Object Oriented Programming

WHAT IS JAVA ?

WHAT IS PROGRAMMING?

- Instruction to computer/device to perform task.
- Computer understands only 0 and 1. Nothing else.
- So, we need to send the instruction in the form of 0, 1
- Do you write program with just 0 and 1?

CLASSIFICATION/EVOLUTION OF PROGRAMMING

Machine level programming
- Send instruction in **binary** format

Assembly Programming
- Send **code** instead of binary code.
- Need **assembler** to convert to binary

High level programming
- Code is **close to English** Language
- Need **Compiler** to convert to binary
- 3 types –
  - Non structured
  - Structured/Procedural
  - Object Oriented Programming

CLASSIFICATION/EVOLUTION OF PROGRAMMING

Non structured

Generate spaghetti code

Sequential and has GoTo

COBOL, BASIC, FORTRAN

Structured/Procedural

Use Subroutine/Function

Improving the clarity, quality, and development time

C, PASCAL

Object Oriented Programming

Object-oriented programming (OOP) is a programming language model organized around objects rather than "actions" and data rather than logic.

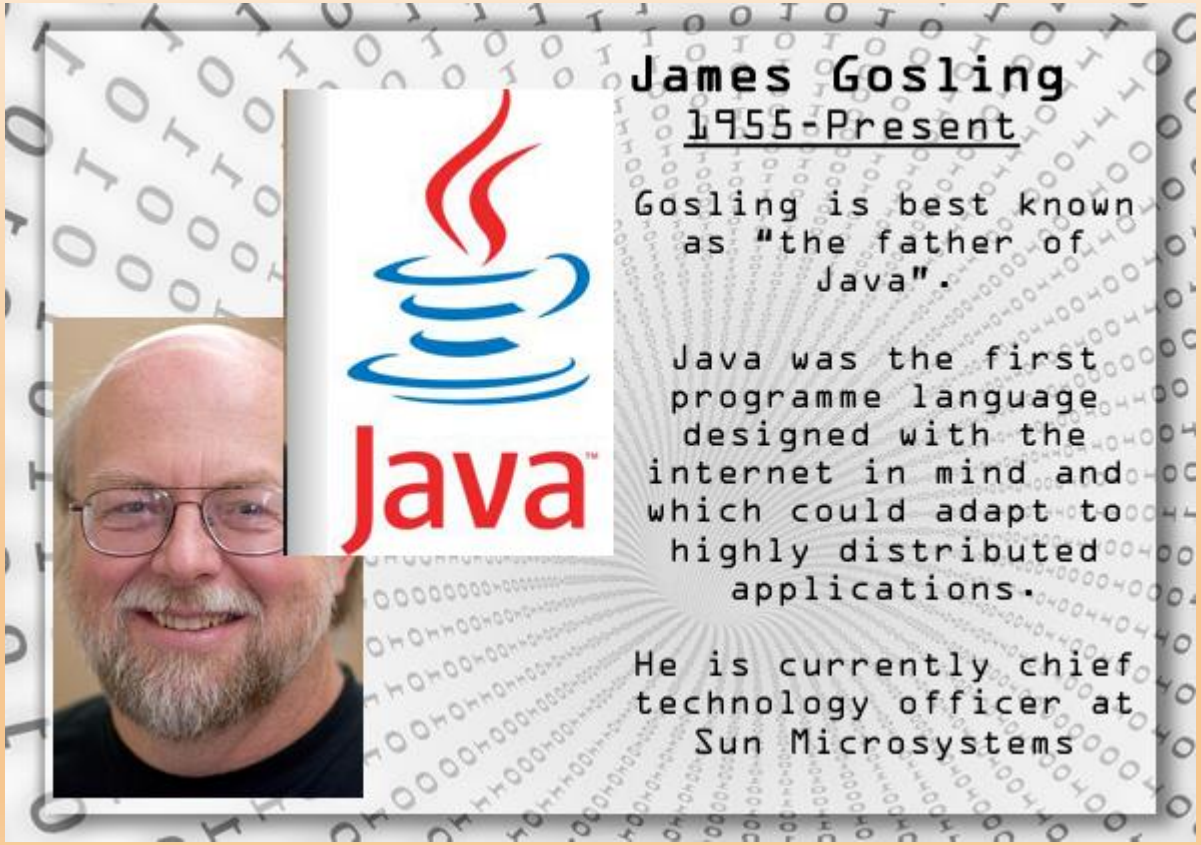
Historically, a program has been viewed as a logical procedure that takes input data, processes it, and produces output data.

