Getting started with Java

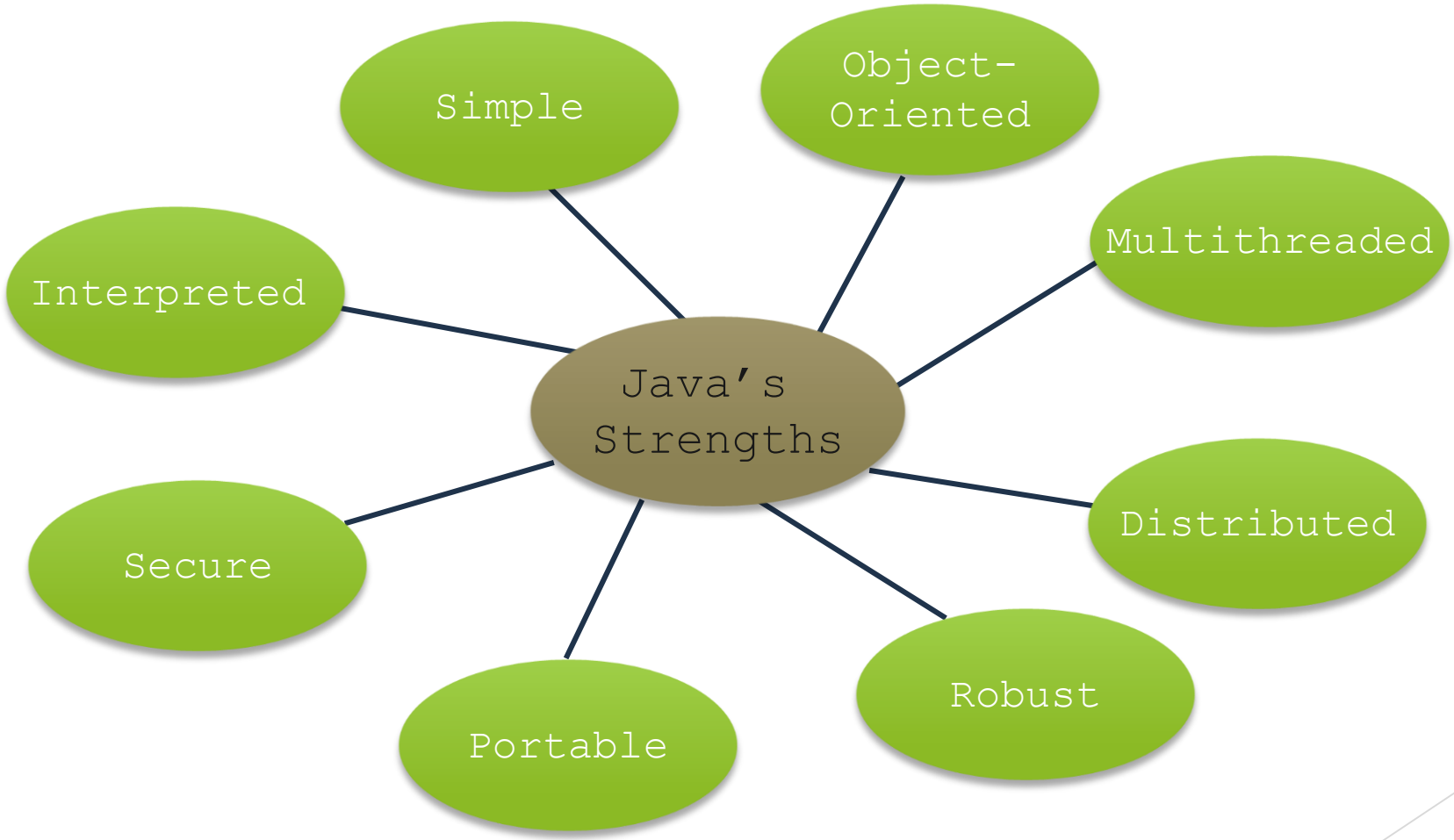
Agenda

- History of Java
- Features of Java
- How Java works
- Types of Java Programs
- Edit, compile, and run Java applications

History of Java

- Java was conceived by *James Gosling, Patrick Naughton and team* in *1991.*
- First version took 18 months (“Oak”).
- Oak was renamed to “Java” in 1995.
- Started from JDK 1.0
- Latest version is JDK 1.8

Features of Java



Simple

- syntax is based on C++ (so easier for programmers to learn it after C++).
  - Declaring variables
  - Writing methods
  - Control structure

- removed many confusing and/or rarely-used features e.g., explicit pointers, operator overloading etc.

