WEEK-2

|  |
| --- |
| Introduction and software requirements for HTML/Javascript PHP programs: |

|  |
| --- |
| 1# What are the softwares that helps to run java programs. |

To run Java programs effectively, a combination of software tools is used. These tools aid in writing, compiling, debugging, and executing Java code. Below is a comprehensive explanation of the essential software components required to run Java programs:

**1. Java Development Kit (JDK)**

The **Java Development Kit (JDK)** is the core software required to develop and run Java programs.

* **What it includes:**
  + **Java Compiler (javac)**: Converts .java source files into bytecode (.class files).
  + **Java Virtual Machine (JVM)**: Executes the compiled bytecode.
  + **Java Runtime Environment (JRE)**: Required for running Java applications.
  + Other tools like javadoc, javap, jdb (debugger), etc.
* **Why it’s needed:** JDK is necessary for both **compiling** and **executing** Java code.
* **Available from:** Oracle, OpenJDK, Amazon Corretto, Eclipse Temurin, etc.
* **Installation Requirement:** Must be installed and properly configured (with JAVA\_HOME and system PATH).

**2. Java Runtime Environment (JRE)**

The **Java Runtime Environment (JRE)** provides the libraries, Java Virtual Machine (JVM), and other components to run applications written in Java.

* **Purpose:** It is used only for **running** Java programs, not for developing or compiling them.
* **Note:** JRE is part of the JDK package, so if the JDK is installed, the JRE is already included.

**3. Java Virtual Machine (JVM)**

The **Java Virtual Machine (JVM)** is a part of the JRE responsible for executing Java bytecode.

* **Function:** It provides platform independence by running the same compiled bytecode on different operating systems.
* **Working:** It interprets or compiles bytecode into machine code specific to the host operating system.

**4. Integrated Development Environments (IDEs)**

IDEs are powerful tools that make writing, compiling, debugging, and running Java programs much easier by combining all necessary tools in one interface.

**Commonly Used Java IDEs:**

| **IDE Name** | **Features** |
| --- | --- |
| **IntelliJ IDEA** | Advanced code suggestions, refactoring, debugging, GUI support |
| **Eclipse** | Widely used in enterprises, plugin support, project management |
| **NetBeans** | Supports GUI design, web applications, and academic purposes |
| **BlueJ** | Simple and educational, designed for beginners |
| **JGrasp** | Lightweight IDE with auto-visualization and simple interface |

* **Advantages of using IDEs:**
  + Auto-complete features
  + Syntax highlighting
  + Debugging tools
  + Integrated terminal and console
  + Error tracking

**5. Text Editors and Code Editors**

For basic Java programming, simple text editors can be used along with manual compilation via command line.

| **Editor Name** | **Description** |
| --- | --- |
| **Notepad (Windows)** | Very basic, no syntax highlighting |
| **Notepad++** | Lightweight, supports syntax highlighting and plugins |
| **Visual Studio Code (VS Code)** | Highly customizable with Java extensions and debugging support |

* These editors require using the command line to compile and run programs using javac and java.

**6. Command Line Tools**

The **Command Prompt (Windows)** or **Terminal (macOS/Linux)** is used to manually compile and run Java programs.

**Commands:**

* **Compile:** javac MyProgram.java
* **Run:** java MyProgram
* These tools are especially useful when working on simple programs or in academic environments.

**7. Online Java Compilers**

For quick testing or for users who cannot install Java locally, online compilers are available.

| **Website** | **Description** |
| --- | --- |
| **JDoodle** | Online compiler and executor with input support |
| **Programiz** | Beginner-friendly interface with explanations |
| **Replit** | Full IDE in the browser with file/project management |
| **OnlineGDB** | Online debugger and compiler with code sharing options |

|  |
| --- |
| 2# What is JDK and JRE. |

**🔷 JDK – Java Development Kit**

**Definition:**  
The **Java Development Kit (JDK)** is a complete software development kit used to develop, compile, debug, and run Java applications.

**Key Features:**

* It is a **superset of JRE** (Java Runtime Environment).
* It provides tools necessary for developing Java programs.

