



# Java Puzzle Ball

Nick Ristuccia

Lesson 0

What is Java?



# Lesson 0 is Optional

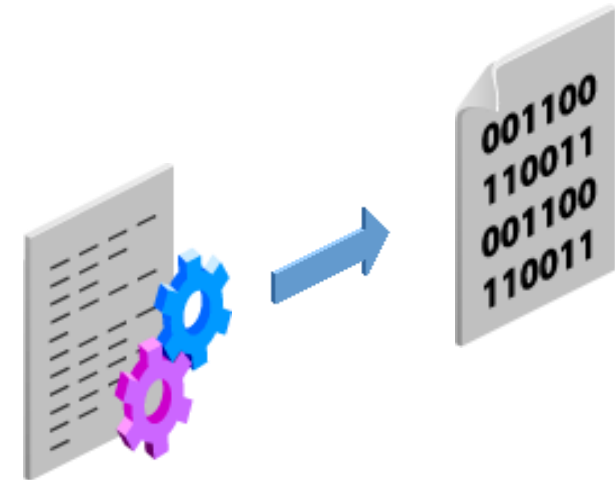
- Lesson 1 is where the real fun starts!
  - But you'll need Java 8 or higher installed to run Java Puzzle Ball.
- Lesson 0 gives an overview of Java and what you'll be installing.
  - But you won't need to know these technical details for this course.
  - You can skip Lesson 0 and go right to Lab 0.
- Lab 0 gives installation instructions so that you can...
  - Play Java Puzzle Ball.
  - Edit Java code.



# Purpose of a Computer Program

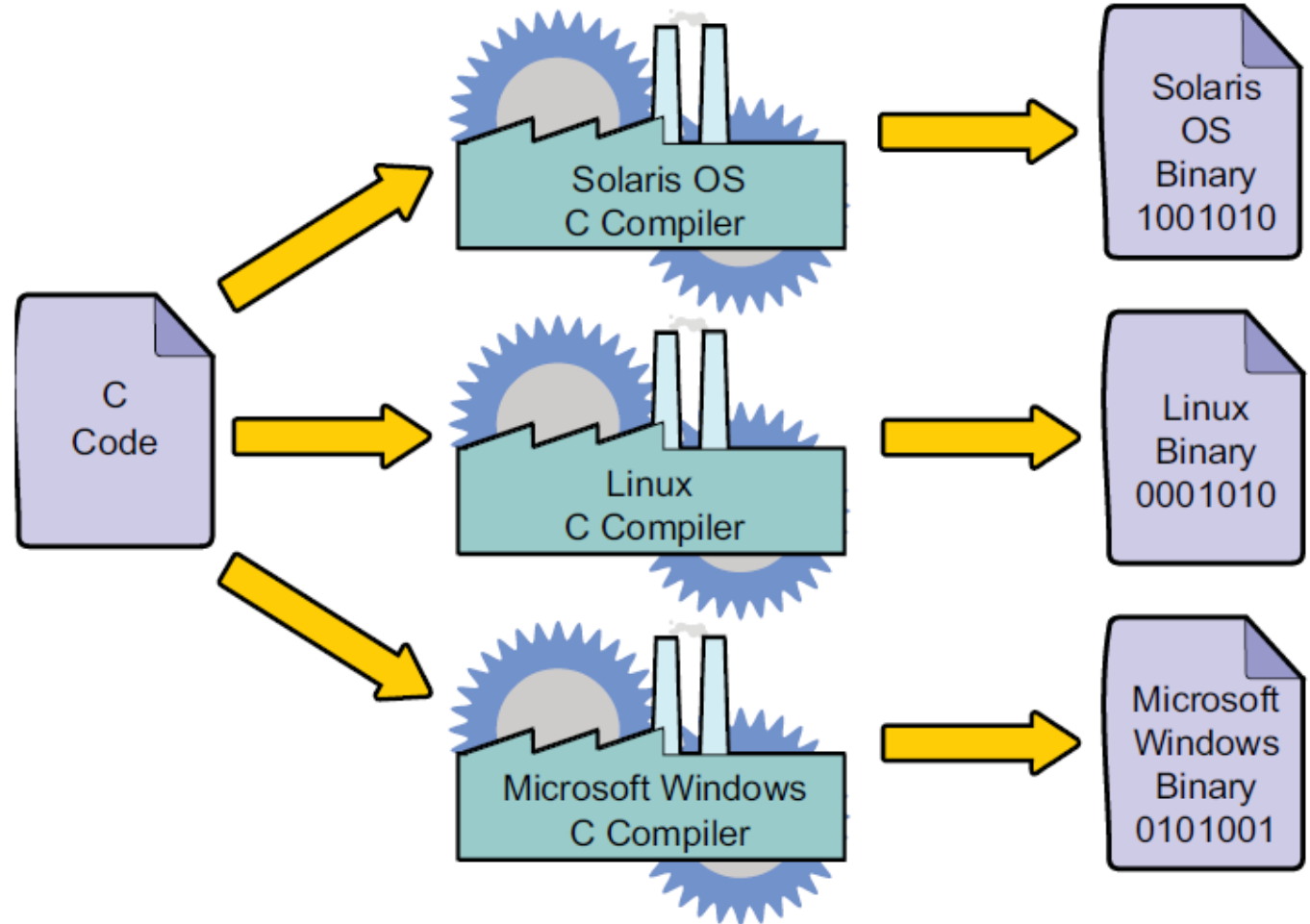
A computer program is a set of instructions that run on a computer or other digital device.

