

What Is a Java Program?



Objectives

After completing this lesson, you should be able to:

- Contrast the terms “platform-dependent” and “platform-independent”
- Describe the purpose of the JVM
- Explain the difference between a procedural program and an object-oriented program
- Describe the purpose of `javac` and `java` executables
- Verify the Java version on your system
- Compile and run a Java program from the command line



Topics

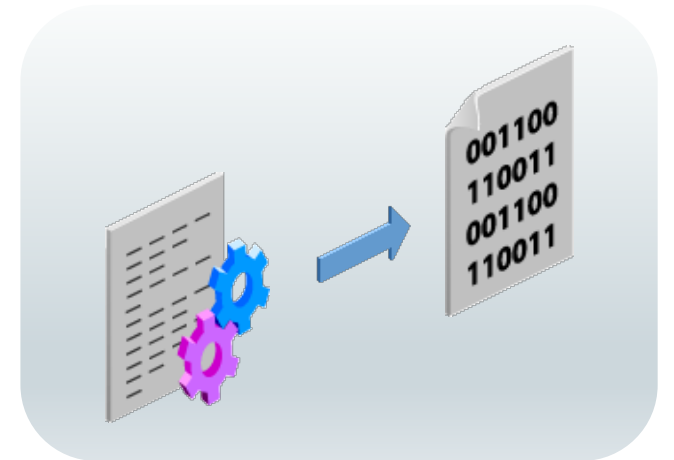
- Introduction to computer programs
- Introduction to the Java language
- Verifying the Java development environments
- Running and testing a Java program