**Main Components of JDK:**

1. **Java Compiler (javac)** – Converts Java source code (.java files) into bytecode (.class files).
2. **Java Virtual Machine (JVM)** – Executes Java bytecode on your computer.
3. **Java Runtime Environment (JRE)** – Contains the libraries and environment to run Java programs.
4. **Development Tools** – Debuggers (jdb), document generator (javadoc), disassembler (javap), etc.

**Purpose:**  
Used by **Java developers** to write and build Java programs.

**Example Use:**

* Writing code in IntelliJ or Eclipse.
* Compiling with javac MyProgram.java.
* Running with java MyProgram.

**🔷 JRE – Java Runtime Environment**

**Definition:**  
The **Java Runtime Environment (JRE)** is a software package that provides the minimum requirements to **run** Java applications.

**Key Features:**

* Includes the **JVM** and **Java class libraries**.
* Does **not include the compiler** or development tools.

**Main Components of JRE:**

1. **JVM (Java Virtual Machine)** – Runs the bytecode.
2. **Class Libraries** – Pre-written Java classes (e.g., for GUI, file handling, networking).
3. **Other Supporting Files** – For memory management, security, and performance.

**Purpose:**  
Used by **end users** who only want to **run** Java applications, not develop them.

**Example Use:**

* Running Java games or applications without writing code.

|  |
| --- |
| 3# What is eclipse IDE. |

**Eclipse IDE** is a **free, open-source, and powerful Integrated Development Environment (IDE)** used for writing, compiling, debugging, and running Java programs. It is developed and maintained by the **Eclipse Foundation** and supports multiple programming languages through plugins, with Java being its most popular and widely supported language.

**Key Features of Eclipse IDE:**

1. **Code Editor**
   * Offers syntax highlighting, code completion, auto-formatting, and error detection.
2. **Compiler and Debugger Integration**
   * Allows you to compile Java programs with a single click.
   * Built-in debugger helps track down and fix bugs easily.
3. **Project Management**
   * Supports project-based development, helping you organize your files and code efficiently.
4. **Plugin Support**
   * Highly customizable. You can install plugins for other languages like C, C++, Python, PHP, etc.
5. **GUI Design Support**
   * You can build Java GUIs using plugins like **WindowBuilder**.
6. **Version Control Integration**
   * Built-in support for Git and other version control systems.
7. **Maven & Gradle Integration**
   * Easily manage external libraries and dependencies using build tools.

|  |
| --- |
| 4# How to run the java program in eclipse/netbeans IDE. |

**🔷 Running a Java Program in NetBeans IDE**

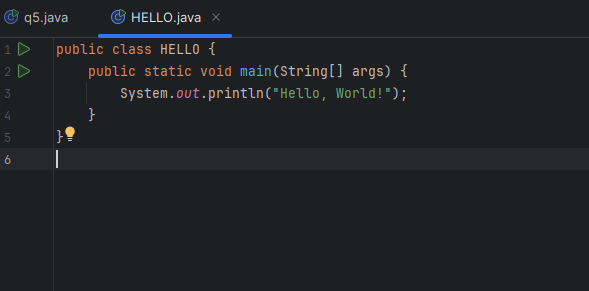
**1. Install JDK and NetBeans IDE**

* Download and install the **JDK**.
* Download **NetBeans IDE** from https://netbeans.apache.org/download/index.html.

1. **Open NetBeans**

* Launch the application.

1. **Create a New Java Project**
   * Go to **File → New Project**.
   * Choose **Java → Java Application**.
   * Click **Next**.
   * Enter the project name (e.g., MyFirstJavaApp).
   * Uncheck the box "Create Main Class" if you want to write it manually.
   * Click **Finish**.
2. **Create a Java Class**
   * Right-click on the Source Packages folder.
   * Select **New → Java Class**.
   * Give it a name (e.g., HelloWorld) and click **Finish**.
3. **Write Java Code**



1. **Run the Program**
   * Right-click on the file → **Run File**, or click the **green play button** on the top toolbar.
   * The output will be displayed in the **Output window** at the bottom.

Both **Eclipse** and **NetBeans IDE** make it easy to develop and run Java programs through graphical tools, automatic build systems, and console integration. While Eclipse is widely used in professional settings, NetBeans is especially beginner-friendly and ideal for students and learners.