15CSE202 Object Oriented Programming
Lecture 3

Overview of Java

Nalinadevi Kadiresan
CSE Dept.
Amrita School of Engg.

About Java

- A programming language.
  - As defined by Gosling, Joy, and Steele in the Java Language Specification
- A platform
  - A virtual machine (JVM) definition.
  - Runtime environments in diverse hardware.
- A class library
  - Standard APIs for GUI, data storage, processing, I/O, and networking.

Java notes for C++ programmers

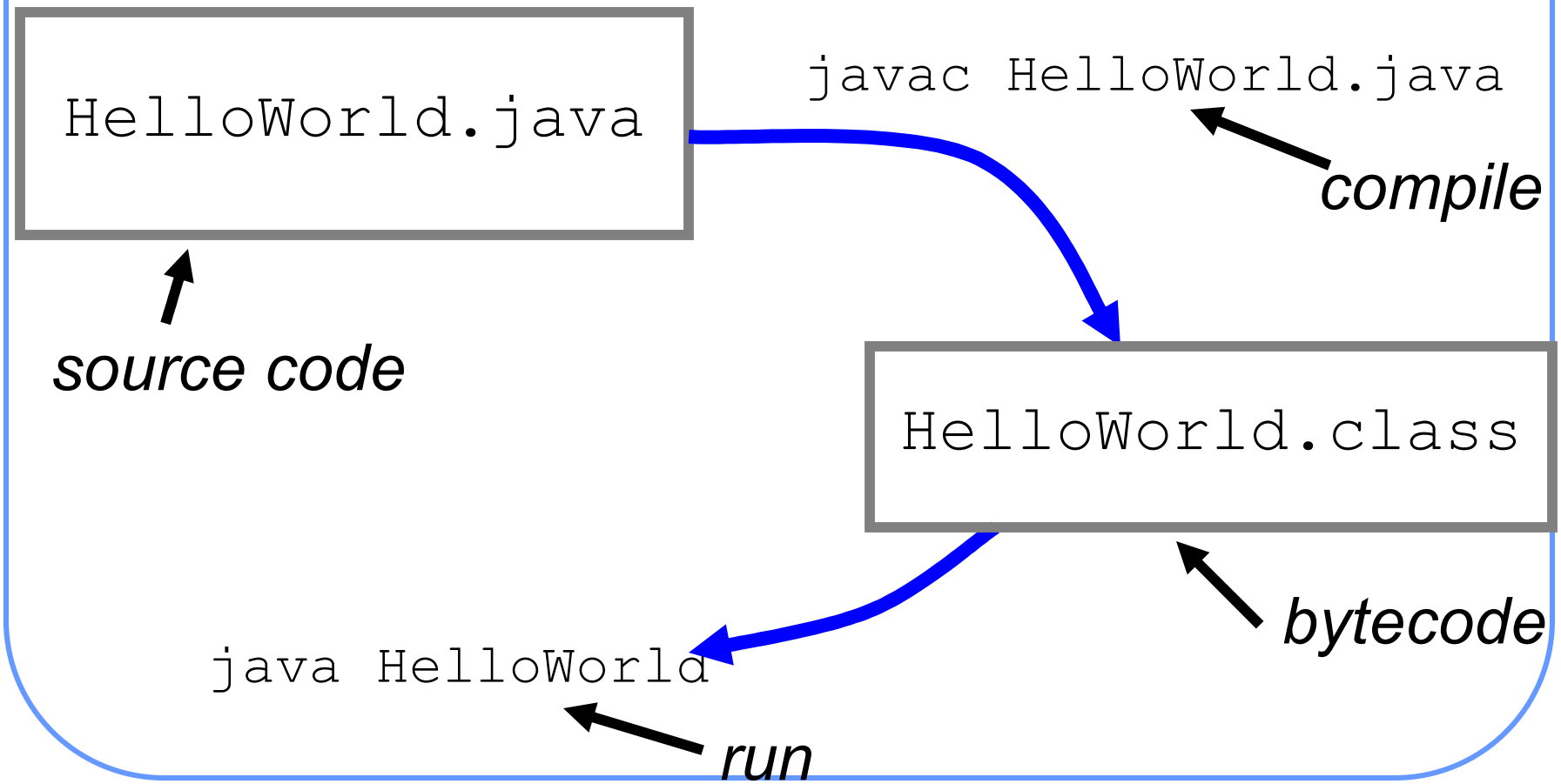
- Everything is an object.
  - Every object inherits from java.lang.Object
- No code outside of class definition!
  - No global variables.
- Single inheritance
  - an additional kind of inheritance: interfaces
- All classes are defined in `.java` files
  - one top level public class per file

