

LEHRBUCH

Sebastian Dörn

Python lernen in abgeschlossenen Lerneinheiten

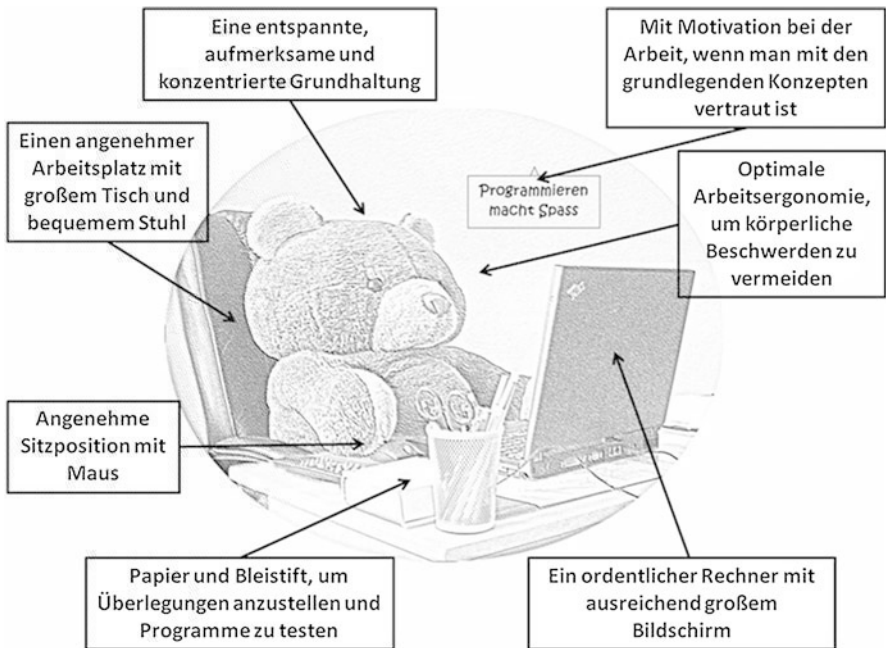
Programmieren für Einsteiger
mit vielen Beispielen

2. Auflage



Springer Vieweg

Python lernen in abgeschlossenen Lerneinheiten



Programmieren für Einsteiger mit vielen Beispielen

Sebastian Dörn

Python lernen in abgeschlossenen Lerneinheiten

Programmieren für Einsteiger
mit vielen Beispielen

2., verbesserte Auflage

 Springer Vieweg

Sebastian Dörn
Hochschulcampus Tuttlingen
Tuttlingen, Deutschland

Dieses Buch basiert auf dem Buch „Java lernen in abgeschlossenen Lerneinheiten“.

ISBN 978-3-658-28975-1 ISBN 978-3-658-28976-8 (eBook)
<https://doi.org/10.1007/978-3-658-28976-8>

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Springer Vieweg

© Springer Fachmedien Wiesbaden GmbH, ein Teil von Springer Nature 2019, 2020

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von allgemein beschreibenden Bezeichnungen, Marken, Unternehmensnamen etc. in diesem Werk bedeutet nicht, dass diese frei durch jedermann benutzt werden dürfen. Die Berechtigung zur Benutzung unterliegt, auch ohne gesonderten Hinweis hierzu, den Regeln des Markenrechts. Die Rechte des jeweiligen Zeicheninhabers sind zu beachten.

Der Verlag, die Autoren und die Herausgeber gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag, noch die Autoren oder die Herausgeber übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen. Der Verlag bleibt im Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutionsadressen neutral.

Springer Vieweg ist ein Imprint der eingetragenen Gesellschaft Springer Fachmedien Wiesbaden GmbH und ist ein Teil von Springer Nature.