Java, C++, C#

Introduction to JAVA

- A computer programming language that is used to build software.
- Also an application, development, and deployment environment.
- Help us visualize the program in real-life terms.
- Syntax of Java is similar to C and C++ based language.

History of JAVA:



Father of the Java programming language

History of JAVA:

- In 1990, Sun Microsystems began a research project named Green to develop the C and C++ based language.
- James Gosling, a network software designer was assigned to this project.
- Gosling's solution to the problems of C++ was a new language called Oak after an oak tree.
- In May 1995, Sun Microsystems formally announced Java (Oak was re-named Java) .
- Finally, in 1996, Java arrived in the market as a programming language.

According to Sun, 3 billion devices run java. There are many devices where Java is currently used. Some of them are as follows:

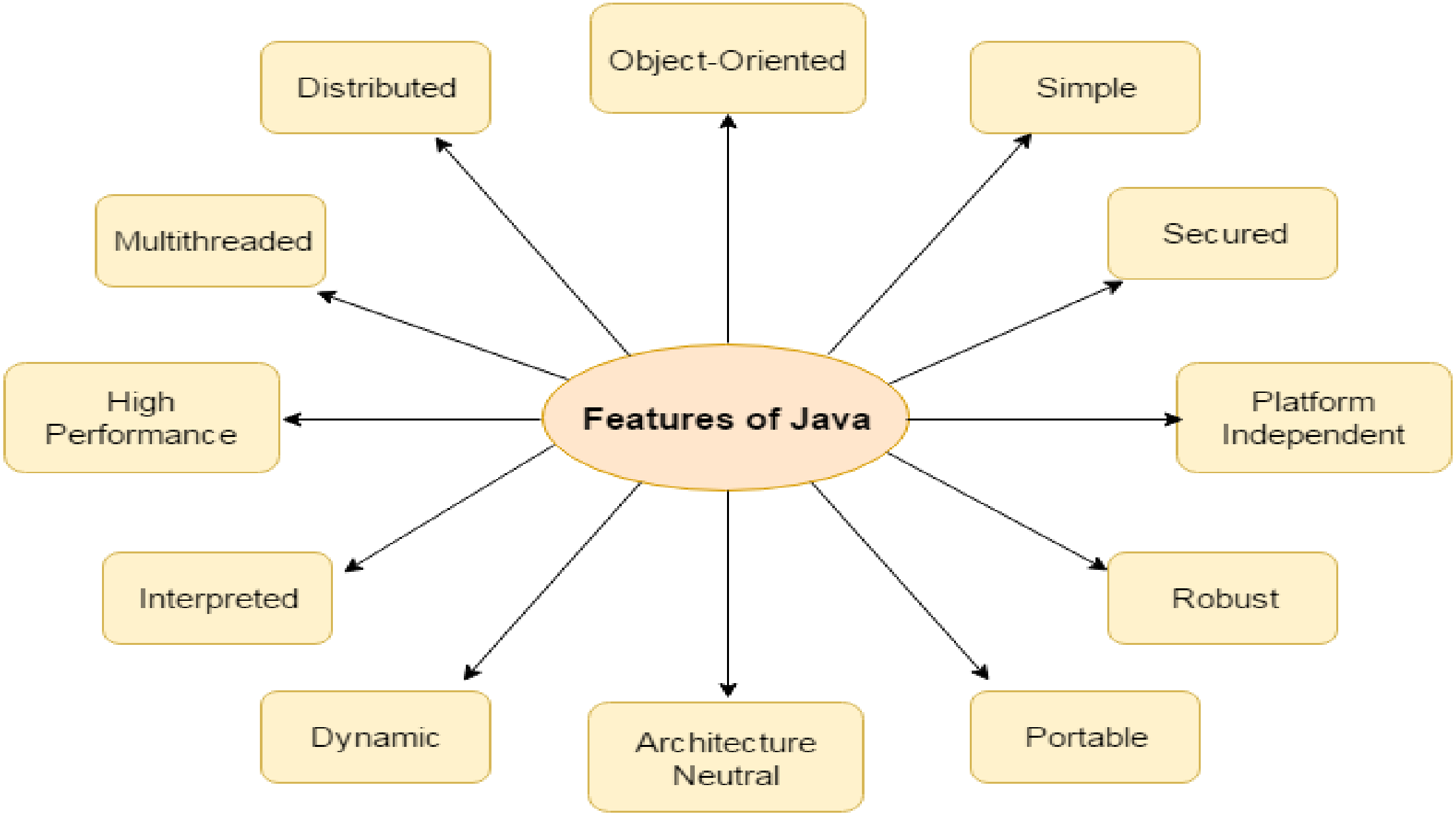
- Desktop Applications such as acrobat reader, media player, antivirus etc.
- Web Applications
- Enterprise Applications such as banking applications.
- Mobile
- Embedded System
- Smart Card
- Robotics
- Games etc.

