



# Erste Schritte mit Java

Eine wichtige Entscheidung auf dem Weg zum Programmierer haben Sie bereits mit der Wahl dieses Buches getroffen: die Entscheidung für die Programmiersprache Java. Obwohl viele wichtige Grundkonzepte des Programmierens unabhängig von der gewählten Programmiersprache sind, gibt es doch Unterschiede. Dieses Kapitel wird einige der Besonderheiten von Java erklären und warum die Sprache eine gute Wahl für Programmierneinsteiger ist. Zuvor erhalten Sie aber zunächst eine Einführung in einige der allgemeinen Grundbegriffe und Konzepte des Programmierens.

## 1.1 Programmiersprachen

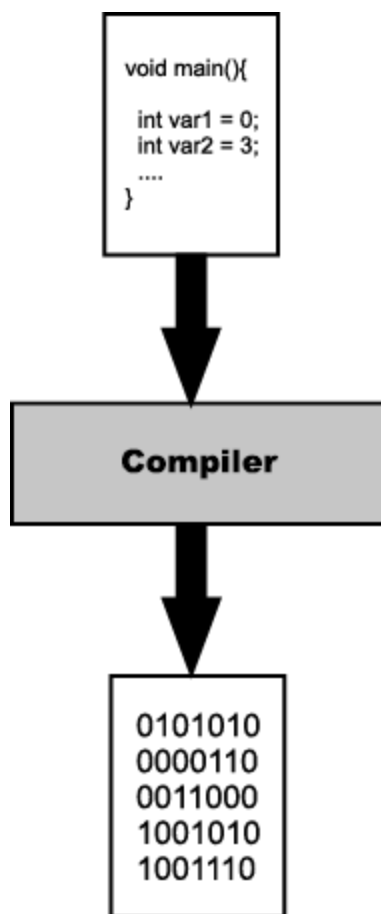
Computer können uns beim Lösen zahlreicher Aufgaben helfen, oft sind sie dabei sogar besser als der Mensch, zum Beispiel beim Rechnen oder dem Sortieren von Daten. Um eine Aufgabe erledigen zu können, benötigt der Computer eine genaue Anleitung zur Lösung eben jener Aufgabe. Eine solche Anleitung mit exakten Handlungsvorschriften für einen Computer nennt man auch *Algorithmus*.

Handlungsvorschriften gibt es nicht nur für Computer, sondern auch für Menschen, zum Beispiel in Form von Rezepten in Kochbüchern oder Aufbauanleitungen für Möbelstücke. Allerdings sprechen Mensch und Maschine unterschiedliche Sprachen, und das aus gutem Grund. Unsere menschlichen Sprachen, egal ob Deutsch, Englisch oder Plansprachen wie Esperanto, sind vage. Eine »Prise Salz«, zum Beispiel, ist eine vollkommen normale Angabe in einem Rezept und bringt keinen Koch ins Schwitzen. Für einen Backroboter aber wäre eine solche Angabe nutzlos, sie ist nicht eindeutig. Unsere menschlichen Sprachen sind voll von solch mehrdeutigen Aussagen. Denken Sie zum Beispiel an den Satz: »Da vorne ist eine Bank.« Ein vollständiger und korrekter deutscher Satz, aber was ist gemeint? Befindet sich »da vorne« eine Parkbank zum Sitzen oder ein Kreditinstitut zum Einzahlen und Abheben von Geld?

Ein Algorithmus muss aus *eindeutigen* Handlungsvorschriften bestehen. Daher sind natürliche Sprachen, wie Deutsch und Englisch, ungeeignet zur Beschreibung von Computerprogrammen. Die »Muttersprache« von Computern sind Einsen und Nullen,

Strom an oder aus. Eine Sprache unverständlich für Menschen. Daher wurden Programmiersprachen entwickelt, als Zwischenrepräsentation. Im Gegensatz zu unserer natürlichen Sprache sind Programmiersprachen nicht doppeldeutig, gleichzeitig sind sie, obwohl sie auf den ersten Blick kompliziert scheinen können, relativ einfach für den Menschen zu verstehen.

Damit der Computer die Programmiersprache versteht, muss sie trotzdem übersetzt werden, in sogenannte *Maschinsprache*. Diese Übersetzung geschieht automatisch mit einem Programm, das den Namen *Compiler* trägt, abgeleitet vom englischen Verb »to compile«, was so viel bedeutet wie »zusammensetzen« oder »erstellen«. Das Ergebnis des *Kompilierens* sind sogenannte *Binärdateien*, die die Maschinsprache, codiert in Nullen und Einsen, enthalten, wie in Abbildung 1.1 gezeigt.