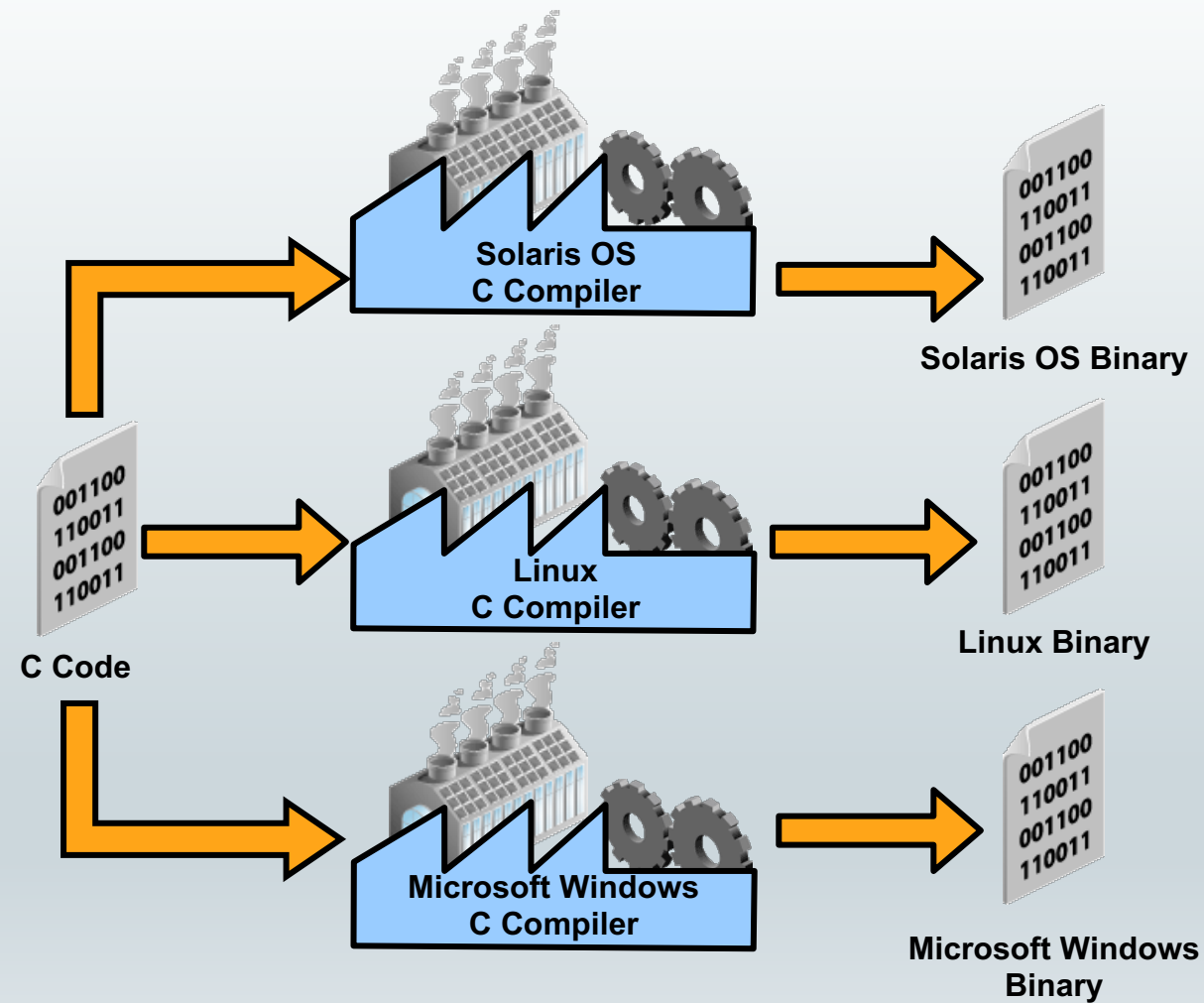
Purpose of a Computer Program

A computer program is a set of instructions that run on a computer or other digital device.

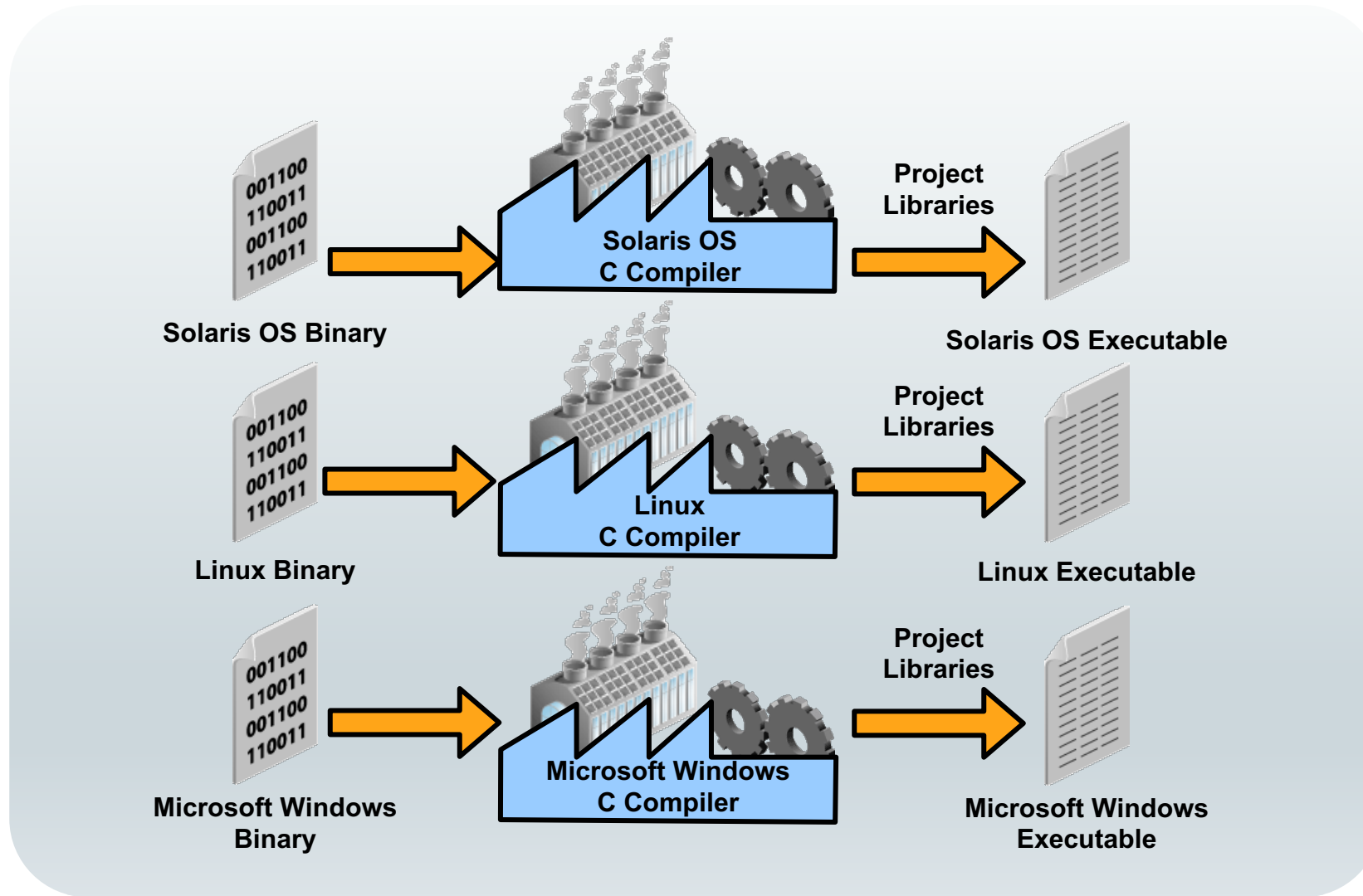
- At the machine level, the program consists of binary instructions (1s and 0s).
 - Machine code
- Most programs are written in *high-level* code (readable).
 - Must be translated to machine code



