("java");</pre>
Java 1.3	2000	- HotSpot JVM - JNDI	<pre>System.out.println ("Hotpot JVM Introduced");</pre>
Java 1.4	1996	- Assert Keyword - Logging APP - XML Parsing support (JAXP)	<pre>Int age = 10 ; Assert age > 18 : "Not an Adult ";</pre>

Java Version	Year	Key Features	Example Code
Java 5	2004	- Generics - Enhanced For Loop - Autoboxing / Un Boxing - Enums , annotations	<pre>List<Integer> numbers = new ArrayList<>(); For (int num : numbers) { System.out.println(num); }</pre>
Java 6	2004	- Scripting APO - Improvements to JDBC	<pre>ScriptEngine engine = new ScriptEngineManager().getEngineByName("javaScript"); Engine.eval("print("Hello Javascript")");</pre>
Java 7	2011	- Try with resources - Diamond operator - Switch with strings	<pre>Try { BufferedReader br = new BufferedReader(new FileReader("file.txt")) System.out.println(br.readLine());} Catch(IOException e){ e.printStackTrace(); }</pre>
Java 8	2014	- Lambda Expressions - Stream API - Date and Time API	<pre>List<String> names = Arrays.asList("Alice","Bob"); Names.forEach(name -> System.out.println(name));</pre>
Java 9	2017	- Jshell (REPL) - Module System (JPMS)	<pre>Module come.example { exports com.example.utils}</pre>
Java 10	2018	- Var keyword	<pre>Var name = "john" System.out.println(name.getClass().getSimpleName());</pre>
Java11	2018	- String methods isBlank() , lines(), strip() - Http Client API	<pre>String text ="java 11" Systemm.out.println(Text.strip());</pre>

Java Version	Year	Key Features	Example Code
Java 12	2019	- Switch Expressions (preview)	Int day = 3 ; String result = switch (day) { case 1,2,3 -> "weekday" ; default -> "weekend ;};
Java 13	2019	- Text blocks (preview)	String html = """"<html> \n <body> Hello World! </body> </html> """" ;
Java 14	2020	- Record keyword - Pattern matching for instance of	Record Person(String name , int age) {} Person p = new Person("Alice", 25);
Java 15	2020	- Seled classes	Sealed class Animal Permits Dog , Cat {} Final class Dog extends Anmial {} Final class Cat extends Animal{}
Java 16	2021	- Records (finalized)	Record Employee (String name , int age) {}
Java 17	2021	- Pattern Matching for switch	Object obj = "Hello" ; If { obj instanceof String} { System.out.println(s.toUpperCase());
Java18	2022	Simple Web Server	String text ="java 11" Systemm.out.println(Text.strip());
Java 19	2022	Virthal Thread	
Java 20	` 2023	More virtual thread enhacements	
Java 21	2023	Virtual Threads Recoprd Patterns & Pattern matching for switch	

LTS Version

Java Version	Release Year	End of Public Updates	Extended Support Until
Java 8 (LTS)	2014	December 2030	N/A
Java 11 (LTS)	2018	September 2026	September 2031
Java 17 (LTS)	2021	September 2029	September 2034
Java 21 (LTS)	2023	September 2031	September 2036