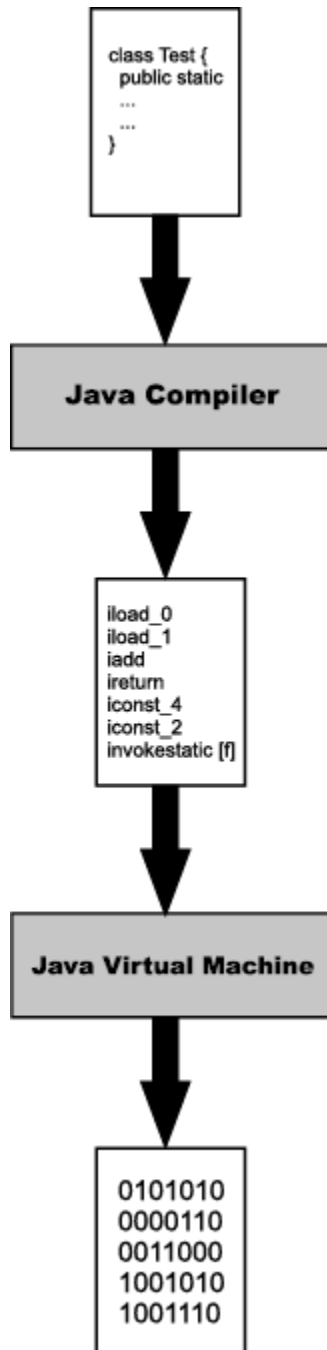


**Abb.1.1:** Funktionsweise eines Compilers

Für die Übersetzung von menschlicher Sprache in Programmiersprache gibt es leider noch keine automatische Übersetzung, dafür sind Sie als Programmierer zuständig, und wie das funktioniert, lernen Sie mit diesem Buch.

## 1.2 Besonderheiten von Java

Als Programmierer haben Sie die Wahl zwischen Hunderten Programmiersprachen. Einige davon, wie zum Beispiel C, Python oder JavaScript, sind weit verbreitet und werden häufig eingesetzt, andere eher obskur und nur für spezielle Einsatzzwecke oder Hardware geeignet. Java gehört zur Gruppe der populären Sprachen mit zahlreichen Einsatzgebieten. Dabei bietet jede Programmiersprache ihre eigenen Vor-, aber auch Nachteile. Eine »richtige« oder »beste« Programmiersprache gibt es nicht – je nach Anwendungsfall kann der Einsatz einer anderen Programmiersprache sinnvoll sein.



**Abb.1.2:** Funktionsweise des Java-Compilers

Eine der wichtigsten Eigenschaften von Java ist die Tatsache, dass Programme, die in Java geschrieben wurden, auf allen gängigen Plattformen und Betriebssystemen, also insbesondere Windows, Linux und macOS, ausgeführt werden können. Im Vergleich dazu kann zum Beispiel ein C-Programm, das unter Windows kompiliert wurde, auch nur unter Windows ausgeführt werden. Um diese Plattformunabhängigkeit zu

ermöglichen, verwendet Java eine weitere Abstraktionsebene zwischen der Programmiersprache und der Maschinsprache, den sogenannten *Bytecode*.

Der Java-Compiler wandelt den Programmcode, wie in Abbildung 1.2 gezeigt, zunächst in diese Zwischenrepräsentation um. Dieser Bytecode kann dann von einem weiteren Programm, einem sogenannten *Interpreter*, ausgeführt werden. Der Java-Interpreter wird auch *Java Virtual Machine (JVM)* genannt. So muss nur ein Programm, nämlich die JVM, auf das jeweilige Betriebssystem angepasst werden und sofort können sämtliche Java-Programme ausgeführt werden.

Allerdings hat dieser Vorteil auch seinen Preis. Die JVM muss den Bytecode in Echtzeit – das heißt, während das Programm ausgeführt wird – in Maschinsprache übersetzen. Bei einem klassischen Compiler ist das nicht notwendig. Das führt dazu, dass Java, aber auch beispielsweise Python, das nach demselben Prinzip funktioniert, in der Ausführung weniger schnell ist als zum Beispiel die Programmiersprache C. Für sehr ressourcenintensive Anwendungen, zum Beispiel 3D-Grafik, ist Java deshalb häufig nicht die beste Wahl.

Eine weitere Besonderheit von Java ist die strikte *Typisierung* und *Objektorientierung*. Was genau es damit auf sich hat, das werden Sie in den folgenden Kapiteln erfahren. Das ist ein Nachteil für Programmierneulinge, den Java mit sich bringt. Gerade am Anfang gibt es ein gewisses Grundgerüst für jedes Programm, das Sie einfach als gegeben hinnehmen müssen, bis Sie Schritt für Schritt besser verstehen werden, was dahintersteckt. Dafür lernen Sie aber von Beginn an wichtige Grundlagen explizit kennen, die andere Programmiersprachen, wie zum Beispiel Python oder JavaScript, vor Ihnen verstecken. Deshalb ist Java vielleicht nicht die einfachste Sprache zum Programmierenlernen, aber trotzdem eine sehr gute Wahl für den Einstieg. Nicht umsonst ist Java elementarer Bestandteil des Informatikstudiums an den meisten Universitäten.



Außer dem Namen haben die beiden Programmiersprachen Java und JavaScript nicht viel gemein. JavaScript, eine Programmiersprache, die ursprünglich hauptsächlich zum Einsatz auf Websites entwickelt wurde, hieß zunächst auch LiveScript. Durch eine Partnerschaft zwischen dem ursprünglichen Entwickler von LiveScript und dem Entwickler von Java, Sun Microsystems, kam es zur namentlichen Annäherung.

Java ist universell einsetzbar. Es wird zum Beispiel häufig zur Programmierung von betrieblichen Anwendungen verwendet, unter anderem, da es einfach ist, ein Programm über mehrere Rechner zu verteilen, was zum Beispiel dabei hilft, steigende Hardwareanforderungen bei wachsenden Nutzerzahlen zu erfüllen. Ebenso lassen sich Android-Apps mit Java programmieren – ja selbst die im Juni 2003 von der NASA gestartete Raumsonde »Spirit« bewegte sich mithilfe von Java über die Oberfläche des Mars.