

# 1 Einführung

## 1.1 Was sind Python-Tricks?

*Python-Trick: Ein kurzes Python-Codefragment, das als Lernmittel gedacht ist. Ein Python-Trick dient entweder zur einfachen Veranschaulichung eines Aspekts von Python oder als motivierendes Beispiel, um es Dir zu ermöglichen, tiefer zu schürfen und ein intuitives Verständnis zu entwickeln.*

Der Ausgangspunkt dieses Buches war eine kurze Reihe von Codeausschnitten, die ich eine Woche lang auf Twitter veröffentlicht hatte. Zu meiner Überraschung ernteten sie begeisterte Rückmeldungen und wurden noch tagelang weitergegeben.

Immer mehr Entwickler fragten mich, wo sie »die komplette Reihe« bekommen könnten. In Wirklichkeit hatte ich einfach nur einige wenige Tricks zu verschiedenen Python-Themen zusammengestellt. Dahinter steckte kein umfassendes Konzept; es war einfach nur ein kleines Twitter-Experiment.

Diese Anfragen machten mir jedoch klar, dass es sich lohnen würde auszuprobieren, meine kurzen Codebeispiele als Lernmittel zu verwenden. So stellte ich eine Reihe weiterer Python-Tricks zusammen und veröffentlichte sie in Form einer Serie von E-Mails. Nach nur wenigen Tagen hatten sich mehrere Hundert Python-Entwickler dafür registriert. Von dieser Reaktion war ich schier überwältigt.

In den folgenden Tagen und Wochen erhielt ich einen nicht enden wollenden Zustrom an Rückmeldungen von Python-Entwicklern. Sie dankten mir dafür, dass ich ihnen Aha-Erlebnisse über Aspekte der Sprache bescherte, mit deren Verständnis sie zu kämpfen hatten. Es war großartig, diese Kommentare zu hören. Während ich die Python-Tricks lediglich für Codeausschnitte gehalten hatte, zogen viele Entwickler großen Nutzen daraus.

Zu diesem Zeitpunkt entschied ich, mein Python-Tricks-Experiment zu forcieren und es zu einer Serie von ca. 30 E-Mails auszubauen. Jede dieser Mails bestand nur aus einer Überschrift und einem Screenshot mit einem Codeausschnitt, aber schon bald erkannte ich die Einschränkungen, die dieses Format mit sich brachte. In einer E-Mail brachte ein blinder Python-Entwickler seine Enttäuschung darüber zum Ausdruck, dass die Python-Tricks als Bilder geliefert wurden, sodass er sie mit seinem Screenreader nicht lesen konnte.

Um mein Projekt ansprechender zu gestalten und einem größeren Publikum zugänglich zu machen, musste ich also mehr Zeit darin investieren. Also setzte ich mich hin und schrieb die ganze Serie von Python-Tricks-E-Mails in Textformat mit übersichtlicher Syntaxhervorhebung in HTML um. Diese neue Inkarnation der Python-Tricks lief eine Weile sehr gut. Laut den Rückmeldungen, die ich bekam, waren die Entwickler froh darüber, dass sie die Codebeispiele nun einfach kopieren konnten, um selbst damit herumzuspielen.

Als sich immer mehr Entwickler für diese E-Mail-Serie registrierten, begann ich in den Kommentaren und Fragen, die mir geschickt wurden, ein Muster zu erkennen. Einige Tricks funktionierten als motivierende Beispiele sehr gut für sich allein. Bei den komplizierteren Beispielen fehlte jedoch eine erklärende Stimme, um die Leser anzuleiten oder sie auf weitere Quellen hinzuweisen, sodass sie ein tieferes Verständnis gewinnen konnten.

Dies war ein weiterer Bereich, in dem Verbesserungen notwendig waren. Der Leitspruch von *dbader.org* lautet: »Python-Entwicklern helfen, noch großartiger zu werden.« Hier bot sich mir offensichtlich eine Gelegenheit, um diesem Motto gerecht zu werden.

Ich entschied mich, auf der Grundlage der besten und wertvollsten Python-Tricks meines E-Mail-Kurses ein neuartiges Python-Buch zu schreiben:

- Ein Buch, das die faszinierendsten Aspekte der Sprache anhand von kurzen und leicht verständlichen Beispielen erklärt
- Ein Buch, das eine Reihe großartiger Python-Merkmale aufzeigt und die Motivation der Leser stets auf hohem Niveau hält
- Ein Buch, das dich an die Hand nimmt und anleitet, um dir zu helfen, dein Verständnis von Python zu vertiefen

Dieses Buch war mir eine Herzensangelegenheit und auch ein enormes Experiment. Ich hoffe, dass du bei der Lektüre viel Freude hast und etwas über Python lernst.

– Dan Bader

## 1.2 Der Zweck dieses Buches

Dieses Buch soll aus dir einen besseren – einen erfolgreichereren und kenntnisreicheren – Python-Programmierer machen. Vielleicht fragst du dich, wie es dir dabei helfen kann. Es handelt sich eben nicht um ein Python-Tutorial oder einen Einsteigerkurs. Wenn du Python erst lernst, wird dieses Buch dich nicht zu einem professionellen Python-Entwickler machen. Die Lektüre wird zwar auch dann von Vorteil für dich sein, aber du musst dir auch noch anderes Material anschauen, um grundlegende Python-Kenntnisse zu entwickeln.

Um den größten Nutzen aus diesem Buch zu ziehen, solltest du bereits über Python-Kenntnisse verfügen, die du erweitern möchtest. Am besten ist es, wenn du schon eine Weile in Python programmierst und bereit bist, in die Tiefe zu gehen, deine Kenntnisse abzurunden und deinen Code »pythonischer« zu machen.

Dieses Buch ist auch hervorragend für dich geeignet, wenn du schon Erfahrungen mit anderen Programmiersprachen hast und dich schnell in Python einarbeiten möchtest. Du wirst hier einen wahren Schatz an praktischen Tipps und Entwurfsmustern finden, die dir helfen, ein erfolgreicherer und qualifizierterer Python-Programmierer zu werden.

### 1.3 Wie du dieses Buch lesen solltest

Die beste Möglichkeit, dieses Buch zu lesen, besteht darin, sich ihm wie einem Büffet zu nähern. Jeder Python-Trick steht für sich selbst, weshalb du einfach zu denen springen kannst, die dich am meisten interessieren. Das ist die Vorgehensweise, die ich empfehle. Du kannst natürlich auch alle Python-Tricks in der Reihenfolge lesen, in der sie im Buch erscheinen. Dadurch verpasst du nichts. Wenn du auf der letzten Seite angekommen bist, kannst du sicher sein, dass du alle gesehen hast.

Einige der Tricks sind unmittelbar verständlich, weshalb du sie problemlos in deine tägliche Arbeit übernehmen kannst, nachdem du einfach das entsprechende Kapitel gelesen hast. Bei anderen Tricks ist etwas mehr Zeit für das Verständnis vonnöten. Solltest du Schwierigkeiten damit haben, einen bestimmten Trick in deinen eigenen Programmen anzuwenden, ist es oft hilfreich, die entsprechenden Codebeispiele im Python-Interpreter durchzuspielen. Wenn es dann immer noch nicht klick macht, wende dich getrost an mich, sodass ich dir helfen und die Erklärungen verbessern kann. Langfristig nützt das nicht nur dir, sondern allen Pythonistas, die das Buch lesen.