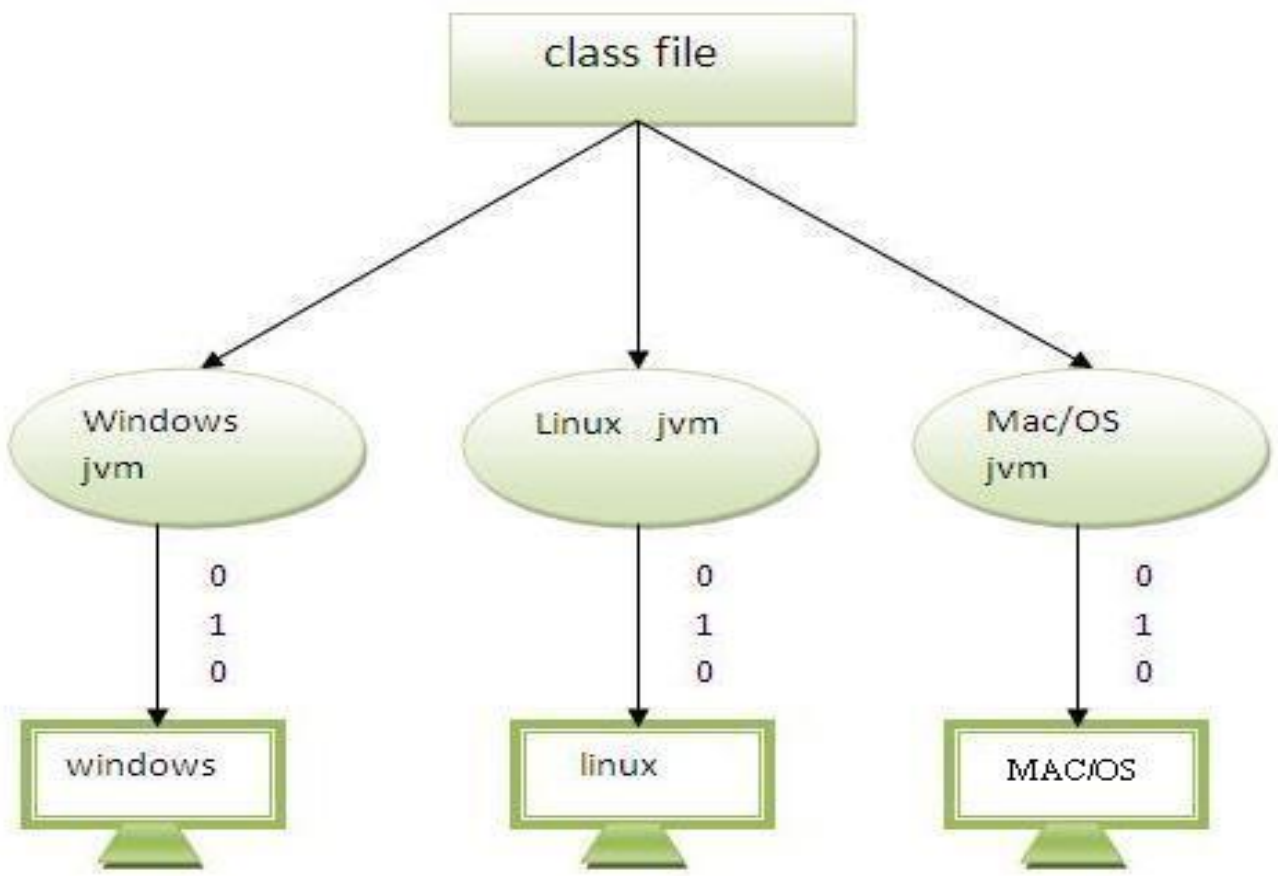
- No need to remove unreferenced objects because there is Automatic Garbage Collection in java.

Object Oriented

- Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behavior.

- Basic concepts of OOPs are :
  - Object
  - Class
  - Inheritance
  - Polymorphism
  - Encapsulation
  - Abstraction

Platform Independent



Secured

- No explicit pointer
- Programs run inside virtual machine sandbox.

Robust

- Robust simply means strong.
- Java uses strong memory management.
- There is automatic garbage collection in java.
- There is exception handling and type checking mechanism in java.
- All these points makes java robust.

Architecture-neutral

- There is no implementation dependent features e.g. size of primitive types is set.

Portable

- We may carry the java bytecode to any platform.

