Xebia

Environment Setup

Installing JDK and JRE

- **Download:**
  - Go to the Oracle or OpenJDK website.
  - Download the latest version of the JDK for your operating system (e.g., Windows, macOS, Linux).
- **Run Installer:**
  - Execute the downloaded installer.
  - Follow the installation wizard's instructions.
  - Specify the installation directory (e.g., C:\Program Files\Java\jdk1.x.x_xx).
- **Set Environment Variables (Optional):**
  - Set the JAVA_HOME environment variable to point to the JDK installation directory.
  - Add the JDK's "bin" directory to your system's PATH variable.

Installing Intellij IDE

- **Download:**
  - Visit the JetBrains IntelliJ IDEA download page.
  - Download the Community (free) or Ultimate (paid) edition based on your requirements.
- **Run Installer:**
  - Execute the downloaded installer.
  - Follow the installation wizard's instructions.
  - Specify the installation directory (e.g., C:\Program Files\JetBrains\IntelliJ IDEA).
- **Initial Configuration:**
  - Launch IntelliJ IDEA after installation.
  - Configure your preferred settings, such as theme, keymap, and plugins, during the initial setup.
- **Create or Open a Project:**
  - Create a new project or open an existing one.
  - Ensure that you select the appropriate JDK version when creating a new project.
- **Configure JDK in IntelliJ IDEA (if necessary):**
  - If IntelliJ IDEA does not automatically detect your JDK installation, configure it manually:
  - Go to "File" > "Project Structure."
  - Under "Project," select the Project SDK (JDK) from the dropdown.
  - Add the JDK if it's not listed by clicking the "New" button.

Java Program Development

- Install JDK and IDE.
- Create a New Java Project.
- Write Java Code.
- Compile Java Code.
- Debugging (Optional).
- Run Your Java Program.
- Review Output.
- Test and Iterate.
- Version Control (Optional).
- Documentation and Comments.
- Packaging and Deployment.
- Maintain Codebase.
- Learn and Improve.

Java Source File Structure

- **Package Declaration (Optional)**
- **Import Statements (Optional)**
  - **Class Declaration**
  - Access Modifier (public, private, etc.)
  - class Keyword
  - Class Name
  - **Class Body**
  - Fields (Optional)
  - Constructors (Optional)
  - Methods (Optional)
  - Inner Classes (Optional)
  - **Main Method (Optional)**
  - public static void main(String[] args)
- **Closing Curly Brace** }

Compilation

- Source Code: Java source code is written in .java files.
- Compiler: The Java compiler (javac) translates source code into bytecode.
- Compilation Process:
  - Lexical Analysis
  - Syntax Analysis
  - Semantic Analysis
  - Bytecode Generation
- Bytecode: It is a platform-independent representation of Java source code.
- Compilation Errors: Errors in the source code are reported during compilation.
- Compilation Output: Successful compilation generates .class files.
- Separate Compilation: Java allows individual source files to be compiled independently.
- JAR Files: Compiled classes can be packaged into JAR files for distribution.
- Runtime Execution: Bytecode is executed by the JVM, which interprets or compiles it into native code.

Executions

- Code Execution: Running a program or script to perform tasks.
- Execution Flow: The sequence in which program statements are executed.
- Exception Handling: Managing the flow of execution when exceptions occur.
- Parallel Execution: Concurrent execution of tasks for improved performance.

xebia.com