Die Anschrift der Gesellschaft ist: Abraham-Lincoln-Str. 46, 65189 Wiesbaden, Germany

Vorwort

Können Sie mit Ihrem Computer sprechen, damit er Ihnen bei der Arbeit hilft? Wenn nicht, dann sprechen Sie vermutlich nicht seine Sprache. Unser Freund, der Computer, versteht kein Deutsch, Englisch, Französisch oder Spanisch. Seine bevorzugten Sprachen sind Java, C/C++, Python, JavaScript, PHP, SQL oder R. Alle Geräte, Maschinen oder Apps, die wir täglich verwenden, werden durch Hunderttausende oder Millionen von Codezeilen gesteuert.

Die Kunst des Programmierens ist in unserer digitalen Informationsgesellschaft eine Schlüsseldisziplin. Die Kreativität besteht darin, für eine Maschine eine Art Handlungsanweisung zu erstellen, um eine spezielle Aufgabe auszuführen. Das Ergebnis ist die Software in Form eines Quellcodes in einer Programmiersprache. Programmieren ist deutlich mehr als das Eintippen von einzelnen Codezeilen. Ein zentraler Aspekt beim Programmieren ist es, korrekten und sinnvoll strukturierten Code zu schreiben. Der Programmierer muss das Programm so schreiben, dass einzelne Abschnitte wiederverwendbar sind. Wenn wir das nicht tun, erhalten wir sogenannten Spaghetticode – ein verworrenes Stück Programmcode. Selbst erfahrene Programmierer können dieses undurchdringliche Gewirr kaum nachvollziehen.

In der digitalen Welt wird es für viele Tätigkeitsfelder immer bedeutender, dass Menschen in der Lage sind, mit Maschinen zu kommunizieren. Programmieren muss heutzutage neben Lesen, Schreiben und Rechnen eine Grundfertigkeit des Menschen darstellen. Eine qualifizierte Programmiergrundausbildung ist für viele Mitarbeiter heute unerlässlich, um bei der Planung, Konzeption und Entwicklung von modernen Produkten und Dienstleistungen mitzuwirken. Um dem Computer mitzuteilen, was er für uns tun soll, müssen wir dessen spezifische Regeln erlernen. Die heutige Arbeitswelt ist durch eine starke Zunahme von programmierbaren Systemen geprägt, die nur durch Programmierkenntnisse effizient bedienbar sind.

Mitarbeiter müssen heute in der Lage sein, sich oft wiederholende Aufgabenstellungen mit Hilfe von Computerprogrammen zu automatisieren. Viele verschwenden jedoch Stunden mit Tippen und Klicken in Office-Anwendungen, um immer die gleichen Routineaufgaben zu bearbeiten. In der Unternehmenspraxis gibt es dafür unzählige Beispiele: Zusammensuchen und Aufbereiten von Daten aus verschiedenen Quellen, Erstellen von Berichten mit Abbildungen und Tabellen oder das Versenden von vielen E-Mails mit vorformulierten Texten.

