

1. Introduction to Java Language

1. Welcome to Java

Java Language

Java is a high-level object oriented programming language originally developed by Sun Microsystems and released in 1995 as a core component of Java platform.

Principles

Primary goals in the creation of the Java language:

1. simple, object-oriented and familiar
2. robust and secure
3. architecture-neutral and portable
4. high performance
5. interpreted, threaded, and dynamic

Portability

- Portability means that computer programs written in the Java language must run similarly on any hardware/operating-system platform.
- Sun's slogan: “**Write Once - Run Anywhere**”

Portability Implementation

- Portability is achieved by compiling the Java language code to an intermediate representation called **Java bytecode**, instead of directly to platform-specific machine code.
- Java bytecode instructions are analogous to machine code, but are intended to be interpreted by a **virtual machine** (JVM) written specifically for the host hardware.

Principles

Primary goals in the creation of the Java language:

1. simple, object-oriented and familiar
2. robust and secure
3. architecture-neutral and portable
4. high performance
5. interpreted, threaded, and dynamic

Performance (1 of 2)

- Java programs' execution speed improved significantly with the introduction of Just-in-time (**JIT**) **compilation** in 1997/1998 for Java 1.1
- The addition of language features supporting better code analysis and optimizations in the Java Virtual Machine itself.

Performance (2 of 2)

- December 2012 - microbench-marks show Java 7 is approximately 1.44 times slower than C++
- It is very difficult to quantify the performance difference between C++ and Java in general terms

Principles

Primary goals in the creation of the Java language:

1. simple, object-oriented and familiar
2. **robust** and secure
3. architecture-neutral and portable
4. high performance
5. interpreted, threaded, and dynamic

Robustness

- Java eliminates pointers (first of all C/C++ style pointer arithmetic)
- Java has a very simple memory model where every object is allocated on the heap and all variables of object types are references.
- Memory management is handled through integrated automatic garbage collection performed by the JVM

Principles

Primary goals in the creation of the Java language:

1. simple, object-oriented and familiar
2. robust and **secure**
3. architecture-neutral and portable
4. high performance
5. interpreted, threaded, and dynamic

Java's Security

- JVM controls Java program execution. This ensures flexible security system because each operation that surpass program privilege leads to exception.
- The Java security APIs span a wide range of areas, including cryptography, public key infrastructure, secure communication, authentication, and access control.

Versions History

- JDK 1.0 (January 23, 1996)
- JDK 1.1 (February 19, 1997)
- J2SE 1.2 (December 8, 1998)
- J2SE 1.3 (May 8, 2000)
- J2SE 1.4 (February 6, 2002)
- J2SE 5.0 (September 30, 2004)
- Java SE 6 (December 11, 2006)
- Java SE 7 (July 28, 2011)
- Java SE 8 (March 18, 2014)

Java Platform

- **The Java programming language** is a high-level object-oriented language that has a particular syntax and style
- **A Java platform** is a particular environment in which Java programming language applications run
- **A Java platform = Java Virtual Machine + API**

Java Platforms

- **Java SE** (Standard Edition).
- **Java EE** (Enterprise Edition).
- **Java ME** (Micro Edition).
- **Java Card**
- **Java FX**

Java SE

- virtual machine
- API provides the core functionality of the Java programming language (collections, I/O, security, multithreading, networking, database access, graphical user interface (GUI), XML parsing)
- development tools
- deployment technologies
- class libraries and toolkits commonly used in Java technology applications

Java EE

- Is built on top of the Java SE platform
- Provides an API and runtime environment for developing and running enterprise applications:
 - large-scale
 - multi-tiered
 - scalable
 - reliable
 - secure
 - network

Java ME

- Running Java programming language applications on small devices
- A Java ME API is a subset of the Java SE API, along with special class libraries useful for small device application development
- Java ME applications are often clients of Java EE platform services

Java Card

- Allows Java-based applications to be run securely on smart cards and similar small memory footprint devices
- It is widely used in SIM cards (used in mobile phones) and ATM cards
- Java Card technology was originally developed for the purpose of securing sensitive information stored on smart cards

Java FX

- Creating rich internet applications using a lightweight user-interface API
- Use hardware-accelerated graphics and media engines to take advantage of higher-performance clients and a modern look-and-feel
- APIs for connecting to networked data sources.
- JavaFX applications may be clients of Java EE platform services

Resources

- <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
— last Java version download (1.8.31)
- <http://www.oracle.com/technetwork/java/index>