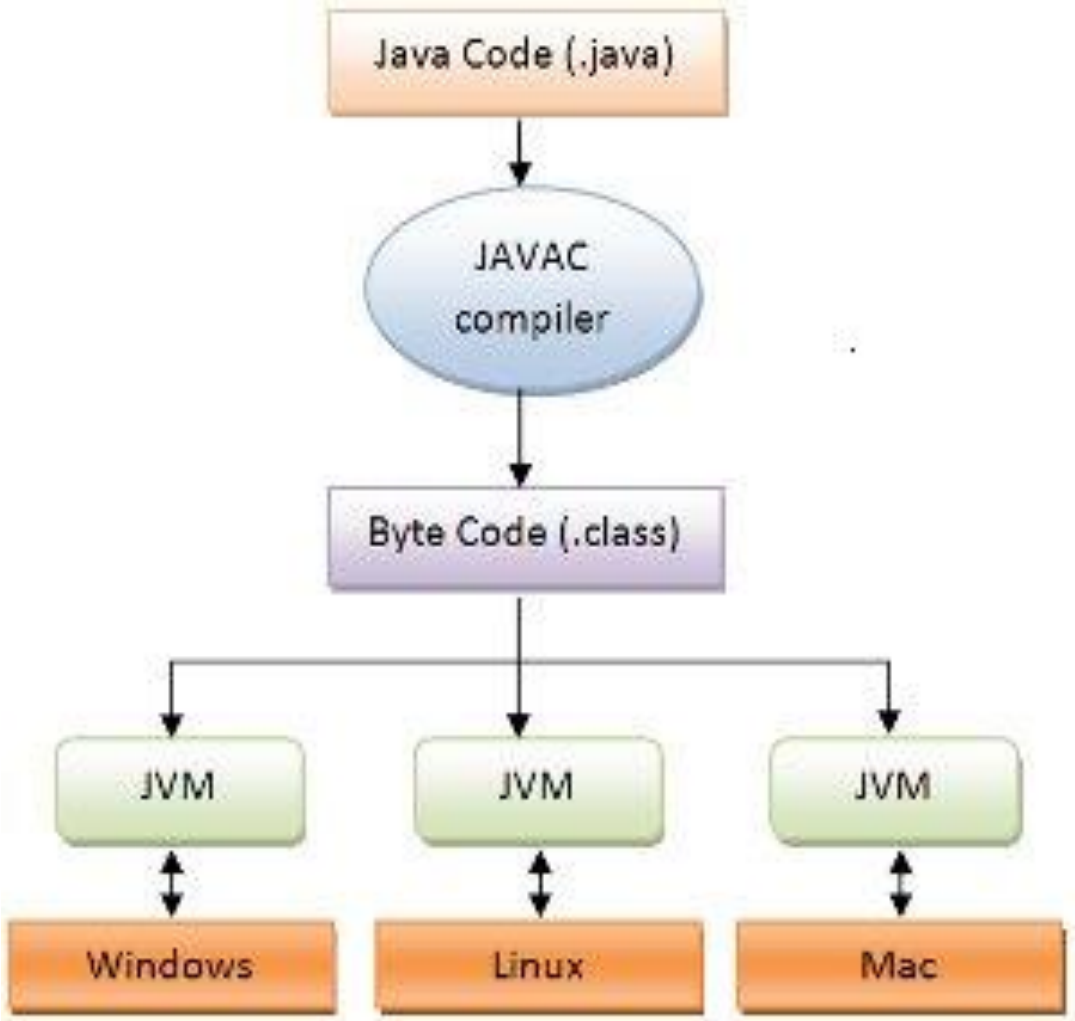
Distributed

- We can create distributed applications in java.
- RMI and EJB are used for creating distributed applications.
- We may access files by calling the methods from any machine on the internet.

