



Java Foundations

1-3

Setting Up Java

ORACLE
Academy



Copyright © 2020, Oracle and/or its affiliates. All rights reserved.

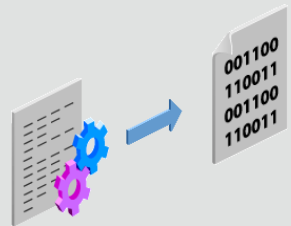
Objectives

- This lesson covers the following objectives:
 - Understand the difference between the JDK and JRE
 - Understand the difference between .java and .class files
 - Describe the purpose of an integrated development environment (IDE)
 - Download and install the JDK, JRE, and NetBeans IDE
 - Import a project into NetBeans

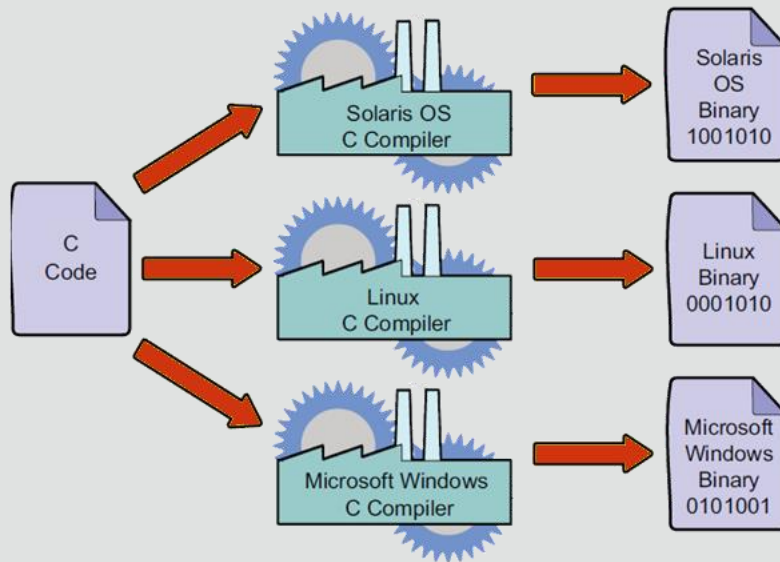


Purpose of a Computer Program

- A computer program is a set of instructions that run on a computer or other digital device
- At the machine level, the program consists of binary instructions (1s and 0s)
 - Machine code
- Most programs are written in high-level code (readable)
 - Must be translated to machine code



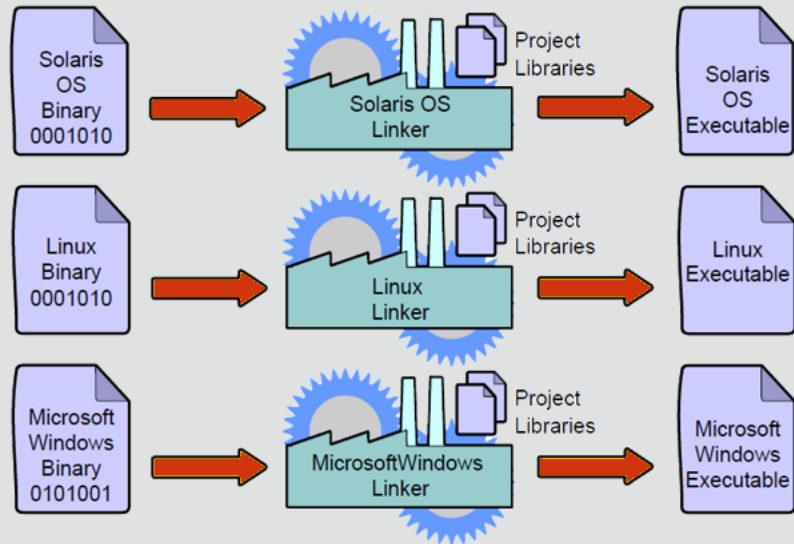
Translating High-Level Code to Machine Code



Programs written in most languages usually require numerous modifications to run on more than one type of computing platform (a combination of a CPU and an operating system). That's because most languages require you to write code specific to the underlying platform. Popular programming languages like C and C++ require programmers to compile and link their programs, resulting in an executable program that's unique to a platform. A compiler is an application that converts a program that you write into a CPU-specific code called *machine code*. These platform-specific files (binary files) are often combined with other files, such as libraries of prewritten code. And a linker creates a platform-dependent program, called an *executable*, that an end user can execute. Unlike C and C++, the Java programming language is platform-independent.