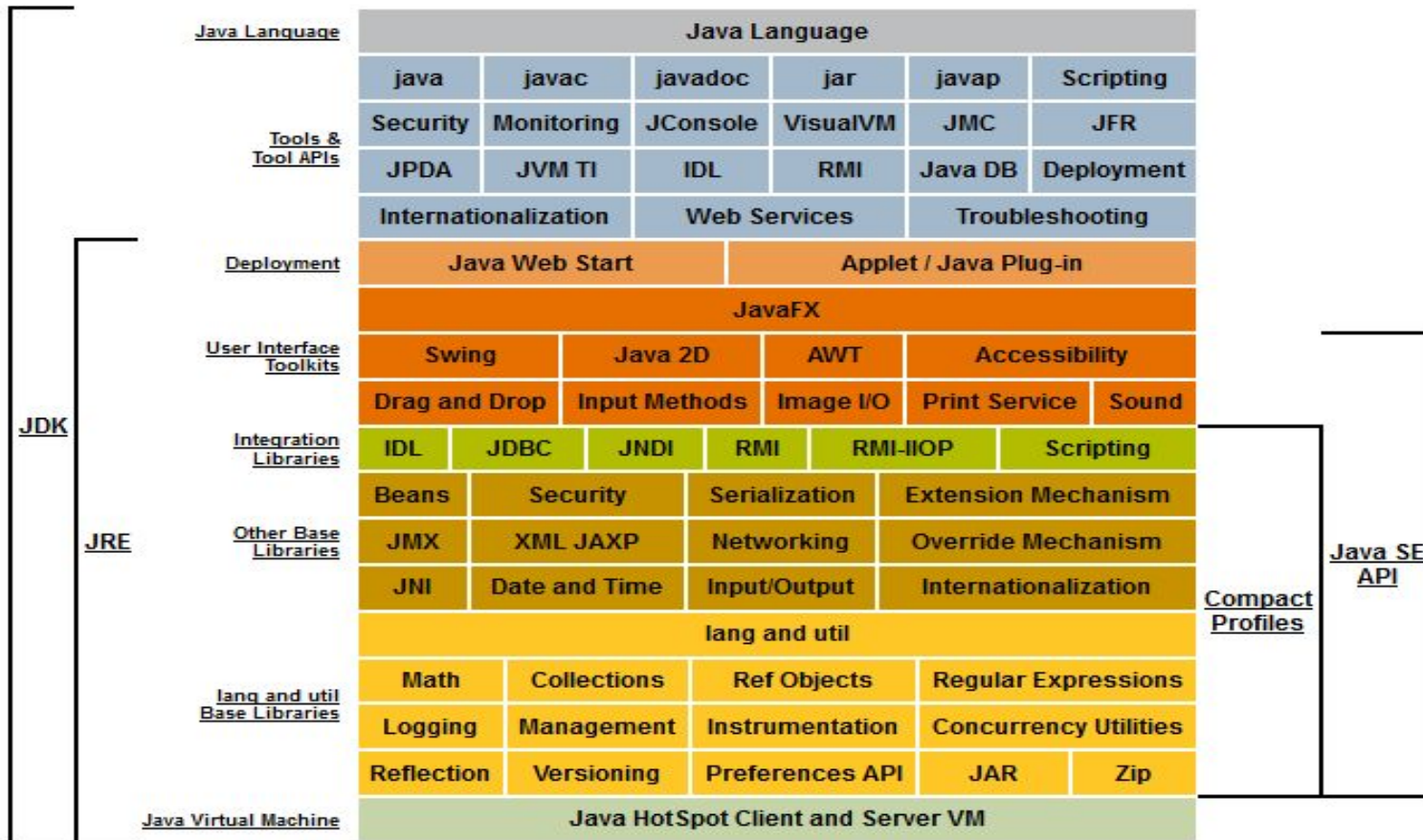
Documentation and Manuals

- <http://docs.oracle.com/javase/tutorial/> - Java 8 tutorials
- <http://docs.oracle.com/javase/8/docs/> - Java 8 docs
- <http://docs.oracle.com/javase/7/docs/> - Java 7 docs
- Core Java, eighth edition by Cay S. Horstmann and Gary Cornell, 2008; ISBN 978-0-13-235476-9
- Thinking in Java by Bruce Eckel, 4th ed , 2006; ISBN 0131872486

Manuals

- Хорстманн Кей С., Корнелл Гари.
Библиотека профессионала. Java 2.
Том 1. Основы. ISBN: 978-5-8459-1378-4
Том 2. Тонкости программирования.
ISBN: 978-5-8459-1482-8
8-е издание, «Диалектика-Вильямс», 2014.
- Брюс Эккель. Философия Java. – 4-е
издание, «Питер», 2009. ISBN
978-5-388-00003-3

Java 8 docs



Java 8 Tutorials

- Basics
- Additional Subjects
- Special Subjects

Java 8 Tutorials. Basics

- [Getting Started](#)
- [Learning the Java Language](#)
- [Essential Classes](#)
- [Collections](#)
- [Concurrency](#)
- [Generics](#)
- [JDBC](#)

Java8 Tutorials. Additional Subjects

- [Deployment](#)
- [Internalization](#)
- [JavaBeans](#)
- [The Reflection API](#)
- [Security](#)
- [Java API for XML Processing \(JAXP\)](#) Java API for XML Processing (JAXP [JAXB](#))

Java 8 Tutorials. Special Subjects

- [Creating a GUI with Swing](#)
- [2D Graphics](#)
- [Full-Screen Exclusive Mode API](#)
- [Sound](#)
- [The Extension Mechanism](#)

Java 8 Tutorials. Special Subjects II

- [Java Management Extensions \(JMX\)](#)
- [Java Naming and Directory Interface \(JNDI\)](#)
- [Remote Method Invocation \(RMI\)](#)
- [Sockets Direct Protocol](#)
- [Preparation for Java Programmer Language Certification](#)

Check Java Installation

- java – version
- Output:
java version “1.8.31”
- Directory structure:
 - C:\Program Files\Java
 - jdk1.8.31
 - jre8