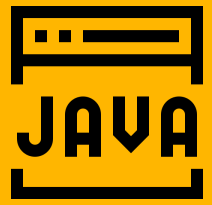


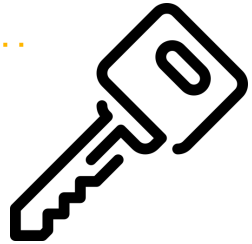
History of Java

Hello!

I am Tariq Hook

Today I will be your instructor

You can find me on github @code-rhino



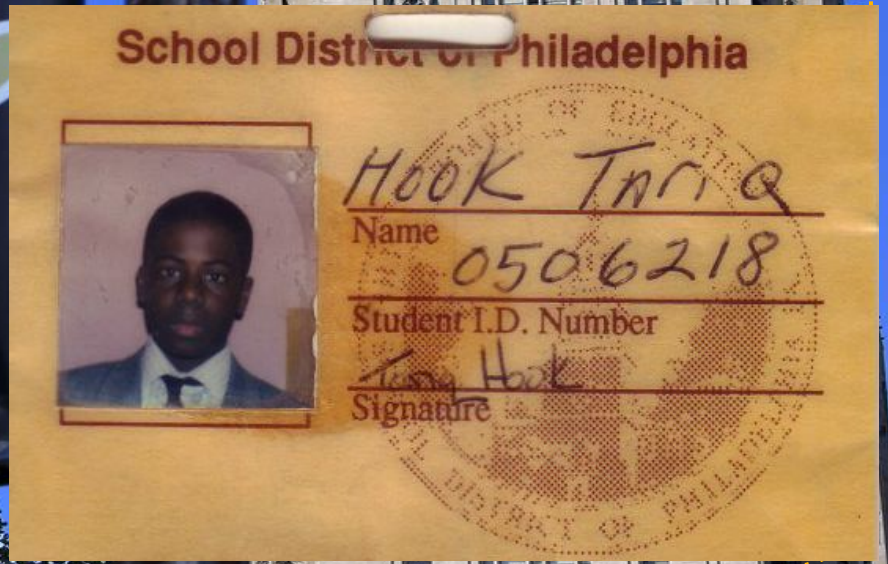
Key Terms

- Garbage Collection
- Distributed System
- Architecturally Neutral
- Java Virtual Machine (JVM)
- Multi-Threaded
- Interpreted
- Just-In-Time
- HEAP

Java Version 1.0

In 1996 Java 1.0 was released at Sun World in Beautiful Las Vegas

1996 Graduated



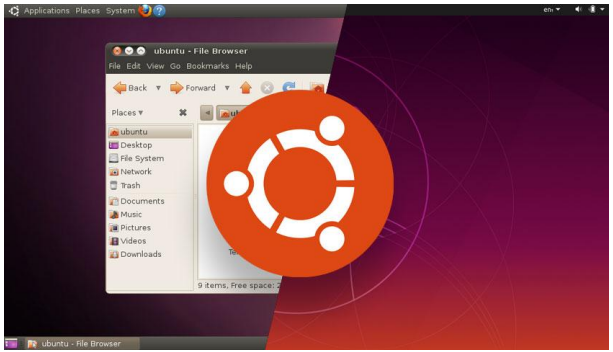
Java is a Platform

Java is not just a language it is a platform which consist of:

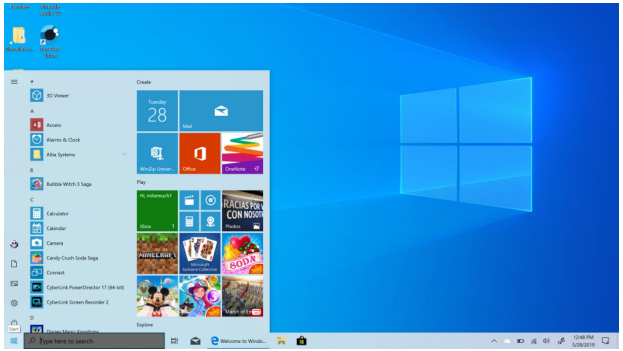
- Libraries
- Reusable Code
- Provided Services
- Garbage Collection