First Program

```
public class HelloWorld
{
  public static void main(String args[])
  {
    System.out.println("Hello World");
  }
}
```

- Save this file as HelloWorld.java
- **To compile:** javac HelloWorld.java
- **To execute:** java HelloWorld
- **Output:** Hello world

Compiling and Running

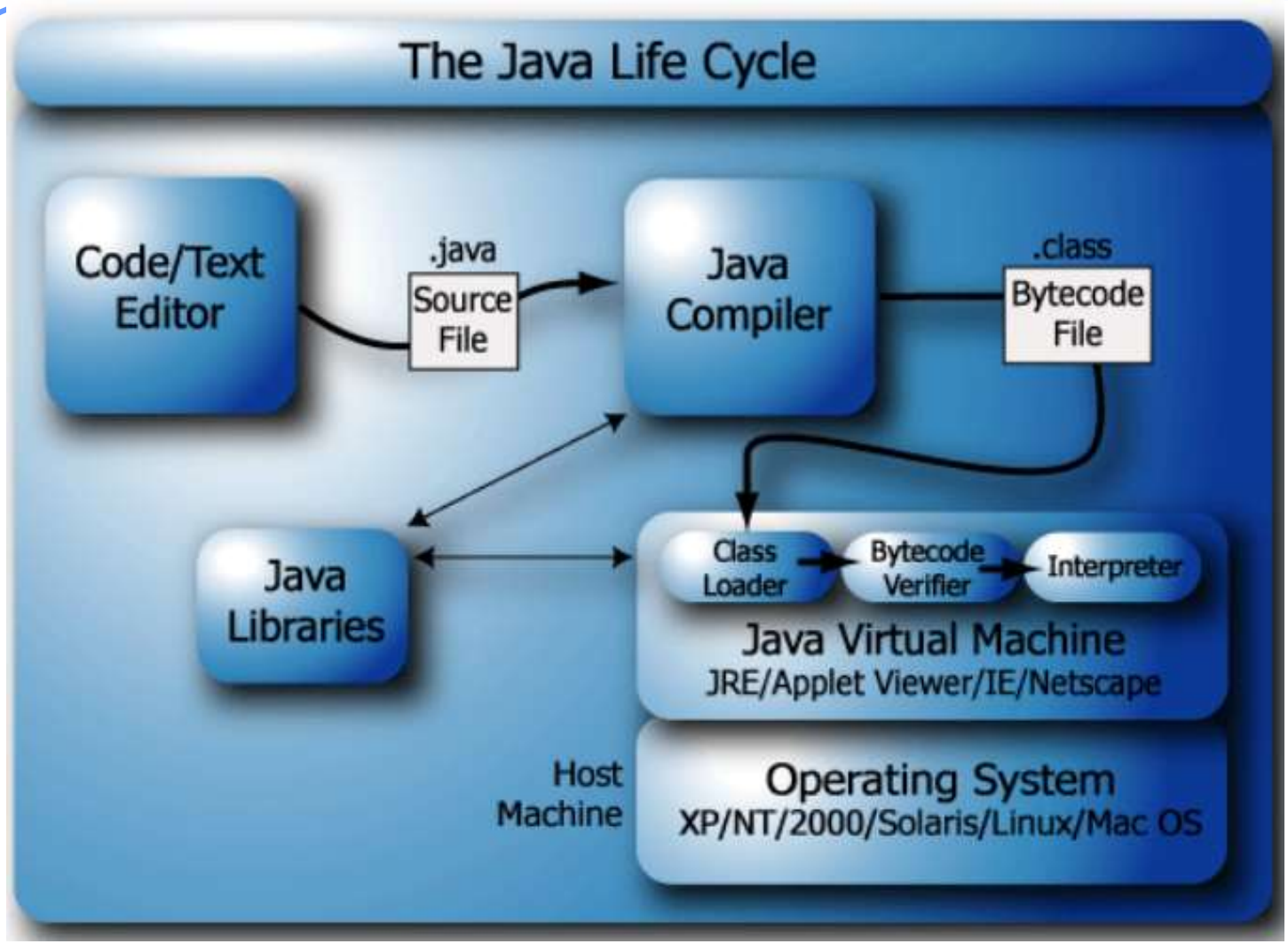


Java bytecode and interpreter

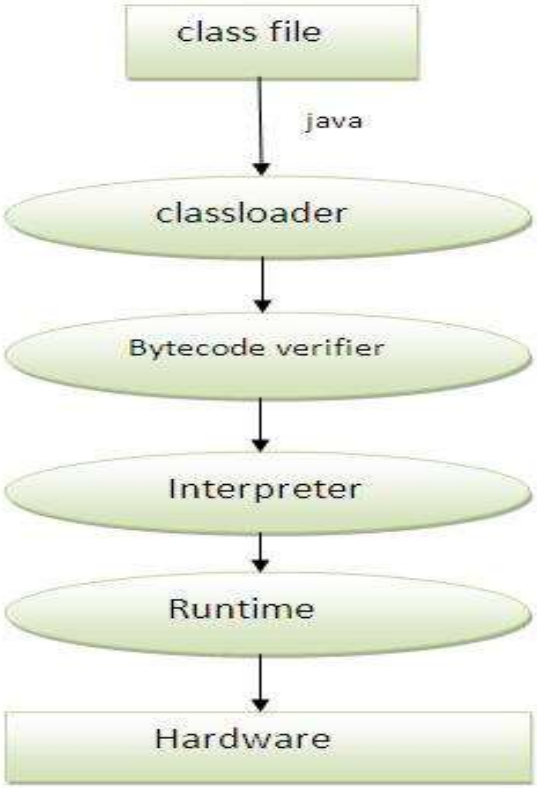
- bytecode is an intermediate representation of the program (class).
- The Java interpreter starts up a new "Virtual Machine".
- The VM starts executing the users class by running it's **`main()`** method.

PATH and CLASSPATH

- The *java_home/bin* directory is in your $PATH
- If you are using any classes outside the java or javax package, their locations are included in your $CLASSPATH


The Java Life Cycle
Code/Text Editor
.java
Source File
Java Compiler
.class
Bytecode File
Java Libraries
Class Loader
Bytecode Verifier
Interpreter
Java Virtual Machine
