OOP Using Java

Faculty: SREEDIVYA
Branch: S6 ECE

History of Java

- Sun Microsystems developed Java in the early 1990s.
- Founder – James Gosling
- First named as ‘Oak’.
- Gosling designed Java with a C/C++ - style syntax that system and application programmers would find familiar.
- Most popular for generating internet – based applications. Hence it is known as the language of internet.

- 1991 - Green Project for consumer electronics market (Oak language → Java)
- 1994 – HotJava Web browser
- 1995 – Sun announces Java
- 1996 – JDK 1.0
- 1997 – JDK 1.1 RMI, AWT, Servlets
- 1998 – Java 1.2 Reflection, Swing, Collections
- 2004 – J2SE 1.5 (Java 5) Generics, enums
- 2014 – Java SE 8 Lambdas - functional programming

- 2017 - Java SE 9
- 2018 - Java SE 10, Java SE 11
- 2019 - Java SE 12, Java SE 13
- 2020 - Java SE 14, Java SE 15
- 2021 - Java SE 16, Java SE 17
- 2022 - Java SE 18, Java SE 19
- 2023 - Java SE 20
- 2024 - Java SE 21
- As of March 2024, Java 22 is the latest version.

Features

- Simple
- Secure
- Portable
- Object-oriented
- Robust
- Multithreaded
- Architecture Neutral
- Interpreted
- Distributed

- **Simple**

  There is no complicated statements, no Struct and Union, no pointer usage, no operator overloading, no multiple inheritance.

- **Secure**

  Enables construction of virus - free systems and always run in Java Runtime Environment with almost null interaction with system OS.

- **Portable**

  Java source code is compiled to an intermediate class file called byte-code and it can be carried to any platform.

- **Object oriented**

  Java programming language is composed of objects and classes.

- **Robust**

  In a well – written Java program, all run – time errors are handled by the system. Strong type-checking and exception handling mechanism is there.

- **Multithreaded**

  Multiple tasks are performed in a single java program by defining multiple threads.

- **Architecture – Neutral**

  Independent of any processor type and machine architecture. Java programs are written based on the principle of “Write Once, Run Anywhere”(WORA).

- **Interpreted**

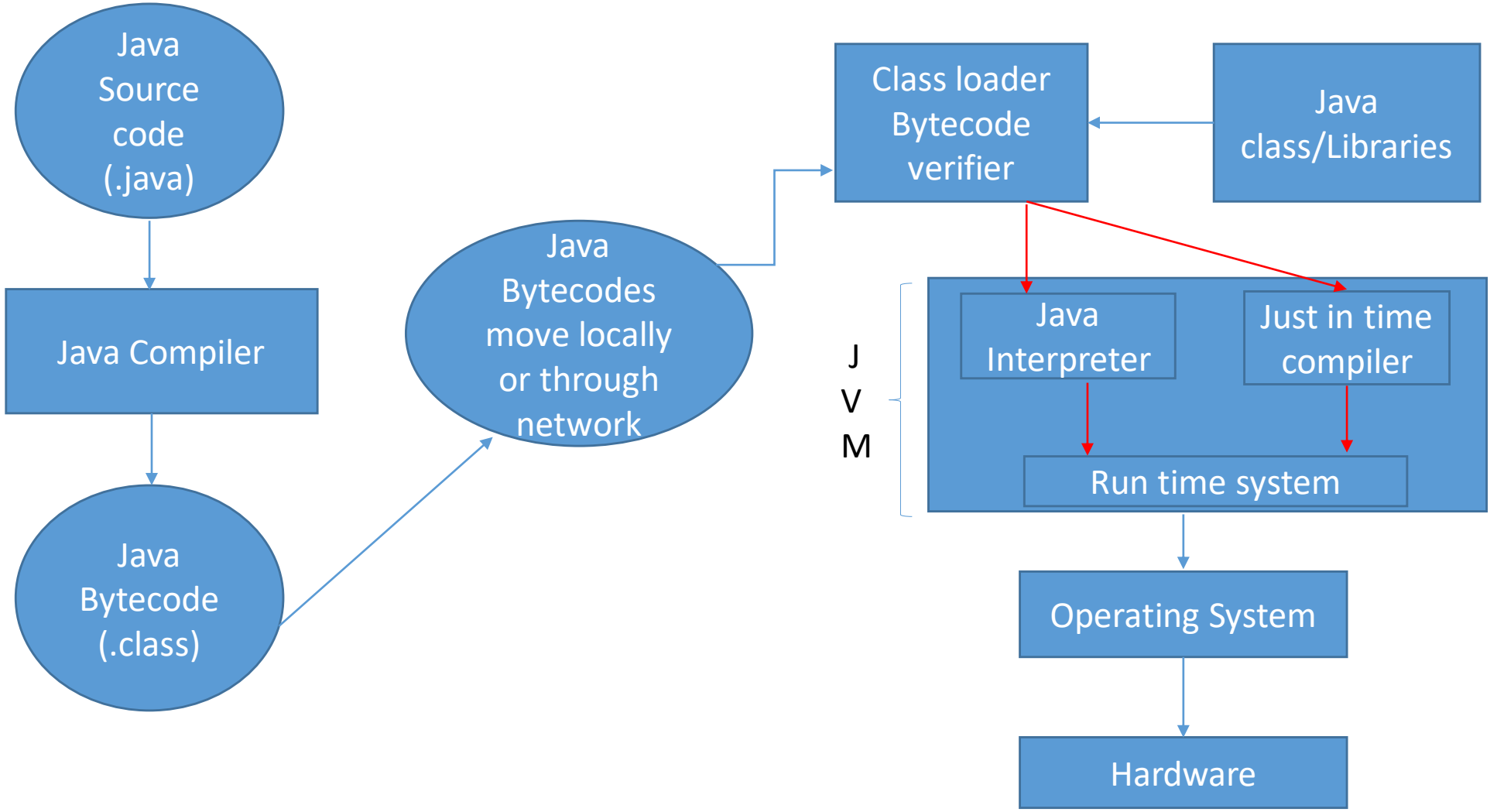
  Java source code is first compiled into a byte-code. This byte-code runs on the Java Virtual Machine (JVM), which is usually a software-based interpreter.