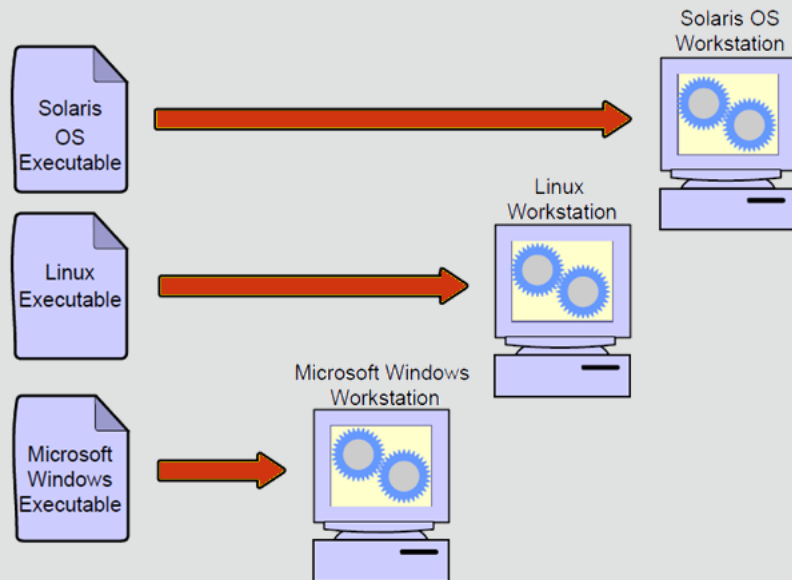
The image illustrates how a compiler creates a binary file.

Linked to Platform-Specific Libraries



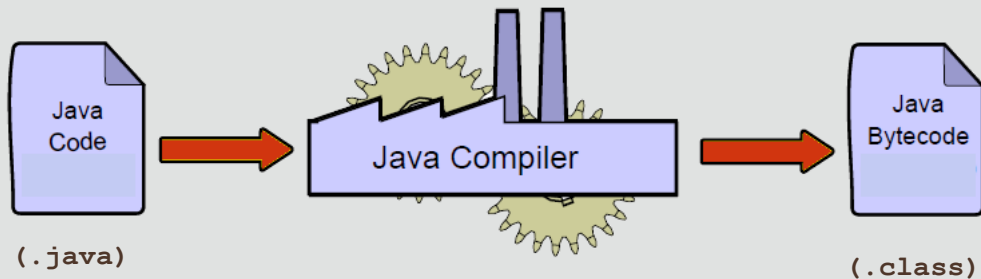
The image illustrates how a binary file is linked with libraries to create a platform-dependent executable.

Platform-Dependent Programs



The image illustrates how platform-dependent executables can execute only on one platform.

Java Is Platform-Independent

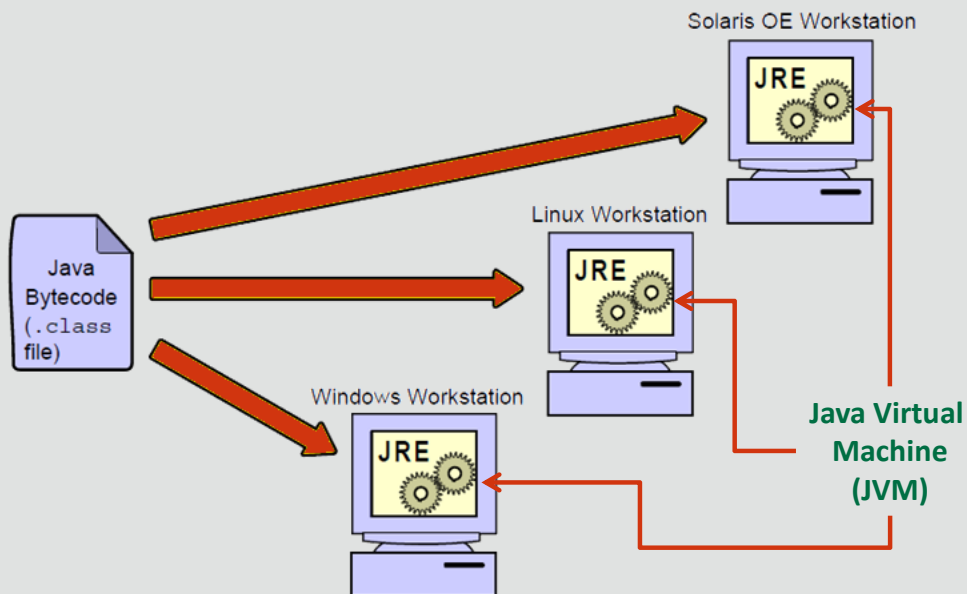


A Java program can run on several different CPUs and operating system combinations, such as the Solaris OS on a SPARC chip, MacOS X on an Intel chip, and Microsoft Windows on an Intel chip, usually with few or no modifications.

