Core Java Programming - 1

Sandeep Kulange

sandeepkulange@sunbeaminfo.com

Day 1: Agenda

- Language, Technology, Platform, Framework
- History
- The Java Programming Language Platforms
- SDK, JDK, JRE, JVM
- JDK Installation Directory Structure
- Simple Hello Application
- Execution Flow

Language

- Example: C, C++, Java, Python, C#, Go etc.
- The Programming Language has
  1. Syntax and semantics
  2. Data Types
  3. Tokens
  4. Built in features
- We can use it to develop different type of application
  1. Console User Interface(CUI) / Command Line Interface(CLI)
  2. Graphical User Interface (GUI)
  3. Library Application
  4. Distributed application
- Generally it is used to implement business logic.

Platform

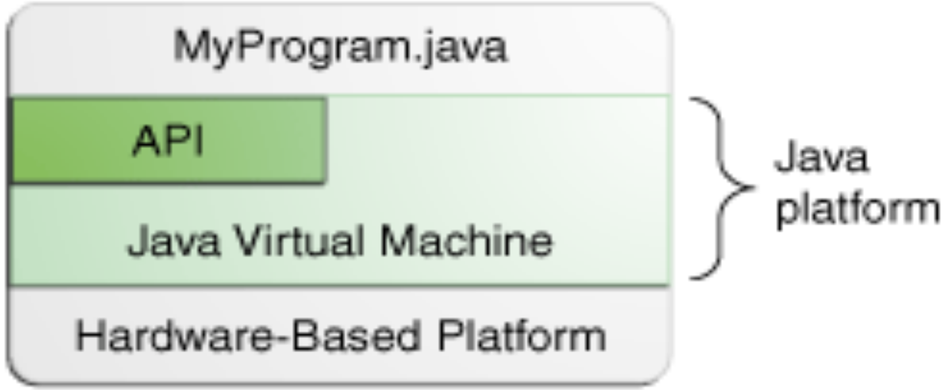
- A platform is the hardware or software environment in which a program runs.
- Most platforms can be described as a combination of the operating system and underlying hardware.
- Types of platform:
  1. Hardware Based Platform
  2. Software-only platform

- Hardware Based Platforms
  1. Microsoft Windows OS(Win Vista, Win 7, Win 8/8.1, Win 10)
  2. Unix/Linux(RHEL, Open SuSE, Ubuntu)
  3. Mac OS(High Sierra, Mojave, Catalina)

- Software-only Platform
  1. PHP, Microsoft .NET, Java

Java Platform

- The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.



- The Java platform has two components:
  1. The Java Virtual Machine
  2. The Java Application Programming Interface (API)

Framework

- It is the library of readymade types(interfaces/classes/enum) that can be used to develop application.

- Example
  1. AWT and Swing: GUI Framework
  2. RMI: Distributed application development framework
  3. Struts: MVC based, web application development framework
  4. Hibernate: Automatic persistence framework
  5. Junit: Testing Framework
  6. Spring: Light weight application development framework.

Technology

- Language provides syntax that we can use to implement business logic.
- Technology provides techniques and features that we can use to develop application.
- Platform provides environment in which we can run application.
- Example: ASP.NET, Java
- ***The Java language is both technology as well as platform.***

History

- The Java is an object oriented programming language developed by James Gosling and his team(Green Team) at Sun Microsystems in 1991.

- The Java programming language is derived from C and C++.

- Initial name of the Java was Oak. But name “Oak” was trademark of Oak Technology hence in 1995 it is renamed to Java.

- In 2009, Sun Microsystems was overtaken by Oracle Corporation.

- It promised **Write Once, Run Anywhere** (**WORA**) functionality.

- Java is object oriented, generic as well as functional programming language.

The Java Language Features

1. Simple
2. Object Oriented
3. Architecture Neutral
4. Portable
5. Robust
6. Multithreaded
7. Dynamic
8. Secure
9. High Performance
10. Distributed

The Java Platforms

- There are four platforms of the Java programming language:
    1. Java Platform, Standard Edition (Java SE)
    2. Java Platform, Enterprise Edition (Java EE)
    3. Java Platform, Micro Edition (Java ME)
    4. Java FX

- All Java platforms consist of a Java Virtual Machine (VM) and an application programming interface (API).

- The Java Virtual Machine is a program, for a particular hardware and software platform, that runs Java technology applications.

The Java Versions

Sr. No	**Java Platform**	**Year**
1	JDK Beta	1995
2	JDK 1.0	January 1996
3	JDK 1.1	February 1997
4	J2SE 1.2	December 1998
5	J2SE 1.3	May 2000
6	J2SE 1.4	February 2002
7	J2SE 5.0	September 2004
8	Java SE 6	December 2006
9	Java SE 7	July 2011
10	**Java SE 8(LTS)**	March 2014
11	Java SE 9	September 2017
12	Java SE 10	March 2018
13	**Java SE 11(LTS)**	September 2018

The Java Versions

Sr. No	**Java Platform**	**Year**
14	Java SE 12	March 2019
15	Java SE 13	September 2019
16	**Java SE 14**	March 2020
17	Java SE 15	September 2020
18	Java SE 16	March 2021
19	**Java SE 17 (LTS)**	September 2021

Software Development Kit

- To develop an application, toolkit need to be install on developers machine it is called software development kit(SDK).