- 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

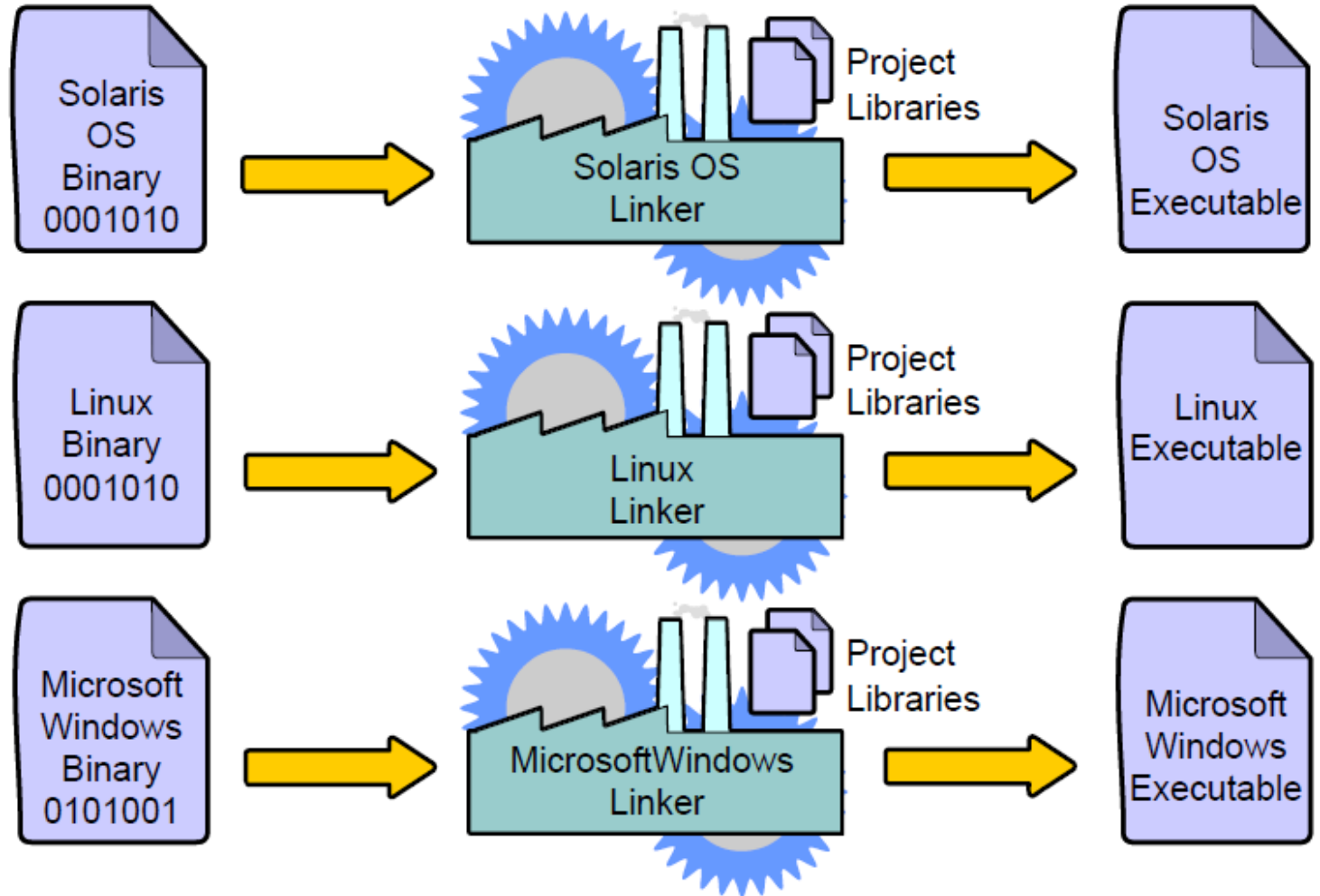
A compiler converts your code into a CPU-specific code called *machine code*. These are binary files, which are platform-specific.