As illustrated in the image, Java programs are compiled with a Java compiler. The resulting format of a compiled Java program is platform-independent Java bytecode instead of CPU-specific machine code.

The created bytecode is interpreted by a bytecode interpreter called the Java Virtual Machine (JVM). A virtual machine is a platform-specific program that understands platform-independent bytecode and can execute it on a particular platform. For this reason, the Java programming language is often referred to as an interpreted language, and Java technology programs are said to be portable or executable on any platform. Another interpreted language is Perl.

Java Programs Run in a JVM



The image illustrates a Java bytecode file executing on several platforms where a Java runtime environment exists.

A virtual machine gets its name because it's a piece of software that runs code, a task usually accomplished by the CPU or hardware machine. For Java programs to be platform-independent, the JVM is required on every platform where your program will run. The JVM is responsible for interpreting Java code, loading Java classes, and executing Java programs.

However, a Java program needs more than just a JVM to execute. It also needs a set of standard Java class libraries for the platform. Java class libraries are libraries of prewritten code that can be combined with the code that you write to create robust applications.

Combined, the JVM software and Java class libraries are referred to as the **Java Runtime Environment (JRE)**. JREs are available from Oracle for many common platforms.

Java Runtime Environment (JRE)

- Includes:

- The Java Virtual Machine (JVM)
- Java class libraries

- Purpose:

- Read bytecode (.class)
- Run the same bytecode anywhere with a JVM



JRE

Java Development Kit (JDK)

- Includes:

- JRE
- Java Compiler
- Additional tools



JRE



JDK

- Purpose:

- Compile bytecode (.java → .class)

Integrated Development Environment (IDE)

- Purpose:

- Provide a sophisticated text editor
- Offer assistance debugging code
- Manage projects
- Write source code (.java)

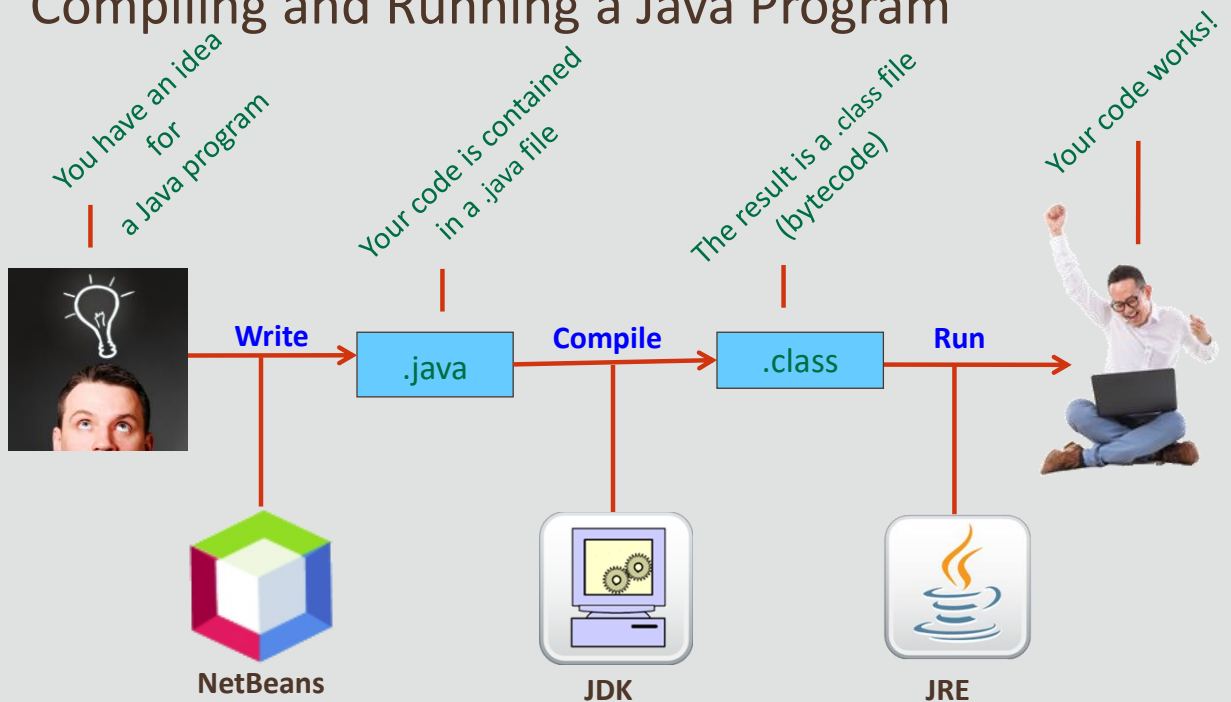
- Examples:

- NetBeans
- Greenfoot and BlueJ
- Alice



NetBeans IDE

Compiling and Running a Java Program



ORACLE
Academy

JFo 1-3
Setting Up Java

Copyright © 2020, Oracle and/or its affiliates. All rights reserved.