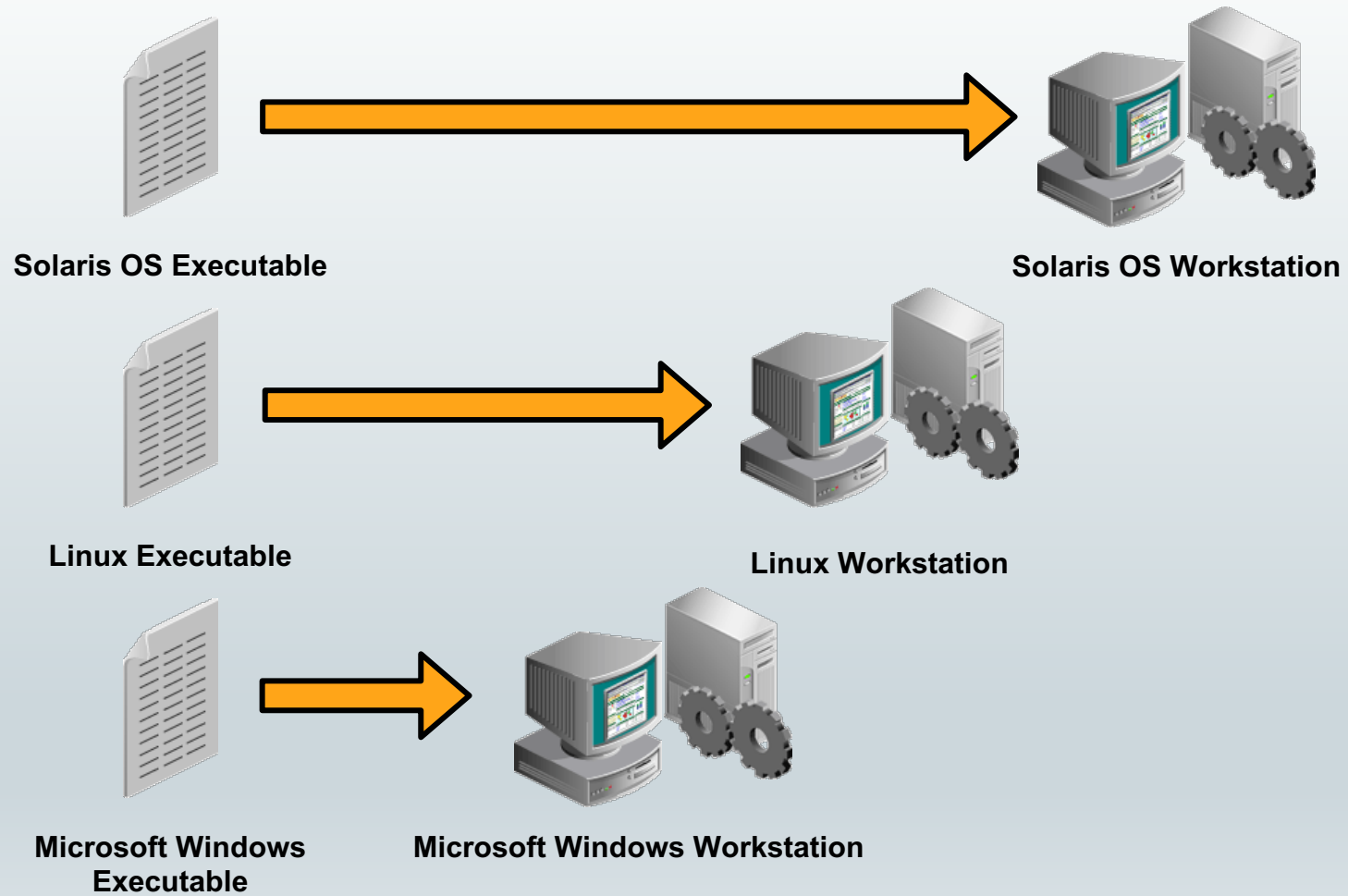
Translating High-Level Code to Machine Code