# Linked to Platform-Specific Libraries

A binary file is linked with libraries to create a platform-dependent executable.

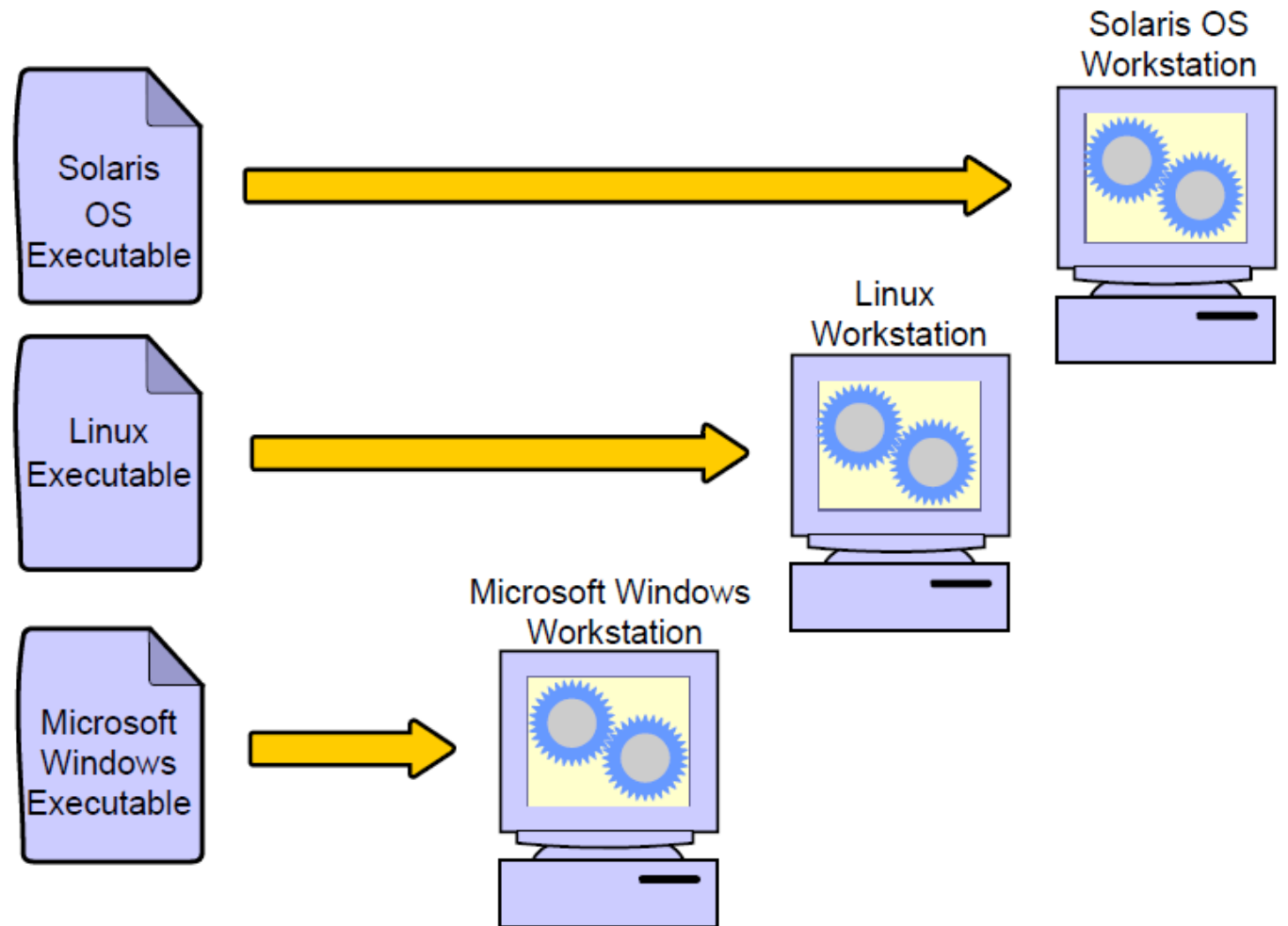
For example, a Windows binary only makes sense to Windows machines.