Other Platforms

Linux



Windows



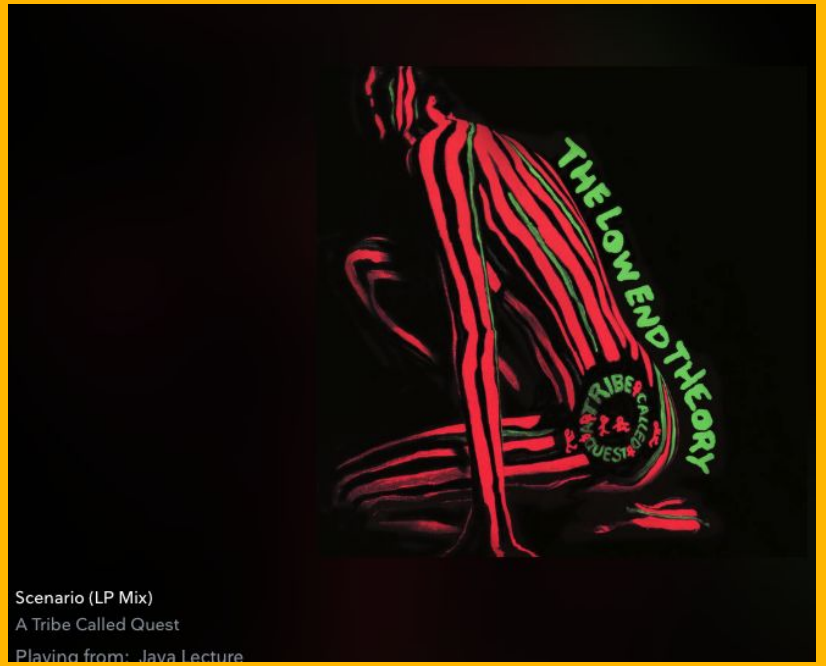
OSX

What is Garbage Collection?

Garbage Collection

The garbage collector is a program that runs on the Java Virtual Machine, which gets rid of objects which are not being used by a Java application anymore. It is a form of automatic memory management.

1991
A Low End Theory



Sun Microsystems

The original Java project was started in 1991 by Sun Micro Systems engineers:

- Patrick Naughton
- James Gosling (Father of Java)

Sun Microsystems

They wanted to develop a programming language that could be used for consumer devices like:

1. Microwaves
2. Calculators
3. Cable Boxes

These devices do not have a lot of memory, so the code executed had to be small and memory efficient.

Project Green

The original name of this project was "Project Green"

The idea was to create a portable language that would generate intermediate code that could be run on any device via a hypothetical or **virtual machine**.

This lead to the creation of a device called the **Java Virtual Machine**.

Virtual Machine

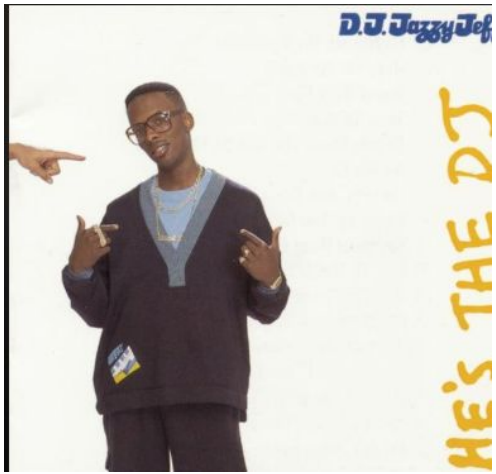
In computing, a virtual machine (VM) is an emulation of a computer system.

Virtual machines are based on computer architectures, and provide functionality of a physical computer.

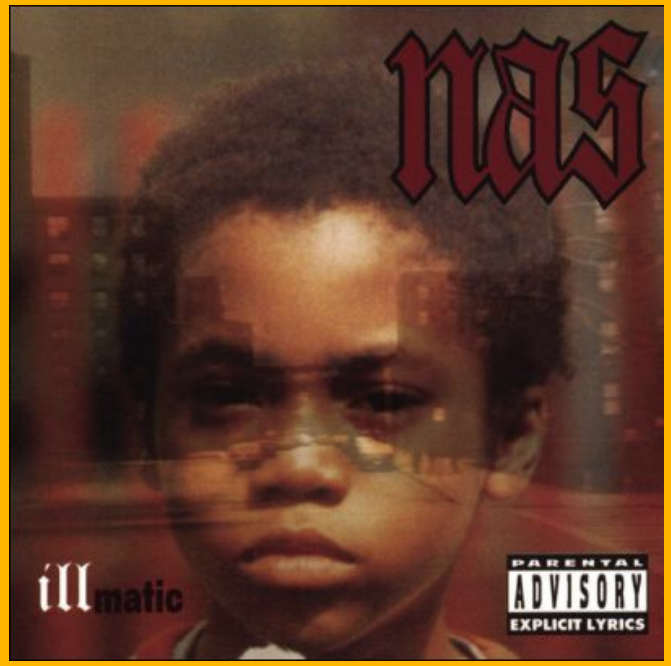
Their implementations may involve specialized hardware, software, or a combination.