Linked to Platform-Specific Libraries



Platform-Dependent Programs



Topics

- Introduction to computer programs
- **Introduction to the Java language**
- Verifying the Java development environment
- Running and testing a Java program

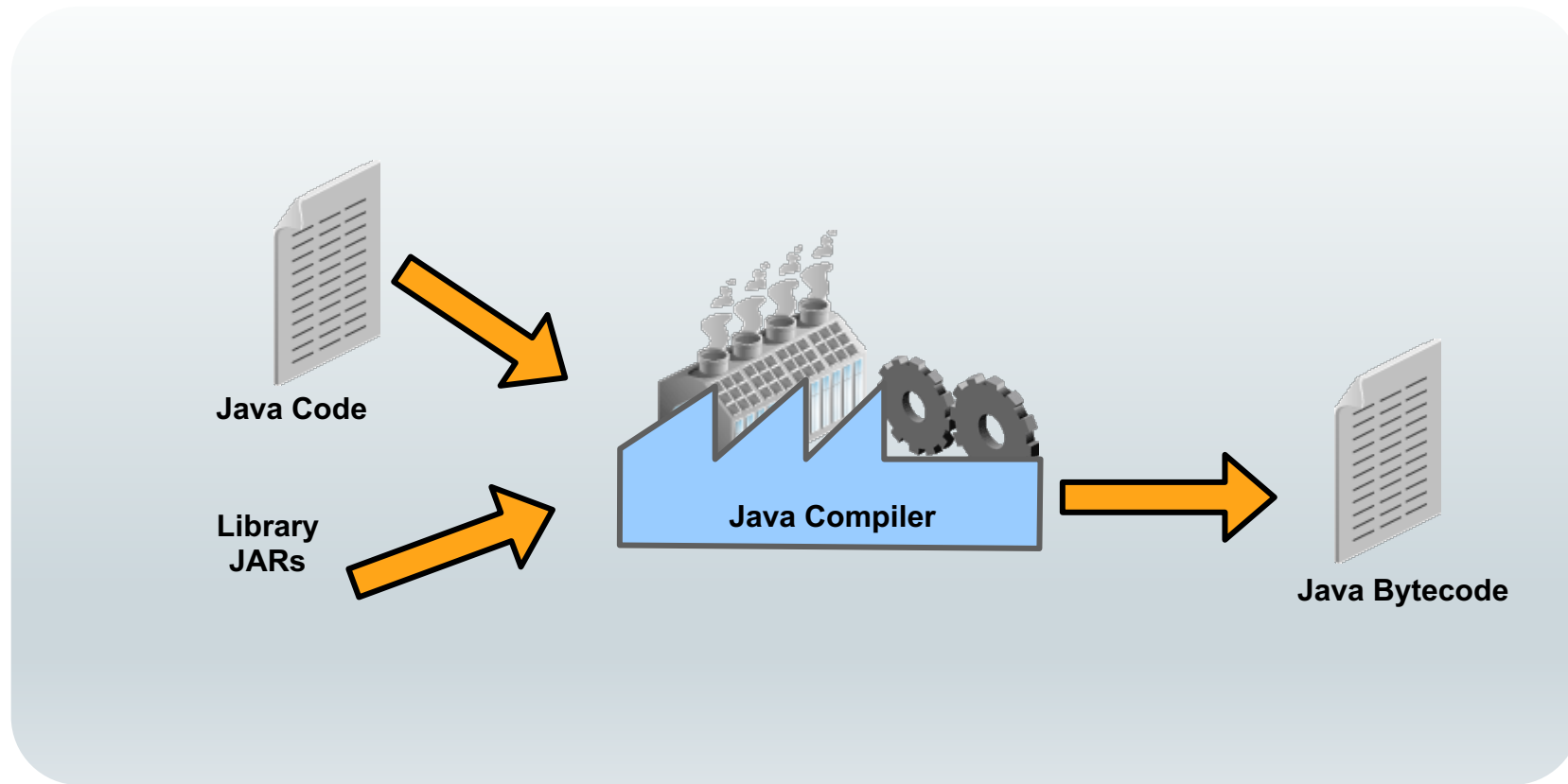


Key Features of the Java Language

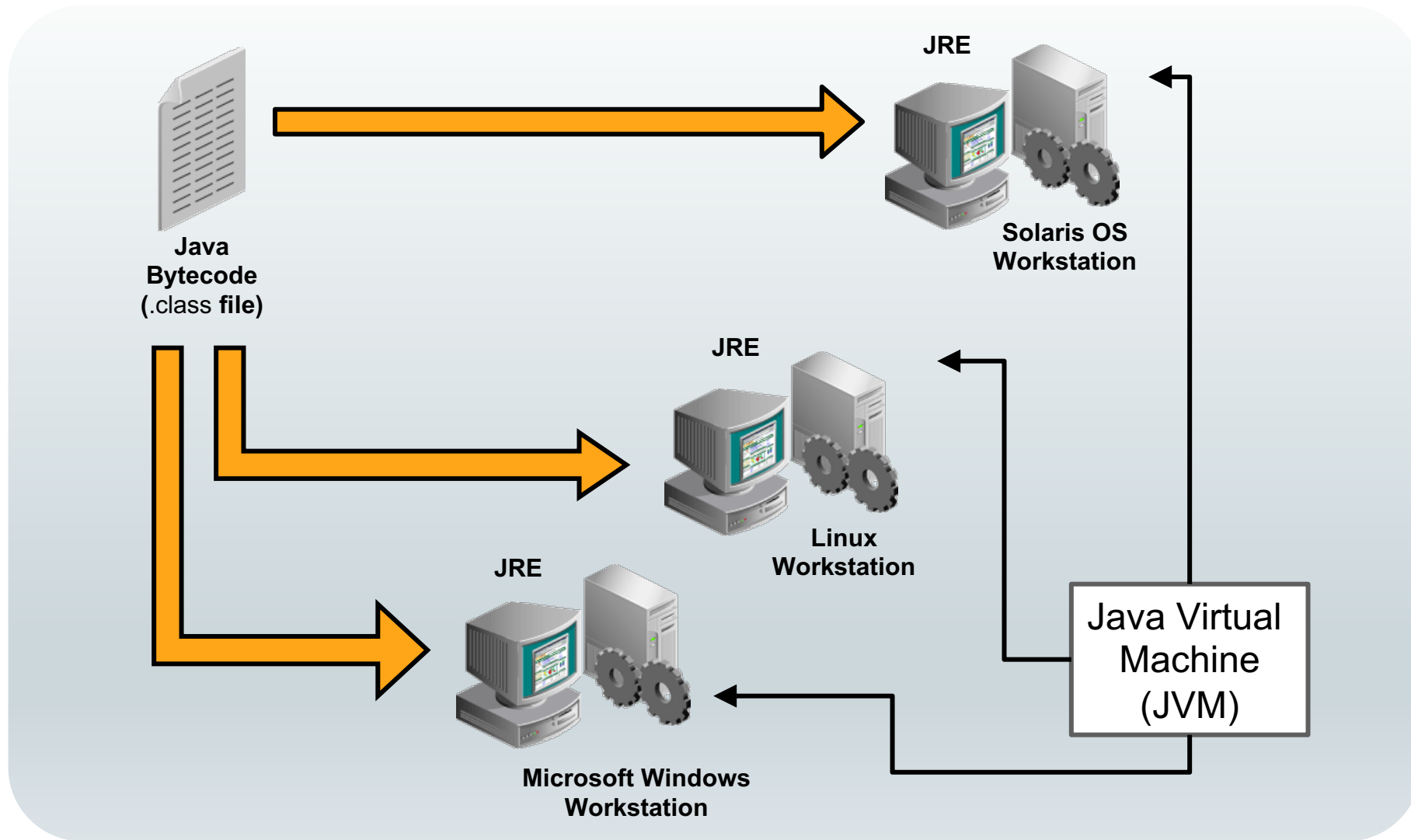
Some of the features that set Java apart from most other languages are that:

- It is platform-independent
- It is object-oriented

Java Is Platform-Independent

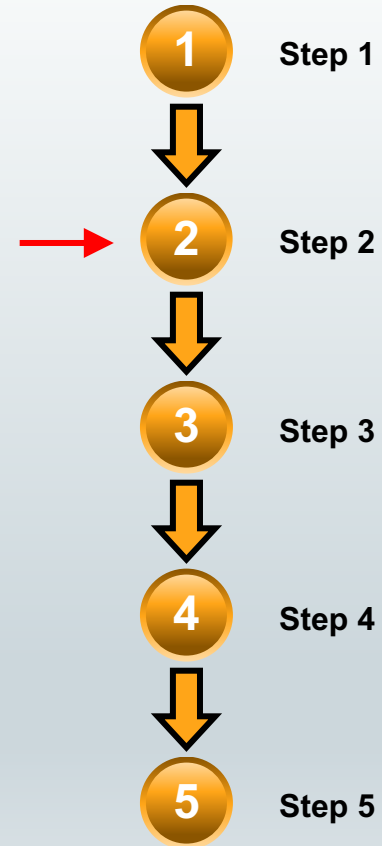


Java Programs Run In a Java Virtual Machine



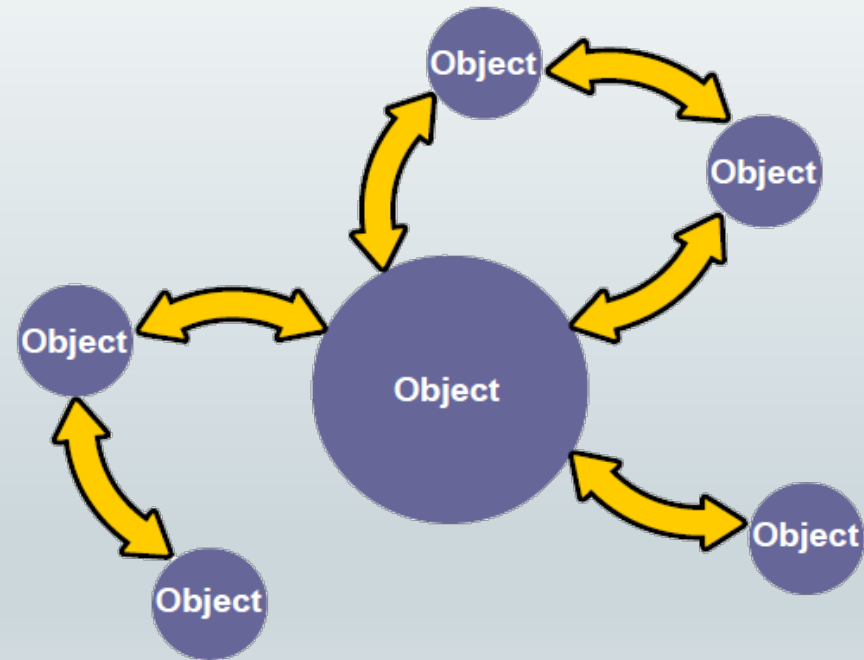
Procedural Programming Languages

- Many early programming languages followed a paradigm called *Procedural Programming*.
- These languages use a sequential pattern of program execution.
- Drawbacks to procedural programming:
 - Difficult to translate real-world use cases to a sequential pattern
 - Difficult to maintain programs
 - Difficult to enhance as needed



