

# Table of Contents

Java ..... 3

*Hello World* ..... 6

*Benefits of Java* ..... 6

*TIOBE Index* ..... 7



# Java

Java (programming language)	
Full Name	Java
Short Name	Java
Description	Designed to allow the creation of programs that can run on several different platforms.
Company	Oracle
Web	Yes
Mobile	Yes
Enterprise	Yes
Embedded	No

## What is Java?

Java is a popular programming language and computing platform that was developed by Sun Microsystems and is now owned by Oracle. It is designed to be platform-independent and to run on a wide range of devices and operating systems.

## What are some features of Java?

Java includes features such as automatic memory management, garbage collection, and a robust standard library. It also supports object-oriented programming, multithreading, and networking.

## What platforms can Java be used on?

Java can be used on a wide range of platforms, including Windows, macOS, Linux, and mobile devices such as Android. It can also be used on embedded devices, such as those found in cars, appliances, and other consumer electronics.

## What types of applications can be built with Java?

Java can be used to build a wide range of applications, including desktop applications, web applications, mobile apps, games, and more. It is particularly well-suited for building large-scale, enterprise-level applications.

## Is Java difficult to learn?

Java is designed to be easy to learn for developers who are already familiar with programming concepts. However, like any programming language, it may take some time to become proficient in using Java effectively. There are many resources available to help developers learn Java, including online tutorials, books, and courses.

## Is Java a popular language?

Yes, Java is a very popular language that is widely used in industry and academia. It is particularly popular in enterprise-level applications, where its reliability, scalability, and security are highly valued. The language has a large and active community of developers, which has contributed to its growth and success.

Snippet from [Wikipedia: Java \(programming language\)](#)

**Java** is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let programmers *write once, run anywhere* (WORA), meaning that compiled Java code can run on all platforms that support Java without the need to recompile. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages.

Java gained popularity shortly after its release, and has been a very popular programming language since then. Java was the third most popular programming language in 2022 according to GitHub. Although still widely popular, there has been a gradual decline in use of Java in recent years with other languages using JVM gaining popularity.

Java was originally developed by James Gosling at Sun Microsystems. It was released in May 1995 as a core component of Sun's Java platform. The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses. As of May 2007, in compliance with the specifications of the Java Community Process, Sun had relicensed most of its Java technologies under the GPL-2.0-only license. Oracle offers its own HotSpot Java Virtual Machine, however the official reference implementation is the OpenJDK JVM which is free open-source software and used by most developers and is the default JVM for almost all Linux distributions.

As of September 2023, Java 21 is the latest version, which is also a long-term support (LTS) version. Java 8, 11, and 17 are previous LTS versions still officially supported.

[Creative Commons Attribution-Share Alike 4.0](#)

### GitHub Topics

- <https://github.com/topics/java>

Java was originally developed as an alternative to the C/C++ programming languages. It is now mainly used for building web, desktop, mobile, and embedded applications. Java is owned and licensed through Oracle, with free and open source implementations available from Oracle and other vendors.

## Free Programming Books

- [3D Programming in Java](#) - Daniel Selman (PDF)
- [Apache Jakarta Commons: Reusable Java Components](#) - Will Iverson (PDF)
- [Artificial Intelligence - Foundations of Computational Agents, Second Edition](#) - David L. Poole and Alan K. Mackworth
- [Building Back-End Web Apps with Java, JPA and JSF](#) - Mircea Diaconescu, Gerd Wagner (HTML,PDF)
- [Category wise tutorials - J2EE](#) - Yong Mook Kim
- [Core Servlets and JavaServer Pages, 2nd Ed. \(2003\)](#) - Marty Hall and Larry Brown ([archived](#))
- [Data Structures in Java for the Principled Programmer \(2007\)](#) - Duane A. Bailey (PDF)
- [Google's Java Style Guide](#)
- [Introduction to Computer Science "booksite"](#) - by Robert Sedgewick and Kevin Wayne (HTML)
- [Introduction to Computer science using Java](#) - Bradley Kjell
- [Introduction to Programming in Java](#) - Robert Sedgewick and Kevin Wayne
- [Introduction to Programming Using Java](#) - David J. Eck (HTML, PDF, ePub + exercises)
- [Introduction to Programming Using Java \(5th Edition - final version, 2010 Jun\)](#) - David J. Eck (HTML, PDF, ePub + exercises)
- [Java Application Development on Linux \(2005\)](#) - Carl Albing and Michael Schwarz (PDF)
- [Java, Java, Java Object-Oriented Problem Solving](#) - R. Morelli and R. Walde
- [Java Language and Virtual Machine Specifications](#) - James Gosling, et al
- [Java Masters](#) - Java Masters (HTML)
- [Java Notes for Professionals](#) - Compiled from StackOverflow documentation (PDF)
- [Java Programming](#) - Wikibooks
- [Java Programming for Kids](#) - Yakov Fain
- [Java Projects, Second Edition](#) - Peter Verhas (Packt account *required*)
- [Learning Java Language](#) - Compiled from StackOverflow Documentation (PDF)
- [Microservices Best Practices for Java](#) (PDF)
- [Object-Oriented Programming in Java™ Textbook](#) - Rick Halterman (PDF per Chapter)
- [OOP - Learn Object Oriented Thinking & Programming](#) - Rudolf Pecinovsky (PDF)
- [Open Data Structures \(in Java\)](#) - Pat Morin (PDF)
- [Playing with Java Microservices on Kubernetes and OpenShift](#) - Nebrass Lamouchi (*Leanpub account or valid email requested*)
- [Processing XML with Java \(A Guide to SAX, DOM, JDOM, JAXP, and TrAX\) \(2002\)](#) - Elliotte Rusty Harold
- [The Java EE6 Tutorial](#) (PDF)
- [The Java EE7 Tutorial](#) - Eric Jendrock, et al (PDF)
- [The Java Tutorials](#)
- [The Java Web Scraping Handbook](#) - Kevin Sahin (PDF, HTML)
- [Think Data Structures: Algorithms and Information Retrieval in Java](#) - Allen B. Downey (PDF, HTML)
- [Think Java: How to Think Like a Computer Scientist, 2nd Edition](#) - Allen B. Downey and Chris Mayfield (HTML, PDF) ([Interactive version by Trinket](#))
  - [Think Java: How to Think Like a Computer Scientist](#) - Allen B. Downey and Chris Mayfield (HTML, PDF) ([Interactive version by Trinket](#))
- [Using RxJava 2 Tutorial](#) - Lars Vogel, Simon Scholz (HTML)
- [Welcome to Java for Python Programmers](#) - Brad Miller
- [Welcome to the Java Workshop \(2006\)](#) - Trevor Miller

