Introduction

Syllabus, Suggested book & Introduction

Syllabus

- Unit-I **Review of Programming Concepts:** program, identifiers, variables, constants, primitive data types, expressions, control statements, structured data types, arrays, functions.
- Unit-II **Object Oriented Concepts**: Abstraction, encapsulation, objects, classes, methods, constructors, inheritance, polymorphism, static and dynamic binding, overloading, Abstract classes, Interfaces and Packages.
- Unit-III **File Handling**: Byte Stream, Character Stream, File I/O Basics, File Operations,
- Serialization.
- Unit-IV **Exception handling**: Throw and Exception, Throw, try and catch Blocks, Multiple Catch Blocks, Finally Clause, Throwable Class, Types of Exceptions, java.lang Exceptions, Built-In Exceptions.
- Unit-V **GUI Design**: GUI based I/O, Input and Message Dialog boxes, Swng components, Displaying text and images in windows.

Suggested Books

- Java - The Complete Reference by Herbert Schildt
- The Java Language Specification by James Gosling, Bill Joy, Guy L. Steele Jr, Gilad Bracha, Alex Buckley
- Think Java_ How to Think Like a Computer Scientist by Allen B. Downey, Chris Mayfield
- OCA Java SE 8 Programmer I Exam Guide by Kathy Sierra and Bert Bates
- Programmer's Guide to Java SE 8 Oracle Certified Associate (OCA), A by Khalid Mughal and Rolf Rasmussen

History

- Java was conceived by James Gosling, Patrick Naughton, Chris Warth, Ed Frank, and Mike Sheridan at Sun Microsystems, Inc. in 1991.
- It took 18 months to develop the first working version. This language was initially called “Oak,” but was renamed “Java” in 1995.

Features of JAVA

- Security & Portability:
- Object Oriented. In Java, everything is an Object
- Platform Independent, Simple, Multithreaded.
  - ✓**Bytecode** is a highly optimized set of instructions designed to be executed by the Java run-time system, which is called the Java Virtual Machine (JVM).
  - ✓Translating a Java program into bytecode makes it much easier to run a program in a wide variety of environments because only the JVM needs to be implemented for each platform.
  - ✓ JVM will differ from platform to platform, all understand the same Java bytecode

Three OOP Principles

- Encapsulation
- Inheritance
- Polymorphism

Simple Program

```
class Example
 {
        public static void main(String args[]) {
        System.out.println("This is a simple Java program.");
        }
}
```

Compile: **javac** name of the file (Compiler) eg. Example.java

Run: **java** name of the class file having main function (interpreter)

Eg. java Example