Java Is an Object-Oriented Language

- Interaction of objects
- No prescribed sequence
- Benefits:
 - Modularity
 - Information hiding
 - Code reuse
 - Maintainability



Topics

- Introduction to computer programs
- Introduction to the Java language
- **Verifying the Java development environment**
- Running and testing a Java program

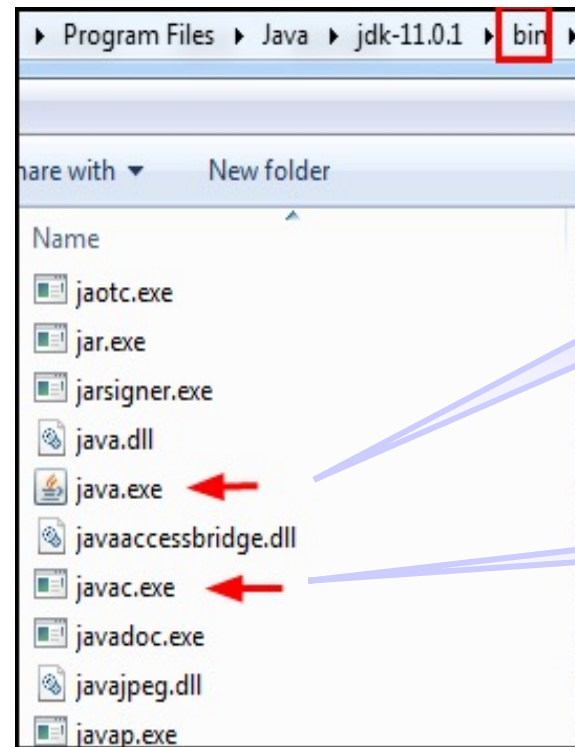
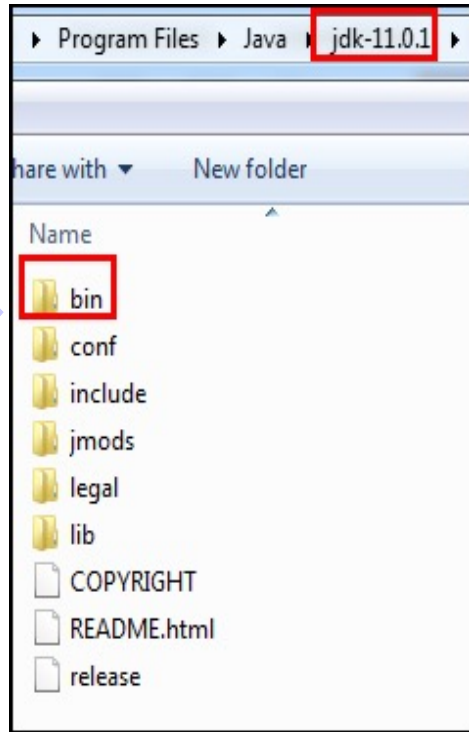


Verifying the Java Development Environment

1. Download and install the Java Development Kit (JDK) from oracle.com/java.
2. Examine the environment.
3. Compile and run a Java application by using the command line.