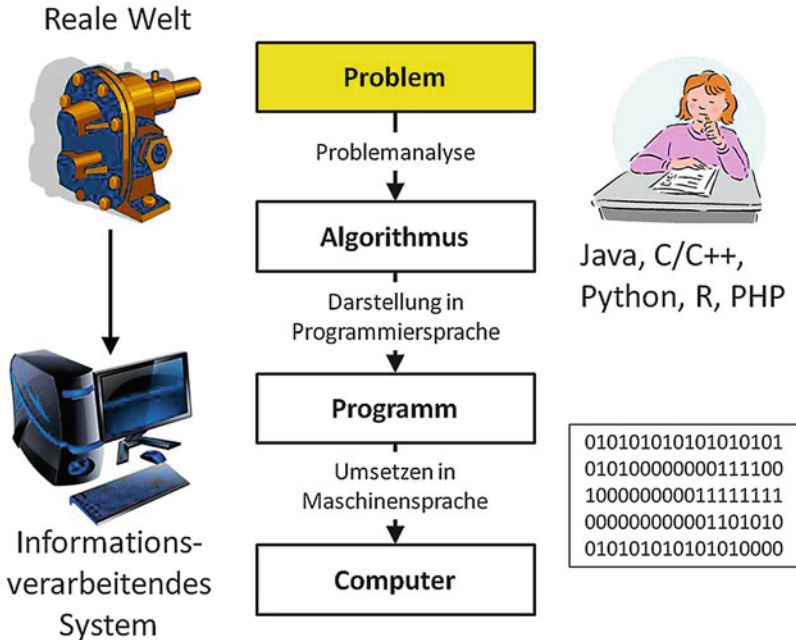


Abb. 1 Das Grundprinzip der Programmierung vom Problem über den Algorithmus zum Computerprogramm

Viele dieser Aufgaben sind extrem zeitaufwendig und so speziell, dass dafür keine fertigen Softwareprodukte existieren. Meistens ist nur wenig Programmierarbeit erforderlich, damit der Computer die notwendigen Daten zusammensucht, sinnvolle Auswertungen erstellt und das ganze automatisch in einen Bericht überträgt. Auf diese Weise lassen sich aufwendige Arbeitspläne oder Qualitätsberichte vollkommen automatisch erzeugen. Wenn sich die Datenbestände ändern, können sie durch einen Knopfdruck neu erstellt werden.

Wie funktioniert Programmieren?

Das Grundprinzip der Programmierung ist in Abb. 1 dargestellt. Am Anfang steht immer ein Problem, das mit Hilfe eines Computerprogramms zu lösen ist. Nach einer sorgsam Problemanalyse wird ein Algorithmus für diese Aufgabenstellung entworfen. Algorithmen sind geniale Ideen in Form von speziellen Handlungsanweisungen. Ein Computerprogramm ist die konkrete Umsetzung des Algorithmus in eine Programmiersprache. Das Programm besteht aus einer Reihe von Anweisungen, die dem Rechner vorgeben, was er zu tun hat.

Eine Programmiersprache ist eine Sprache zur Formulierung von Algorithmen. Ähnlich einer natürlichen Sprache gibt es bei einer Programmiersprache einen bestimmten Wortschatz (Schlüsselwörter) und eine Grammatik (Syntax), nach deren Regeln der Programmcode zu bilden ist. Die Sätze in einer Programmiersprache sind die syntaktisch korrekten Anweisungen. Programmierer erstellen den

sogenannten Quellcode mit Hilfe eines Texteditors. Die einzelnen Anweisungen des Codes ergeben sich aus dem zugehörigen Algorithmus. Die Informatiker sprechen beim Erstellen eines Programmiercodes von der Implementierung. Die geschriebenen Programme werden anschließend durch einen Compiler automatisiert in die Maschinensprache des jeweiligen Rechners übersetzt. Damit ist unser Algorithmus von einer Maschine ausführbar.

Es gibt heute eine große Vielfalt von unterschiedlichen Programmiersprachen für die unterschiedlichsten Aufgaben: Entwurf von Softwaresystemen und Apps, Auswerten von statistischen Datensätzen, Erzeugen von Steuerungsinformationen für Werkzeugmaschinen, Programmieren von Mikroprozessoren, Abfrage von Datenbanken, Erstellen von graphischen Benutzeroberflächen oder zur Konzeption von Internetseiten.

Warum Programmieren lernen?