JRE/Applet Viewer/IE/Netscape
Host Machine
Operating System
XP/NT/2000/Solaris/Linux/Mac OS

Java program at run time



Classloader: is the subsystem of JVM that is used to load class files.

Bytecode Verifier: checks the code fragments for illegal code that can violate access right to objects.

Interpreter: read bytecode stream then execute the instructions.

Difference between JDK, JRE and JVM

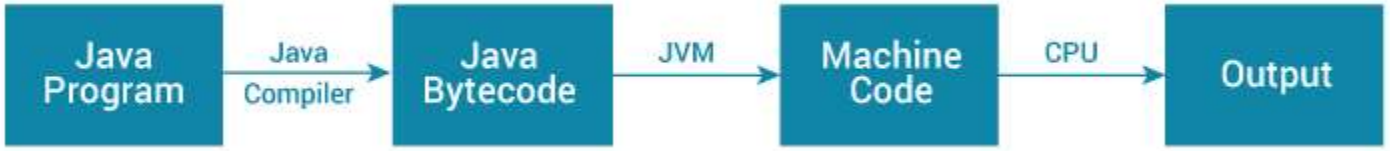
- Understanding the difference between JDK, JRE and JVM is important in Java.
  - We are having brief overview of JVM here.
  - Let's see the basic differences between the JDK, JRE and JVM.

JVM

JVM (Java Virtual Machine) is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed.

JVMs are available for many hardware and software platforms. JVM, JRE and JDK are platform dependent because configuration of each OS differs. But, Java is platform independent.

- The JVM performs following main tasks:
  - Loads code
  - Verifies code
  - Executes code