Examining the Installed JDK: The Tools

PATH points
here



Runtime

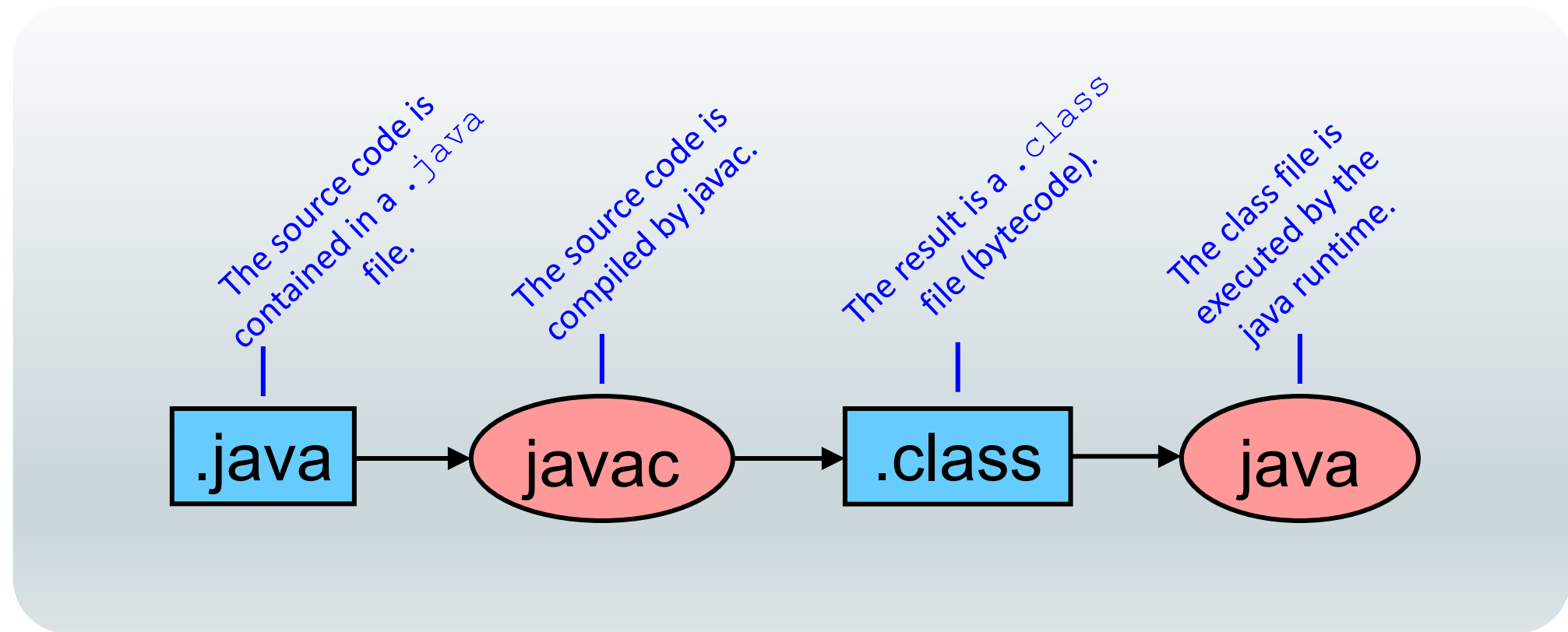
Compiler

Topics

- Introduction to computer programs
- Introduction to the Java language
- Verifying the Java development environment
- Running and testing a Java program



Compiling and Running a Java Program



Compiling a Program

1. Go to the directory where the source code files are stored.
2. Enter the following command for each `.java` file you want to compile.

- Syntax:

```
javac <source file>
```

- Example:

```
javac SayHello.java
```

Executing (Testing) a Program

1. Go to the directory where the class files are stored.
2. Enter the following for the class file that contains the main method:

- Syntax:

```
java <classname>
```

- Example:

Do **not** specify `.class`.

```
java SayHello
```

- Output:

```
Hello World!
```

Output for a Java Program

A Java program can output data in many ways. Here are some examples:

- To a file or database
- To the console
- To a webpage or other user interface

JDK 11 new feature: Launch Single-File Source-Code Programs

Benefits:

- Skip the compilation "ceremony".
- Run a program with one quick command:

```
java <source file>
```

Requirements:

- Write the entire program as single source file.
- The file may contain any number of classes.
- The top-most class declares a `main` method.

Use Cases:

- Experiment quickly to learn Java.
- Write small utility or "shebang" files.

Circle.java

```
public class Test {  
    public static void main(String args[]) {  
        double area = Circle.findArea(7.5);  
        System.out.print("Area of circle=" +area);  
    }  
}  
  
public class Circle {  
    public static double findArea(double radius){  
        return Math.PI * radius * radius;  
    }  
}
```

```
java Circle.java
```

```
Area of circle=176.714...
```

Quiz



Which of the following is correct? (Choose all that apply.)

- a. `javac OrderClass`
- b. `java OrderClass`
- c. `javac OrderClass.java`
- d. `java OrderClass.java`



Summary

In this lesson, you should have learned how to:

- Describe the distinction between high-level language and machine code
- Describe what platform-independence means
- Describe how a Java program is compiled and to what format
- Explain what it means to say that Java is an object-oriented language
- Determine the version number of a Java install
- Compile and run a Java program from the command line