Parents Just Don't Understand

The **Green Project** spent years trying to market **Java** as the language of choice for small devices, but they could not get any development teams inside of Sun to adopt it.



1994
Illmatic
Naz



WWW

In 1994, the **World Wide Web** part of the internet was expanding, which at the time consisted of simple browsers translating static hyper-text pages.

1995 Liquid Swords

Hot Java Browser

In 1995, Patrick Naughton and Jonathane Payne created the **Hot Java Browser** to show off the capabilities of **Java 1.0**.

This was shown at **Sun World 95** on May 23rd and it's why **Java** is so big now.

The Java Family

Java has a few versions:

- SE - Standard Edition
- ME - Micro Edition
- EE - Enterprise Edition

A Java Timeline

1996 Java 1.0

Endtroducing by DJ Shadow

1997 Java 1.1

Inner Classes
Jewelz by O.C.

1998 Java 1.2

Aquemini by **Outkast**

2000 Java 1.3

Deltron 3030 by **Deltron 3030**

2002 Java 1.4

Assertions

The Lost Tapes by Nas

2004 Java 5.0

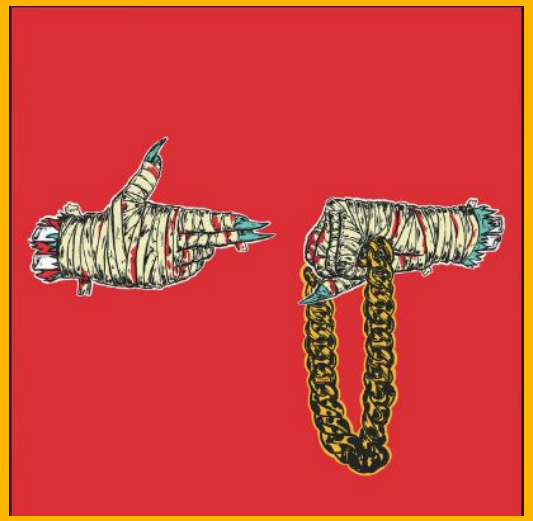
Generics, Loops, Autoboxing
Madvillany by Madvillan

2006 Java 6.0

Donuts by J.Dilla

2011 Java 7.0

Switch with Strings
Section.80 by Kendrick



2014 Java 8.0

Lambda Expressions
Run the Jewels 2 by RTJ

Java is Simple

Java is a subset of C++.

It is designed to be a more straightforward version of this language.

Java is Object Oriented

Object Oriented Languages are designed to focus on the problem being solved, and not the tools used to do it.

Java is designed with objects in mind, which allows flexibility in the creation of a solution.

Java is Distributed

Java is designed to work over a network, and intended to enhance communication between systems.

Java is Robust

Java is a strictly typed language, it is designed to detect potential problems at compile time.

Java is Secure

Java design was always focused around facilitating communication over a network, so security has always been a priority.

Java runs inside of the **HEAP**, which is a memory sandbox- that allows Java to run in isolation inside a machine.

Java is Architecturally Neutral

Java code is compiled and translated into byte code.

This byte code is executed on the **JVM**. The **JVM** can run on any platform or operating system.

Java is Portable

Java is designed to run on a vast number of devices.

Its specification standardised its data types to allow it to provide predictable functionality across devices.

Java is High Performance

Java uses a **just-in-time** compiler.

As your application is running, the interpreter can identify code that is used frequently and compile that down to machine-code to optimize and increase performance.

Java is multi-threaded

Java is designed with agility at the forefront. It is capable of executing actions concurrently.

Java is Dynamic

Due to its ability to go between interpreted and compiled execution.

Java has the ability to dynamically update its programs during runtime via **reflection**.