# Platform-Dependent Programs

Platform-dependent executables work on only one platform. For example, .exe's for Windows.

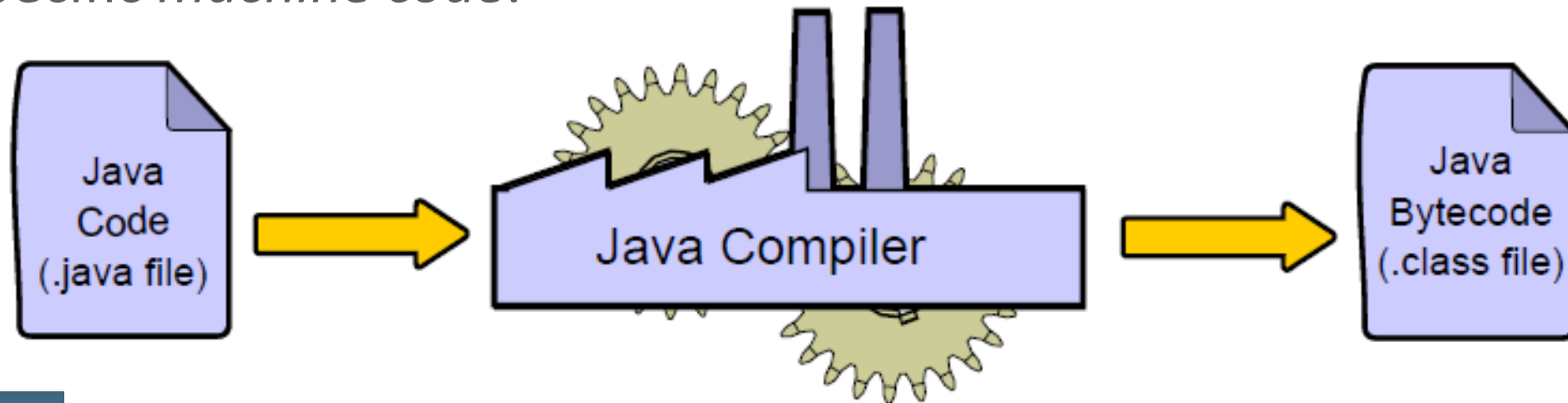
Programs written in most languages usually require numerous modifications to run on more than one type of platform. Programmers don't like this extra work.



# Java Is Platform-Independent

A Java program can run on several different CPUs and operating system combinations, usually with few or no modifications.

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*.