13

The diagram shows what happens when you compile and run a Java program:

You have an idea for a Java program.

The Java code is written in NetBeans, and the file has a .java extension. This is called the “Java source code.”

The compiler component of the JDK compiles the source code into a bytecode file with a .class extension. This is called a Java class.

The JVM component of the JRE runs the Java class. This is your Java program.

Celebrate triumphantly when your code works, because most of the time it won’t work.

This diagram oversimplifies the debugging aspect of development.

Time to Set Up!



JRE



JDK



NetBeans IDE

Downloading JDK 8 & NetBeans

1. Go to the Java SE Downloads page at <https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
2. Locate latest Java SE Development Kit

Java SE Development Kit 8u191		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
<input type="radio"/> Accept License Agreement <input checked="" type="radio"/> Decline License Agreement		
Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	72.97 MB	jdk-8u191-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	69.92 MB	jdk-8u191-linux-arm64-vfp-hflt.tar.gz
Linux x86	170.89 MB	jdk-8u191-linux-i586.rpm
Linux x86	185.69 MB	jdk-8u191-linux-i586.tar.gz
Linux x64	167.99 MB	jdk-8u191-linux-x64.rpm
Linux x64	182.87 MB	jdk-8u191-linux-x64.tar.gz
Mac OS X x64	245.92 MB	jdk-8u191-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	133.04 MB	jdk-8u191-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	94.28 MB	jdk-8u191-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	134.04 MB	jdk-8u191-solaris-x64.tar.Z
Solaris x64	92.13 MB	jdk-8u191-solaris-x64.tar.gz
Windows x86	197.34 MB	jdk-8u191-windows-i586.exe
Windows x64	207.22 MB	jdk-8u191-windows-x64.exe

You'll need Java SE Development Kit 8 (JDK 8) for this course. The version numbers of the download may be different as they are frequently updated

Downloading JDK 8 & NetBeans

3. Accept the license agreement
4. Select the correct download for your system

Java SE Development Kit 8u191

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software

3 ☒ Accept License Agreement ☐ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	72.97 MB	jdk-8u191-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	69.92 MB	jdk-8u191-linux-arm64-vfp-hflt.tar.gz
Linux x86	170.89 MB	jdk-8u191-linux-i586.rpm
Linux x86	185.69 MB	jdk-8u191-linux-i586.tar.gz
Linux x64	167.99 MB	jdk-8u191-linux-x64.rpm
Linux x64	182.87 MB	jdk-8u191-linux-x64.tar.gz
Mac OS X x64	245.92 MB	jdk-8u191-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	133.04 MB	jdk-8u191-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	94.28 MB	jdk-8u191-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	134.04 MB	jdk-8u191-solaris-x64.tar.Z
Solaris x64	92.13 MB	jdk-8u191-solaris-x64.tar.gz
Windows x86	197.34 MB	jdk-8u191-windows-i586.exe
Windows x64	207.22 MB	jdk-8u191-windows-x64.exe

Downloading JDK 8 & NetBeans

5. Save the .exe file to your computer, noting the location



Downloading JDK 8 & NetBeans

1. . Go to the Netbeans Downloads page at <https://netbeans.org/downloads/8.2/>
2. Download Java SE

NetBeans IDE 8.2 Download

8.1 | 8.2 | Development | Archive

Email address (optional):

Subscribe to newsletters: ☒ Monthly ☐ Weekly ☒ NetBeans can contact me at this address

IDE Language: English Platform: Windows

Note: Greyed out technologies are not supported for this platform.

