

Punkt 73: `heapq` für  
Prioritätswarteschlangen verwenden

Punkt 74: `memoryview` und `bytearray` für  
Zero-Copy-Interaktionen mit `bytes`  
verwenden

## **Kapitel 9: Testen und Debuggen**

Punkt 75: Debuggen mit `repr`-Strings

Punkt 76: Ähnliches Verhalten mit  
`TestCase`-Unterklassen überprüfen

Punkt 77: Tests mit `setUp`, `tearDown`,  
`setUpModule` und `tearDownModule`  
voneinander abschotten

Punkt 78: Mocks für Tests von Code mit  
komplexen Abhängigkeiten verwenden

Punkt 79: Abhängigkeiten kapseln, um das  
Erstellen von Mocks und Tests zu  
erleichtern

Punkt 80: Interaktives Debuggen mit pdb

Punkt 81: Nutzung des Arbeitsspeichers und Speicherlecks mit tracemalloc untersuchen

## **Kapitel 10: Zusammenarbeit**

Punkt 82: Module der Python-Community

Punkt 83: Virtuelle Umgebungen zum Testen von Abhängigkeiten

Punkt 84: Docstrings für sämtliche Funktionen, Klassen und Module

Punkt 85: Pakete zur Organisation von Modulen und zur Bereitstellung stabiler APIs verwenden

Punkt 86: Modulweiter Code zur Konfiguration der Deployment-Umgebung

Punkt 87: Einrichten einer Root-

Exception zur Abschottung von Aufruffern und APIs

Punkt 88: Zirkuläre Abhängigkeiten auflösen

Punkt 89: warnings für Refactoring und Migration verwenden

Punkt 90: Statische Analyse mit typing zum Vermeiden von Bugs

# Stimmen zum Buch

»Ich habe dieses Buch seit dem Erscheinen der ersten Ausgabe 2015 mit Begeisterung empfohlen. Die neue Ausgabe, die für Python 3 erweitert und aktualisiert wurde, ist eine wahre Fundgrube an Erfahrungen der Python-Programmierung in der Praxis, von denen Programmierer auf jedem Niveau profitieren.«

**Wes McKinney –**  
*Begründer des Python-Projekts Pandas,*  
*Leiter von Ursa Labs*

»Wenn Sie vorher in einer anderen Sprache

programmiert haben, finden Sie mit diesem Buch eine Anleitung, die einzigartigen Features, die Python zu bieten hat, zu Ihrem vollen Vorteil zu nutzen. Ich verwende Python schon seit fast zwanzig Jahren, habe aber dennoch eine Menge nützlicher Tricks gelernt, insbesondere was die neuen Features in Python 3 betrifft. Das Buch ist randvoll mit in der Praxis verwertbaren Ratschlägen und definiert gewissermaßen, was die Community meint, wenn von typisch Python-artigem Code die Rede ist.«

**Simon Willison –  
*Mitbegründer von Django***

»Ich programmiere seit Jahren in Python und dachte eigentlich, dass ich die Sprache ziemlich gut kenne. Dank dieser Fundgrube