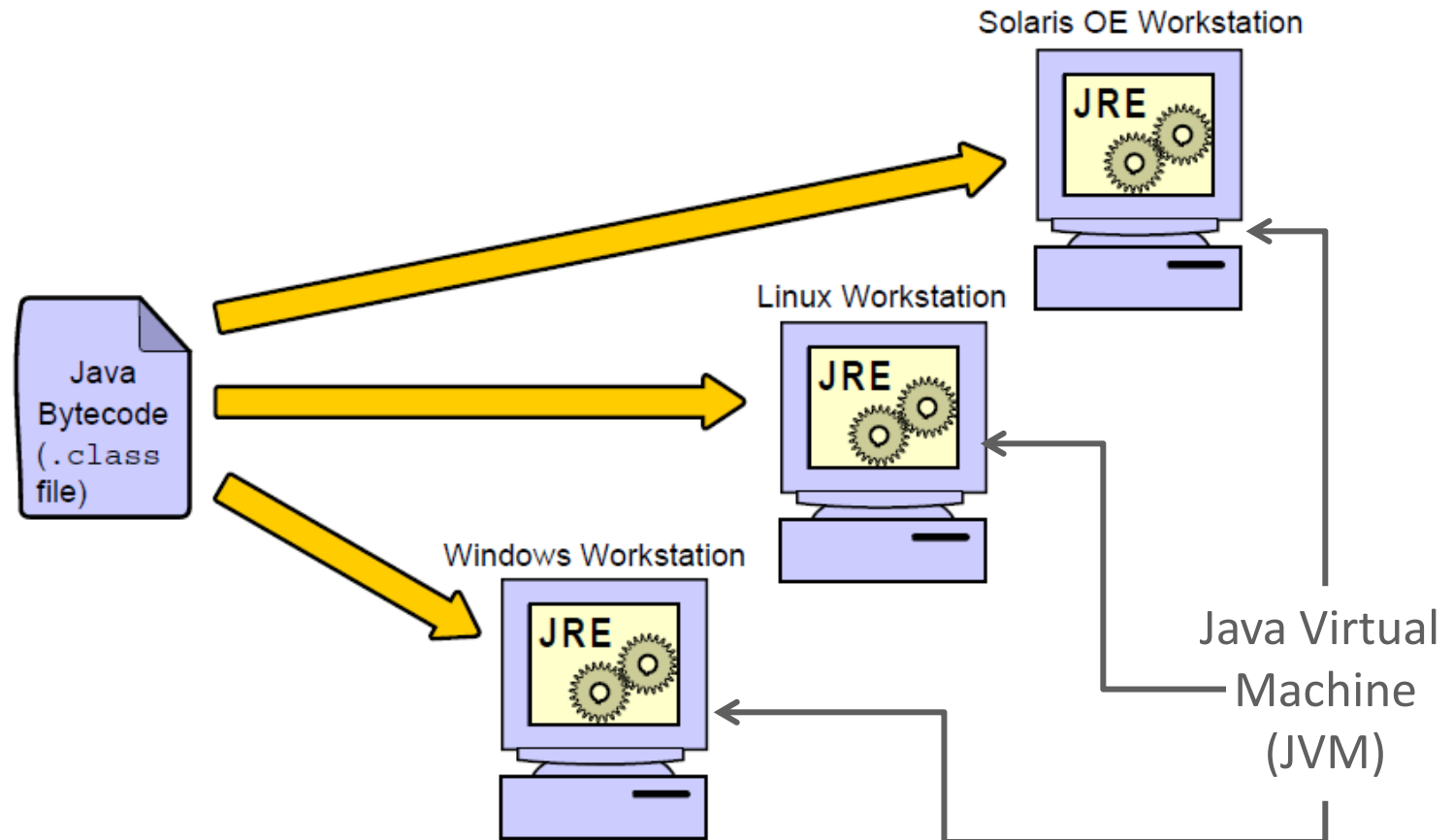




# Java Programs Run in a JVM

A Java Virtual Machine (JVM) interprets the bytecode, allowing the program to run on any machine with a Java Runtime Environment (JRE) installed.

Programmers like writing their applications once, and having it run everywhere.



