

Was ist Java?

Java ist eine beliebte Programmiersprache, die 1995 entwickelt wurde.

Sie gehört zu Oracle, und mehr als 3 Milliarden Geräte verwenden Java.



Kurze Historie von Java

- **Entstehungsdatum:** Java wurde 1991 von James Gosling und seinem Team bei Sun Microsystems entwickelt und 1995 offiziell veröffentlicht.

- **Warum wurde Java entwickelt?**

Java entstand mit dem Ziel, eine plattformunabhängige Programmiersprache zu schaffen.

Die Idee war, Software einmal zu schreiben und sie auf verschiedenen Geräten ohne Anpassungen lauffähig zu machen – bekannt durch das Motto:

"Write once, run anywhere."

Einsatzgebiete von Java:

- Mobile Anwendungen (insbesondere Android-Apps)
- Desktop-Anwendungen
- Webanwendungen
- Webserver und Anwendungsserver
- Spiele
- Datenbankverbindungen
- Und vieles, vieles mehr!

Warum Java verwenden?

- Java funktioniert auf verschiedenen Plattformen (Windows, Mac, Linux, Raspberry Pi, usw.)
- Es ist eine der beliebtesten Programmiersprachen der Welt
- Es gibt eine hohe Nachfrage nach Java im aktuellen Arbeitsmarkt
- Es ist leicht zu erlernen und einfach zu benutzen
- Es ist Open-Source und kostenlos
- Es ist sicher, schnell und leistungsstark
- Es gibt eine riesige Community (Zehntausende von Entwicklern)

- Java ist eine objektorientierte Sprache, die Programmen eine klare Struktur gibt und die Wiederverwendung von Code ermöglicht, wodurch Entwicklungskosten gesenkt werden
- Da Java C++ und C# ähnlich ist, fällt es Programmierern leicht, zwischen diesen Sprachen zu wechseln

Wie funktioniert Java?

Java ist so entworfen, dass Programme unabhängig vom Betriebssystem oder Gerät ausgeführt werden können. Das Herzstück dieses Prinzips ist die **Java Virtual Machine (JVM)**.

Ablauf der Programmausführung in Java

1. Schreiben des Codes

Entwickler schreiben den Programmcode in **.java-Dateien**.

2. Kompilierung

Der **Java-Compiler (javac)** übersetzt den Quellcode nicht direkt in Maschinencode, sondern in eine plattformunabhängige Zwischensprache, den **Bytecode**.

- Ergebnis: **.class-Dateien**, die den Bytecode enthalten.

3. Ausführung durch die JVM

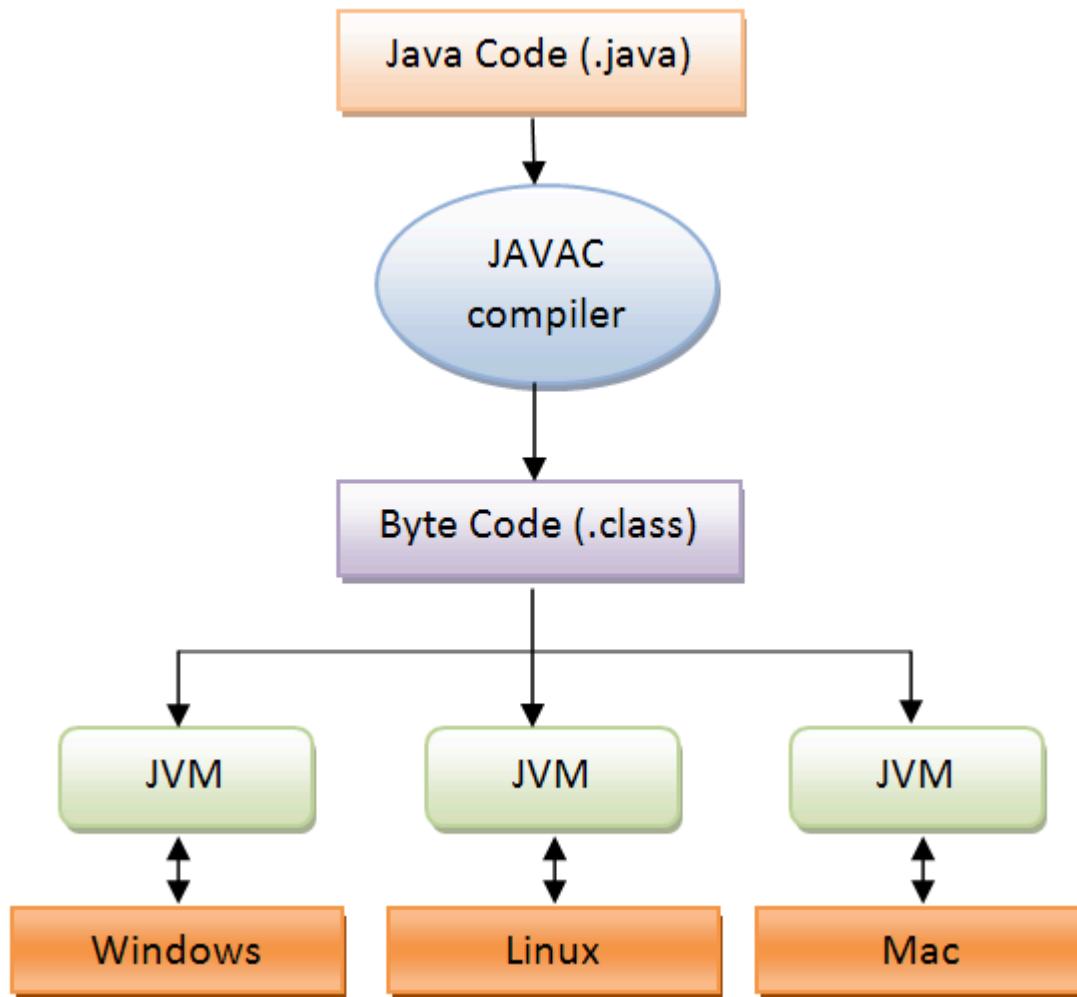
Die **Java Virtual Machine (JVM)** liest den Bytecode und übersetzt ihn **zur Laufzeit** in den jeweiligen Maschinencode des aktuellen Systems (Windows, Linux, MacOS, etc.). Dadurch ist der gleiche Bytecode auf allen Plattformen lauffähig.

Warum ist Java plattformunabhängig?

- Der Bytecode ist immer derselbe, egal auf welchem System entwickelt wurde.
- Jede Plattform hat ihre eigene JVM-Implementierung, die den Bytecode korrekt ausführt.
- Damit gilt das Prinzip: "**Write once, run anywhere.**"

Vorteile dieses Ansatzes

- **Portabilität:** Programme laufen auf nahezu jedem Gerät mit einer JVM.
- **Sicherheit:** Der Bytecode läuft in einer kontrollierten Umgebung, was Schutz vor direkten Zugriffen auf das Betriebssystem bietet.
- **Flexibilität:** Entwickler müssen sich nicht um unterschiedliche Hardware oder Betriebssysteme kümmern.



Wie werde ich ein guter Java Entwickler?



© Sarah Andersen