In unserer digitalen Gesellschaft und Arbeitswelt gibt es viele verschiedene Gründe, die Fähigkeit des Programmierens zu erlernen:

1. Die heutigen Arbeitsumgebungen sind durch eine starke Zunahme von programmierbaren Systemen geprägt, die nur durch Programmierkenntnisse effizient bedienbar sind.
2. Für viele Wissensarbeiter gehört es heute zu einer Schlüsselqualifikation, gewisse sich oft wiederholende Aufgabenstellungen mit Hilfe von Computerprogrammen zu automatisieren.
3. Eine qualifizierte Programmiergrundausbildung ist für viele Mitarbeiter heute unerlässlich, um bei der Planung, Konzeption und Entwicklung von modernen Produkten und Dienstleistungen mitzuwirken.
4. Die Software gewinnt gegenüber der Hardware, sodass die Wertschöpfung in Unternehmen nur durch die Kombination von Hardware mit zusätzlichen digitalen Serviceleistungen auf Basis von Daten und Algorithmen zu sichern ist.
5. In der digitalen Welt wird es für viele Tätigkeitsfelder immer bedeutender sein, dass Menschen in der Lage sind, mit Maschinen zu kommunizieren.
6. Die Maschinensprachen werden in Zukunft wichtiger als die gesprochenen Sprachen, da sich die Kommunikation zwischen Menschen deutlich einfacher automatisieren lässt, als die Kommunikation zwischen Mensch und Maschine.
7. Die Analyse von Daten mit Computerprogrammen ist eine Tätigkeit, die für Ingenieure, Naturwissenschaftler, Mediziner, Psychologen und Soziologen von herausragender Bedeutung ist.
8. Im Zeitalter der Digitalisierung ist digitales Grundwissen für viele Berufszweige von Relevanz, um neuartige Konzepte und Technologien kompetent zu vermitteln (Lehrer), zu berichten (Journalisten), zu beurteilen (Politiker) oder zu entscheiden (Manager).
9. Programmierkenntnisse und Algorithmen sind zentrales Grundlagenwissen für das maschinelle Lernen, das heute im Zentrum der digitalen Wirtschaft steht.
10. Programmieren macht Spaß.

Warum Python lernen?

Python ist eine universelle Programmiersprache, die zahlreiche Programmierparadigmen unterstützt. Der Name Python ist von der britischen Komikergruppe Monty Python hergeleitet. Python-Programmierer werden als „Pythonistas“ bezeichnet und die Dokumentation besitzt viele Anspielungen auf Monty Python, sowie auch auf Schlangen. Python gibt es seit 1991 und ist ein offenes kostenfreies Entwicklungsmodell, das durch die gemeinnützige Python Software Foundation gestützt wird. Python ist plattformunabhängig und der Code läuft unmodifiziert auf allen wichtigen Betriebssystemen. Ähnlich wie Java oder C# erzeugt Python mit einem Compiler einen sogenannten Byte-Code. Dieser Byte-Code wird mit Hilfe einer virtuellen Maschine des jeweiligen Betriebssystems ausgeführt.

Die Sprache Python ist leicht zu erlernen, sehr flexibel und wird durch die umfangreiche Standardbibliothek häufig als Skriptsprache verwendet. In vielen Fällen ist es mit geringem Aufwand möglich, praktische kleine Programme zu schreiben. Python bietet die Möglichkeit zur objektorientierten Programmierung und zum Schreiben von umfangreicheren Softwaresystemen. Heute ist Python einer der beliebtesten und meistgenutzten Programmiersprachen. Die Programmiersprache Python ist insbesondere im Bereich der Datenanalyse und des maschinellen Lernens in der Praxis weit verbreitet. Mit Hilfe des Webframeworks Django lassen sich dynamische Webseiten erstellen. In Python-Programme ist es möglich vorhandenen C-, C++- und Fortran-Code zu integrieren. Python eignet sich gut für das Programmieren des Raspberry PI.

Wie lernen Sie Programmieren?