- **Distributed**

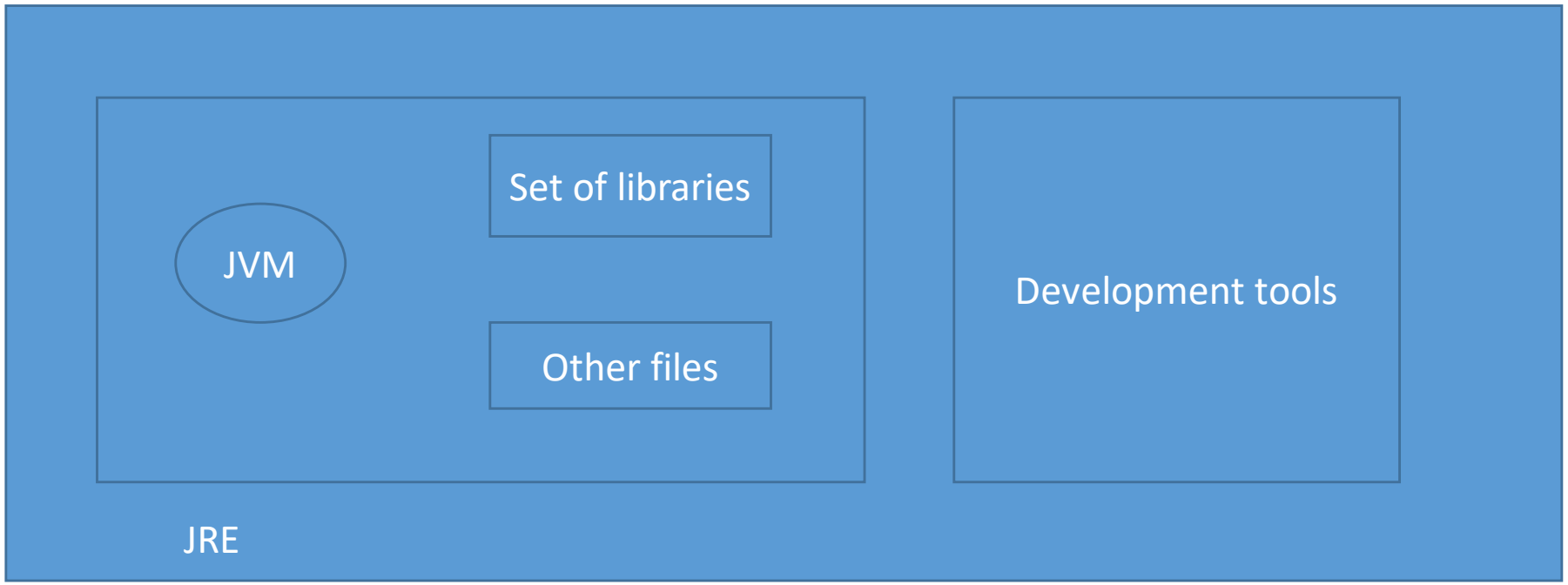
  Designed for the distributed environment of Internet.