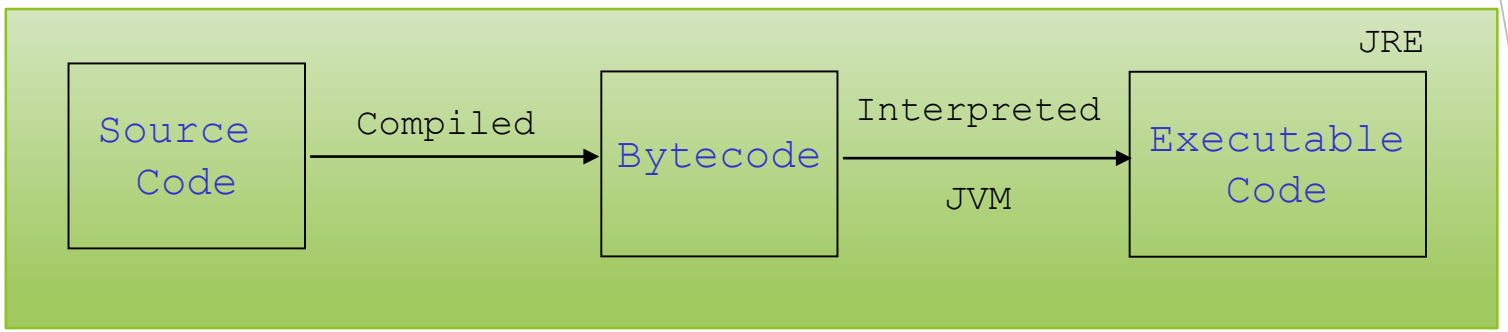
Multi-threaded

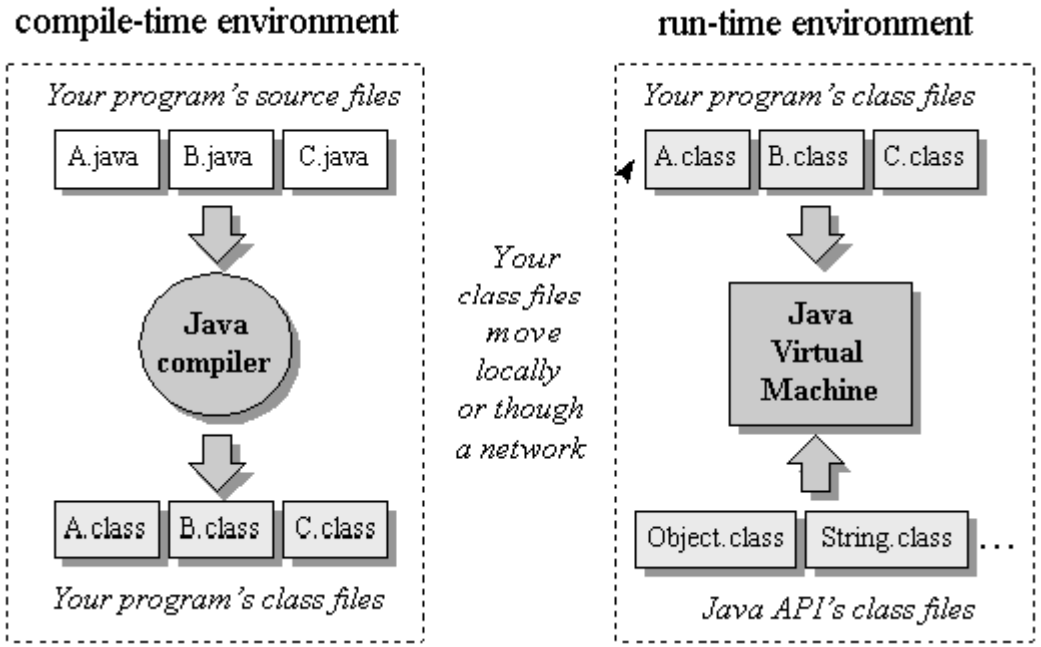
- A thread is like a separate program, executing concurrently.
- We can write Java programs that deal with many tasks at once by defining multiple threads.

How Java works?



Compilation and Execution





The Bytecode

- Bytecode is the intermediate representation of Java programs.
- Bytecode understanding helps in debugging and doing performance and memory usage tuning.

Java Virtual Machine

- JVM uses stack-based model of computation.
- Each thread has a JVM stack which stores frames.
- Each time a method is invoked a new stack frame is created.
- Each stack frame consists of Operand Stack, Array of local variables, and a reference to Constant Pool

Types of Java Programs

- Java application
- Java applet

Java Application

- A desktop application
- Used on a machine (desktop)

Java Applet

- Stored on a website
- Downloaded and run on a client computer from within a web browser.

What you need for Java Development

- JDK – Java Development Kit
- JRE – Java Runtime Environment

Where to get it from?

- http://www.oracle.com/technetwork/java/javase/downloads/index.html
- The link holds the downloadable.

The code

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

keywords

Save the file with the name “HelloWorld.java”

1. Write the program in notepad
2. Save the program
3. Compile program - javac HelloWorld.java
4. Run program – java HelloWorld

How to compile?

- javac is the command used for compiling the code
- E.g.

  javac HelloWorld.java

How to run?

- java command is used for running the program
- E.g.

  java HelloWorld

Assignment

- Print your name on console using Java Application.
- Print “Welcome To Java” on the console.

Thank you