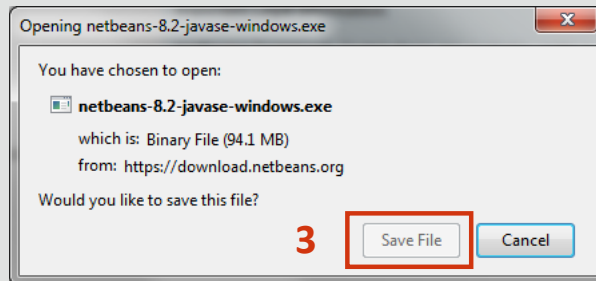
Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME		•				•
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						•
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

Download bundles:

Download	Download	Download x86	Download x86	Download x86	Download
Free, 95 MB	Free, 197 MB	Free, 108 - 112 MB	Free, 108 - 112 MB	Free, 107 - 110 MB	Free, 221 MB

Downloading JDK 8 & NetBeans

3. Save the .exe file to your computer, noting the location

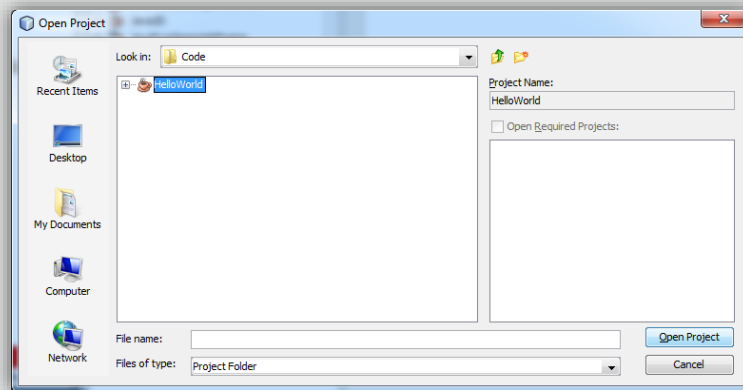


Installing JDK 8 & NetBeans

1. Locate and run the JDK installer file on your computer to start the installer
2. Follow on-screen instructions to complete the installation
3. Locate and run the NetBeans installer file on your computer to start the installer
4. Follow on-screen instructions to complete the installation

Testing NetBeans

1. Download and unzip the HelloWorld project
2. Launch NetBeans
3. Select File > Open Project, and then select HelloWorld

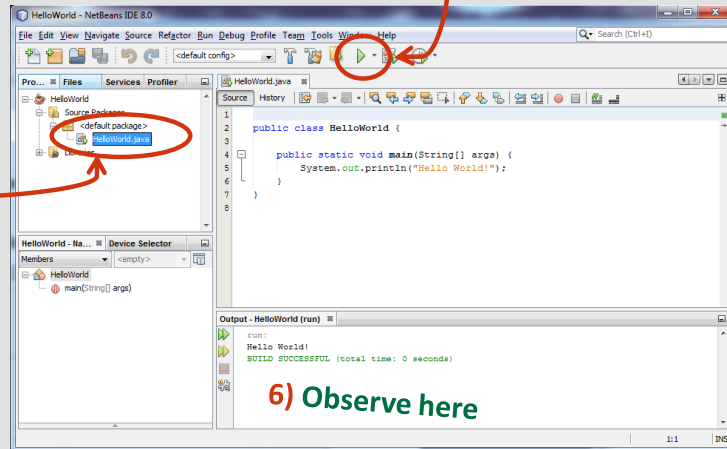


Testing NetBeans

4. Expand the project and open HelloWorld.java
5. Compile and run the project
6. Observe the output

4) Double-click here

5) Click here



6) Observe here



Exercise

- Edit the code so that it prints a message other than **"Hello World!"**
 - Compile and run your code to verify that your changes work
- Change the 'S' in "System" to lowercase and try to compile it
 - Is Java syntax case-sensitive?
- Remove the semicolon (;) and try to compile
 - Is the semicolon important to Java syntax?

Note to Instructors: Print statements are formally introduced in Section 2. The goal of this exercise is to check that everything is installed properly. The exercise also offers students a chance to learn through experimentation, which should make them more receptive when print statements are discussed later.

Summary

- A computer program is written in a high-level language, but must be compiled into machine code
- Most programming languages compile a separate executable for each platform
- Java is platform-independent



An IDE like NetBeans is used to **write** source code (`.java`)



The JDK **compiles** bytecode (`.java` → `.class`)



Bytecode **runs** in a JVM, which is part of the JRE

Summary

- In this lesson, you should have learned how to:
 - Understand the difference between the JDK and JRE
 - Understand the difference between .java and .class files
 - Describe the purpose of an IDE
 - Download and install the JDK, JRE, and NetBeans IDE
 - Import a project into NetBeans



The Oracle Academy logo is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, dark gray, sans-serif font. The entire logo is framed by a thin black border, with dark gray horizontal bars at the top and bottom.

ORACLE

Academy