There are 4 platforms or editions of Java:

- 1) Java SE (Java Standard Edition)
- 2) Java EE (Java Enterprise Edition)
- 3) Java ME (Java Micro Edition)
- 4) Java Fx (It is used to develop rich internet applications. It uses light-weight user interface API)

The Java Buzzwords

- The key considerations were summed up by the Java team in the following list of buzzwords:
  - Simple
  - Secure
  - Portable
  - Object-oriented
  - Robust
  - Multithreaded
  - Architecture-neutral
  - Interpreted
  - High performance
  - Distributed
  - Dynamic


Features of Java
Distributed
Object-Oriented
Simple
Multithreaded
Secured
High Performance
Platform Independent
Interpreted
Robust
Dynamic
Architecture Neutral
Portable

Java Platform Independent

- Java is platform independent because it is different from other languages like C, C++ etc. which are compiled into platform specific machines while Java is a write once, run anywhere language.
- A platform is the hardware or software environment in which a program runs.
- There are two types of platforms software-based and hardware-based. Java provides software-based platform.

- Java code can be run on multiple platforms e.g. Windows, Linux, Sun Solaris, Mac/OS etc. Java code is compiled by the compiler and converted into Bytecode. This Bytecode is a platform-independent code because it can be run on multiple platforms i.e. Write Once and Run Anywhere(WORA).

- 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).

- This code can be executed on any system that implements the Java Virtual Machine

- The execution of Bytecode by the JVM is the easiest way to create truly portable programs.

Java Virtual Machine (JVM)

- A Java virtual machine is a software that is capable of executing Java byte code.
- Code for the JVM is stored in .class or .jar
- Each of file contains at most one public class.
- JVM enables the Java application to be platform independent.

JAVA – THE PLATFORM

- Java has a large API (application programming interface) covering a wide range of areas. The following list of Java APIs and applications from Sun show the range of applications of Java .
- For reference http://java.sun.com/products/index.html
- Java Foundation Classes (JFC) – GUI
- JDBC Database Access
- Java Web Server
- Embedded Java - Java on embedded devices

JAVA IDE

- Creating, Compiling, Debugging and Execution for these four steps JDK is not user friendly. IDE is provided for that.
- A list of IDEs are:
  Eclipse.
  Netbeans
  IntelliJ IDEA

Using JDK you can compile and run java program from command line.

c:> javac HelloWorld. Java
  Compiling here and it will produce HelloWorld.class i.e. bytecode.

c:>java HelloWorld

  It runs java byte code on native machine

AN EXAMPLE HELLOWORLD

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