# Java Runtime Environment (JRE)




JRE

- Includes:
  - The Java Virtual Machine (JVM)
  - Java class libraries
- Purpose:
  - Read bytecode (`.class`)
  - **Run** the same bytecode anywhere with a JVM

# Java Development Kit (JDK)



JDK

- Includes:
  - JRE 
  - Java Compiler
  - Additional tools
- Purpose:
  - **Compile** bytecode (`.java` → `.class`)

# Integrated Development Environment (IDE)



NetBeans  
IDE

- Purpose:
  - Provide a sophisticated text editor
  - Offer assistance debugging code
  - Manage projects
  - **Write** source code (`.java`)
- Examples:
  - NetBeans
  - Eclipse
  - Greenfoot and BlueJ

# Your Code

The code you type in NetBeans will look like this:

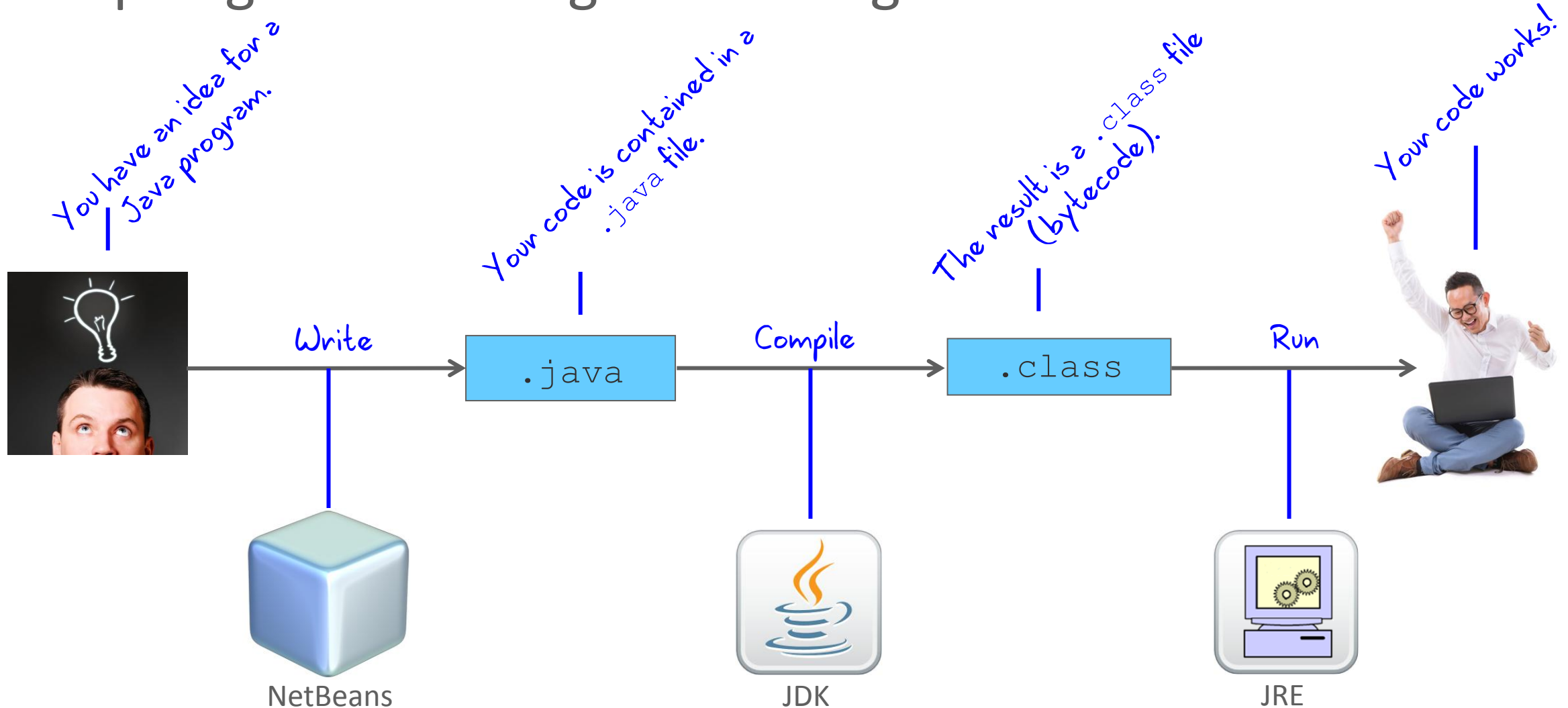
```
2 public class HelloWorld {
3
4     public static void main(String[] args) {
5         System.out.println("Hello World!");
6     }
7 }
```

You'll look like this when your code finally works:

It often won't work and will need debugging first.



# Compiling and Running a Java Program



# 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.



Java™  
ORACLE®