Compile-time environment

Run-time environment



Java Platform



Java Development Kit (JDK)

- JDK contains tools needed to develop the Java programs and JRE to run the programs.

- The tools include Compiler(javac), Java application launcher(java), AppletViewer, The Java Debugger(jdb), Java Documentation Generator(javadoc) etc.

Java Runtime Environment(JRE)

- JRE is made up of a JVM, Java class libraries, and the Java class loader.

- Java Virtual Machine(JVM) executes bytecode. It is available on many types of software and hardware platforms which enables java to function as a platform on its own. JVM also performs garbage collection.

Basic Data types

- A data type tells the compiler:
  - how much memory to allocate
  - format in which to store data
  - type of operations to perform on data
- 8 Primitive data types supported by Java are,

  byte, char, boolean, short, int, long, float, double
- Supports literals(constant values) and strings

- Java uses variables for storing values.
- Supports expressions.
- Supports all arithmetic, increment and decrement, assignment, bitwise, relational, logical operators.
- Control statements such as if, if else, switch, for, while, do while, for each and jumping statements such as break, continue, return are also used in Java.

Type Casting

Process of converting value of a data type to some other data type e.g., int to byte or float to double. There are 2 types:

(1) Implicit typecasting (Widening conversion)

automatic conversion of lower data type to higher data type.

(2) Explicit typecasting (Narrowing conversion)

explicit conversion of higher data type to lower data type.

Java Program to print “HelloWorld”

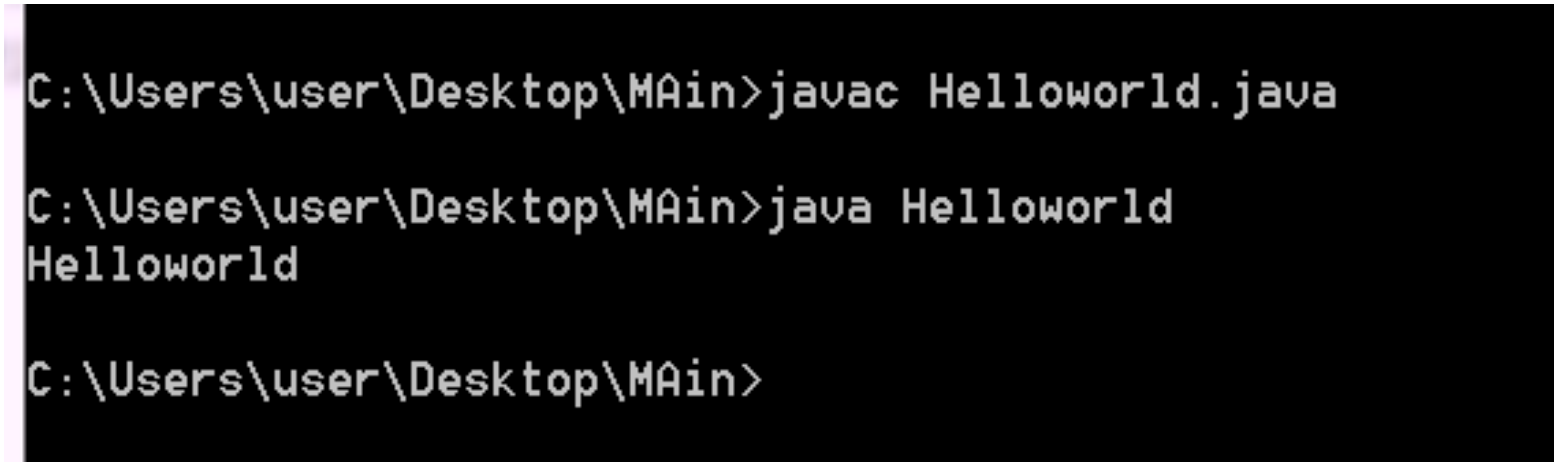
```
public class Helloworld
{
        public static void main(String ar[])
        {
                System.out.println("Helloworld");
        }
}
```

Save the program as Helloworld.java in any folder.

For compilation: javac Helloworld.java

For execution: java Helloworld

OUTPUT


C:\Users\user\Desktop\MAin>javac Helloworld.java
C:\Users\user\Desktop\MAin>java Helloworld
Helloworld
C:\Users\user\Desktop\MAin>