

How To Manual

Recommended OS: Windows 11 (64-bit)

This manual walks you through installing and configuring the core tools needed for a software engineering course project:

- Java (JDK)
- Apache Tomcat 9.1.xx
- Maven
- Node.js (v20+)
- JavaScript runtime & tooling
- Code editor (VS Code or Eclipse)
- Common post-installation configurations

1. System Prerequisites

Before starting, ensure:

- Windows 11 (64-bit)
- At least 8 GB RAM (16 GB recommended)
- Administrator access
- Stable internet connection

2. Install Java (JDK)

2.1 Download

- Install Java Development Kit (JDK) 17 or newer (LTS recommended)

2.2 Installation Steps

1. Run the installer
2. Use default installation path (recommended), E
Example: C:\Program Files\Java\jdk-17

2.3 Environment Variables

1. Open **System Properties** → **Advanced** → **Environment Variables**
2. Under **System Variables**, add: JAVA_HOME = JDK installation path

3. Edit Path and add: %JAVA_HOME%\bin

2.4 Verify Installation

Open Command Prompt:

java -version

javac -version

3. Install Apache Maven

3.1 Download

- Download Apache Maven 3.9+ (binary zip)

3.2 Installation Steps

1. Extract zip to: C:\Program Files\Apache\maven

3.3 Environment Variables

1. Add system variable: MAVEN_HOME = Maven directory
2. Add to Path: %MAVEN_HOME%\bin

3.4 Verify Installation

mvn -version

4. Install Apache Tomcat 9.1.xx

4.1 Download

- Download Apache Tomcat 9.1.xx (Windows zip)

4.2 Installation Steps

1. Extract to: C:\Program Files\Apache\tomcat9

4.3 Configure Tomcat

Set JAVA_HOME

- Ensure JAVA_HOME is already configured (required for Tomcat)

Optional: Set CATALINA_HOME

- Add system variable:
 - CATALINA_HOME = Tomcat directory

4.4 Start / Stop Tomcat

- Run:
bin\startup.bat
bin\shutdown.bat

4.5 Verify Installation

Open browser: <http://localhost:8080>

5. Install Node.js (v20 or higher)

5.1 Download

- Install **Node.js LTS (20+)**

5.2 Installation Steps

1. Run installer
2. Enable **npm** during installation

5.3 Verify Installation

```
node -v  
npm -v
```

6. JavaScript Tooling

JavaScript is included with Node.js. Common tools:

6.1 Install Common Packages

```
npm install -g npm@latest  
npm install -g nodemon
```

6.2 Verify

```
nodemon -v
```

7. Install Code Editor

Option A: Visual Studio Code (Recommended)

Installation

- Install **VS Code (64-bit)**

Recommended Extensions

- Java Extension Pack
- Maven for Java
- ESLint
- Prettier
- Node.js Extension Pack

Verify Java Integration

- Open a `.java` file
- Confirm IntelliSense and build support

Option B: Eclipse IDE

Installation

- Install Eclipse IDE for Enterprise Java and Web Developers

Configuration

- Set JDK path: Window → Preferences → Java → Installed JREs
- Configure Maven: Preferences → Maven → Installations

8. Post-Installation Configurations

8.1 Git (Recommended)

`git --version`

8.2 Maven Settings (Optional)

- Configure `settings.xml` for proxies or repositories

8.3 Tomcat User Configuration

Edit `conf/tomcat-users.xml`:

```
<role rolename="manager-gui"/>
<user username="admin" password="password" roles="manager-gui"/>
```

Access Manager: `http://localhost:8080/manager`

8.4 Firewall / Port Check

- Ensure port **8080** is allowed

8.5 Project Folder Structure

Recommended:

```
project-root/
├── backend/ (Java, Maven)
├── frontend/ (Node.js, JS)
└── docs/
```

9. Common Issues & Fixes

Issue	Solution
java not recognized	Check JAVA_HOME and Path
Tomcat fails to start	Ensure no other app uses port 8080
Maven build fails	Run <code>mvn clean install</code>
Node permission errors	Run terminal as Administrator

10. Verification Checklist

- Java installed and configured
- Maven working
- Tomcat running on localhost

- Node.js v20+ installed
- Code editor configured
- Sample project builds successfully

11. Conclusion

This setup provides a complete development environment using Java, JavaScript, Maven, Node.js, and Apache Tomcat on Windows 11. These are what were used during the for our project.

End of Manual