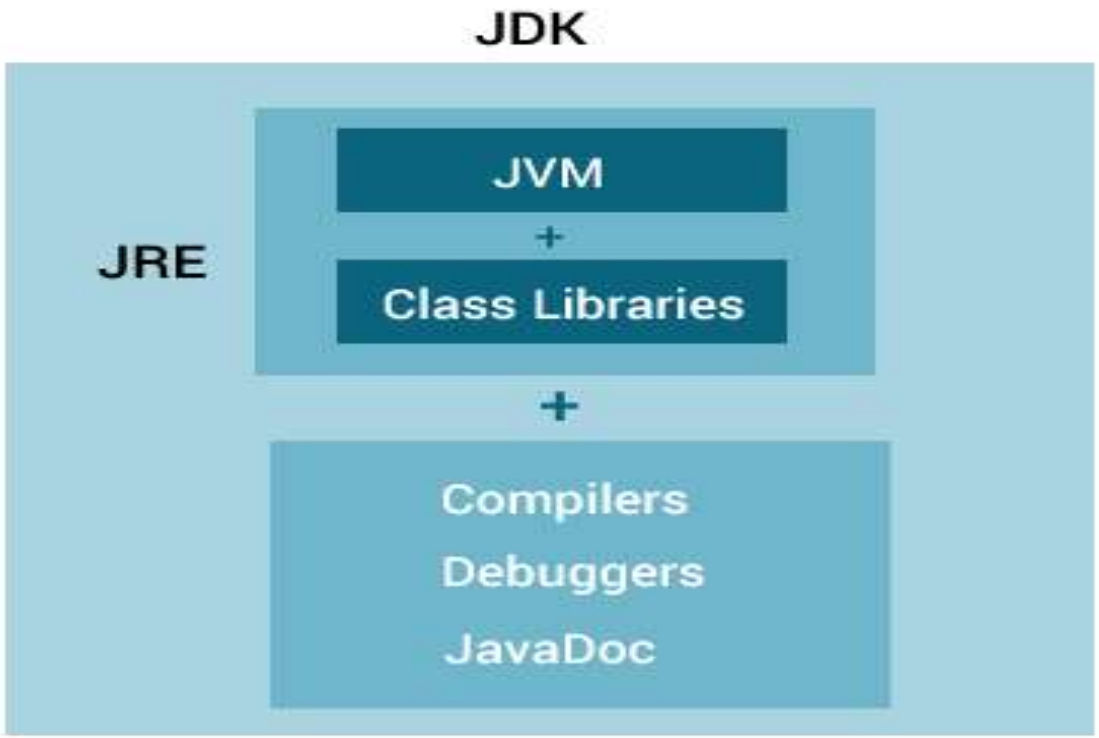
JRE

JRE is an acronym for Java Runtime Environment .It is used to provide runtime environment. It is the implementation of JVM. It physically exists. It contains set of libraries + other files that JVM uses at runtime.

JDK

- JDK is an acronym for Java Development Kit. It physically exists.
- When you download JDK, JRE is also downloaded
- It contains JRE + development tools.

JVM, JRE, JDK relationship



Java vs Python

Java	Python
Java is a compiled language	Python is a interpreted language
Java is a OO language	Python is a multi-paradigm language – OO and functional
Java is statically typed (Type checking at compile time)	Python is dynamically typed (Type checking at runtime)
Java implements concurrency with multithreading	Python implements concurrency at process level.

Java vs Python – Execution Speed

In terms of speed, Java is **faster** than Python as it is a compiled language. It takes less time to execute a code.

Python is an interpreted language and it determines the type of data at run time which makes it **slower** comparatively.

Java vs Python – Legacy system

Java's history in the enterprise and its slightly more verbose coding style mean that Java legacy systems are typically **larger** and more numerous than python's.

Python has less legacy problem so organization finds **difficulty** for the script to **copy and paste** codes and gives it a slight edge over the other languages.

Java vs Python – practical agility

Java enjoys more undeviating refactoring support than python thanks to its static type system and universality of IDE's in development. It is more popular for **mobile and web applications**.

Python has always had an existence in the talent space and is considered to be the most favorable language for Machine Learning, Artificial Intelligence, IoT and a lot more.

- **Refactoring is a controlled technique for improving the design of an existing code base.**
  - **small, manageable steps that incrementally increase the cleanliness of code while still maintaining the functionality of the code**

Java vs Python

- Choosing which language to goes with depends on your company's needs, and which setbacks you're willing to accept.
  - While Java churns out higher performance speed, Python is more suited to evolve legacy systems.
  - When it comes to practical agility, Java is a more proven option, while Python has more flexibility for experimentation.

Java vs Python

Comparison Factors	Java	Python
Speed	✓	✗
Legacy	✗	✓
Code	✗	✓
Practical Agility	✓	✓
Trends	✗	✓
Salary	✗	✓
Syntax	✓	✓

Courtesy: https://stackify.com/java-vs-python/
https://www.edureka.co/blog/java-vs-python/

Supplemental reading

- ***Getting Started***

  http://java.sun.com/docs/books/tutorial/getStarted/index.html

- ***Nuts and bolts of the Java Language***

  http://java.sun.com/docs/books/tutorial/java/nutsandbolts/index.html

- ***Compiling and Running a Simple Program***

  http://developer.java.sun.com/developer/onlineTraining/Programming/BasicJava1/compile.html

Object Oriented Development

- First Goal: Model the *objects* of the world
  - Noun-oriented
  - Focus on the *domain* of the program
- Phases
  - **Object-Oriented Analysis: understand the domain**
    - Define an object-based model of it
  - **Object-Oriented Design: Define an implementation**
    - Design the solution
  - **Object-Oriented Programming: Build it**

Next Session will be

Object Oriented Analysis