- SDK = Language Tools + Documentation + Supporting Libraries + Runtime environment.

- Language Tools:
  1. Compiler
  2. Debugger etc.

JDK, JRE, JVM

- Java language SDK is also called Java Development Kit(JDK).

- **JDK = Java Language Tools + Java Documentation + rt.jar + Java Virtual Machine(JVM).**

- JVM and rt.jar are integral part of Java Runtime Environment(JRE).
- JRE is runtime environment of java in which we can execute java application.

- **JDK = Java Language Tools + Java Documentation + JRE.**

- The Java Virtual Machine is a program, for a particular hardware and software platform, that runs Java technology applications.
- “rt.jar” file contains all core java API’s in compiled form.

Basic Tools

- Following tools are foundation of the JDK.

Sr. No.	Tool Name	Description
1	javac	The compiler for the Java programming language.
2	java	The launcher for Java applications
3	jdb	The Debugger for Java application
4	jar	Create and manage Java Archive (JAR) files.
5	javap	Class file disassembler
6	javadoc	API documentation generator.
7	appletviewer	Run and debug applets without a web browser.

- %JDK’s installation directory%/bin contains all java language tools.
- Default directory For Windows : "**C:\Program Files\Java\jdk1.x.x\**
- Default directory For Ubuntu : /usr/lib/jvm/java-8-openjdk-amd64
- Default directory For Mac OS : /Library/Java/JavaVirtualMachines/jdk1.8.0_201.jdk/Contents/Home/bin

JDK Installation Directory Structure

- Assuming the JDK software is installed at /jdk1.8.0, here are some of the most important directories:
  1. **src** : it contains source code of Java API
  2. **bin** : Executables for all the development tools contained in the JDK. The PATH environment variable should contain an entry for this directory.
  3. **include** : C-language header files that support native-code programming with the Java Native Interface
  4. **lib** : Files used by the development tools. Includes tools.jar, which contains non-core classes for support of the tools and utilities in the JDK.
  5. **jre** : Root directory of the Java Runtime Environment (JRE) used by the JDK development tools. The runtime environment is an implementation of the Java platform.
  6. **man** : Contains man pages for the JDK tools.
  7. **docs** : Contains documentation for Java API.
  8. **db** : Contains Java DB.

Simple Hello Application

- Consider a file name “MyProgram.java”

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

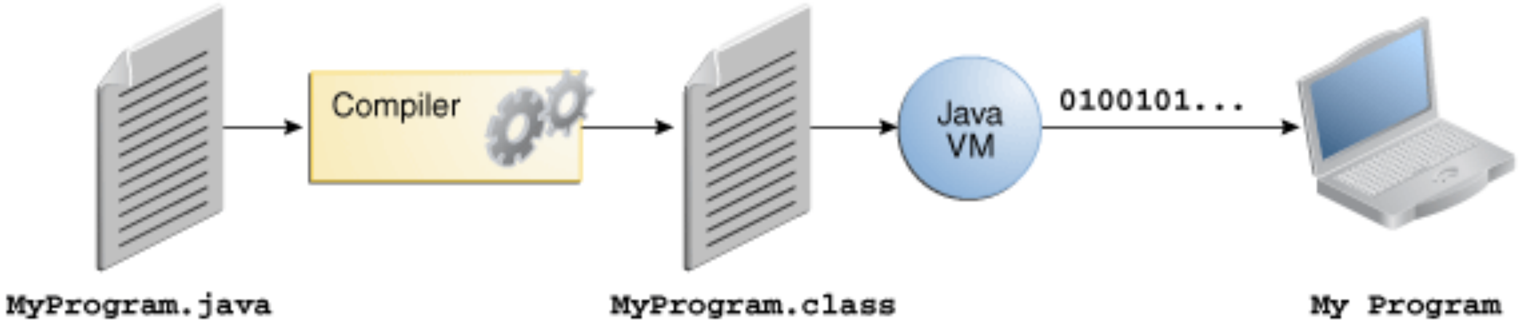
- Compilation and execution steps:

```
➢ javac MyProgram.java  [ Output : MyProgram.class ]
➢ java MyProgram        [ Output : Hello World!! ]
```

- Command to view .class file is:

```
➢  javap –c MyProgram.class
```

Execution Flow



- ”MyProgram.java” file contains java language source code. It means that it is a text file.
- “javac” is a java compiler which converts java source code into bytecode.
- “MyProgram.class” file contains bytecode.
- Bytecode is an object oriented assembly language code which is designed to run by JVM.
- JVM converts Bytecode into native CPU Code.

Thank You.