Das Programmieren kann viel Spaß bereiten, wenn Sie sich mit den allgemeinen Prinzipien und der Syntax der jeweiligen Sprache auskennen. In vielen Programmiersprachen wie Python ist es möglich, bereits mit wenigen einfachen Mitteln sinnvolle Programme zu schreiben. Die Grundlagen der Programmierung mit ihren Grundkonzepten ändern sich nur sehr wenig. Viele Bücher im Bereich der Programmiersprachen leiden unter dem „Semikolon-Syndrom“, d. h. der Leser wird in die Grundlagen einer speziellen Programmiersprache eingeführt. Bei dieser Art der Einführung in die Programmierung beschäftigen Sie sich weniger mit den Konzepten des Programmierens, sondern mehr mit den Eigenschaften einer bestimmten Programmiersprache. Diese Syntaxeigenschaften werden dann oftmals bis ins letzte Detail behandelt, was viele Anfänger überfordert, da sie die zugehörigen Anwendungsbereiche noch nicht kennen.

Beachten Sie, dass Sie das richtige Programmieren nicht in einer Woche oder einem Monat lernen. Wie Sportler ihre Muskeln trainieren, trainieren Programmierer ihr logisches und abstraktes Denken. Es kommt dabei gar nicht darauf an, was wir programmieren, sondern nur, dass wir programmieren. In der Regel benötigt ein Anfänger selbst nach einem Programmierkurs vielleicht einen Monat für die Erstellung eines Programms, das ein sehr guter Programmierer an einem halben Tag schafft. Programmieren besitzt auch einen sozialen Aspekt, denn viele Softwareprodukte entstehen in Teamarbeit von mehreren Programmierern.

Wie ist dieses Buch aufgebaut?

Das Ziel dieses Buches ist, Ihnen das Programmieren in einzelnen abgeschlossenen Lerneinheiten beizubringen. Mit dem Entwurf von effizienten Daten- und Ablaufstrukturen möchte ich Sie in die Lage versetzen, algorithmische Konzepte zu verstehen und in Programmcode umzusetzen. Die zentralen Lernziele sind das Verstehen der Abstraktionskonzepte moderner Programmiersprachen und das Erlernen des logischen und algorithmischen Denkens. Mit diesem Wissen können Sie im Anschluss selbstständig eigene Computerprogramme implementieren, um damit praxisrelevante Aufgaben schnell und sicher zu bearbeiten.

Die einzelnen Kapitel dieses Buches sind nach einem einheitlichen Schema aufgebaut:

1. **Einführung:** Kurze Beschreibung der Inhalte des Kapitels mit den zugehörigen Lernzielen.
2. **Konzept:** Vorstellung der syntaktischen Bestandteile und Programmiermethodiken mit zahlreichen Beispielen.
3. **Beispiele:** Ausführliche Beschreibung einiger ausgewählter Beispielprogramme zur Festigung der Lerninhalte.
4. **Zusammenfassung:** Zusammenfassung der vorgestellten Konzepte zum schnellen Überblick.

Dieses einheitliche Schema soll Ihnen als Leser helfen, die ersten Schritte in die Welt des Programmierens zu setzen. Großen Wert lege ich in diesem Buch auf Programmierbeispiele, damit Sie die einzelnen Konzepte begreifen und nachvollziehen können¹

Programmieren ist nur dann erfolgreich, wenn Sie . . .

- . . . **die allgemeinen Prinzipien und die Syntax der jeweiligen Sprache kennen.**

Das genaue Lesen und Verstehen der Konzepte sowie das Ausprobieren der vorgestellten Beispiele ist wichtig, um Programmieren richtig zu erlernen.

- . . . **wissen, dass Anfänger sehr lange für das Schreiben von Programmen benötigen.**

Programmieranfänger sollten sich nicht entmutigen lassen, wenn sie am Anfang sehr lange brauchen, um ein Computerprogramm zu schreiben (Faktor 1 zu 100 oder mehr).

- . . . **selbstständig versuchen, Computerprogramme zu entwerfen und zu schreiben.**

Das Programmieren lernen Sie nicht durch Lesen oder Zuschauen, sondern nur durch eigenständige Arbeit am Programmcode.

¹Dieses Buch basiert auf dem gleichnamigen Java-Buch und beinhaltet dadurch Überschneidungen in Inhalt.