- [What's New in Java 8](#) - Adam L. Davis
- [Writing Advanced Applications for the Java 2 Platform](#) - Calvin Austin and Monica Pawlan (PDF)

Source: [EbookFoundation](#)

## Hello World

```
class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Source: [Wikibooks](#)

## Benefits of Java

Benefit	Description
Platform independence	Java code can run on any platform without any modifications, as long as a Java Virtual Machine (JVM) is available for that platform. This makes Java a very portable language.
Object-oriented programming	Java is an object-oriented language, which means that it is designed to be modular, flexible, and scalable. This makes it well-suited for building complex, large-scale applications.
Automatic memory management	Java includes a garbage collector that automatically manages memory allocation and deallocation. This helps to prevent memory leaks and makes it easier to write stable and reliable code.
Large standard library	Java includes a large and comprehensive standard library that provides a wide range of functionality, including networking, file I/O, and user interface development. This makes it easier to build complex applications without having to reinvent the wheel.
Robust security	Java has built-in security features that help to protect against a wide range of security threats, such as buffer overflows, injection attacks, and denial-of-service attacks.
Strong community	Java has a large and active community of developers who contribute to the development of the language, as well as to the creation of tools and libraries that make it easier to use. This community also provides support and resources for developers who are learning Java.
High performance	Java is designed to be fast and efficient, which makes it well-suited for building high-performance applications. It includes features such as just-in-time compilation and bytecode optimization that help to improve performance.

Benefit	Description
<b>Multithreading</b>	Java includes built-in support for multithreading, which allows for concurrent execution of code. This makes it easier to build scalable applications that can take advantage of multiple processors or cores.
<b>Easy to learn</b>	Java has a simple and easy-to-learn syntax that is similar to other popular programming languages such as C++ and C#. This makes it a good choice for beginners who are just starting to learn programming.
<b>Open-source</b>	Java is an open-source language, which means that the source code is freely available and can be modified and distributed by anyone. This has led to the development of many open-source tools and libraries that make it easier to use Java.
<b>Backward compatibility</b>	Java is designed to be backward compatible, which means that new versions of the language can run code that was written for older versions. This helps to ensure that Java applications will continue to run even as new versions of the language are released.
<b>Versatile</b>	Java can be used to build a wide range of applications, from small command-line tools to large-scale enterprise applications. It is also used for developing Android mobile apps and can be used in web development with Java-based frameworks such as Spring and Struts.

- Developer-friendly - Simple
- Object-Oriented
- Cross Platform
- Garbage collector
- Memory allocation
- Multithreaded
- Exception handling
- Portability

## TIOBE Index

- <https://www.tiobe.com/tiobe-index/java/>

### External links:

- [https://en.wikibooks.org/wiki/Java\\_Programming](https://en.wikibooks.org/wiki/Java_Programming)

## ## ToDo ##

- Object-Oriented Programming (OOP)
- Classes
- Inheritance
- Polymorphism
- Interfaces
- Abstract classes
- Encapsulation
- Packages

- Exceptions
- Generics
- Collections
- Threads
- Reflection
- Annotations
- Streams
- JDBC (Java Database Connectivity)
- JPA (Java Persistence API)
- Spring Framework
- JUnit (Java Unit Testing)
- Maven (Java Build Tool)
- Gradle (Java Build Tool)

See also: [Programming Languages](#)

From:

<https://almbok.com/> - **ALMBoK.com**

Permanent link:

<https://almbok.com/dev/java>

Last update: **2023/05/01 17:20**

