

# Funktionale Programmierung verstehen

Konzepte und Entwurfsmuster für guten Code

» Hier geht's  
direkt  
zum Buch

# DAS VORWORT

# Vorwort

Die funktionale Programmierung (FP) hat in den letzten Jahren an Popularität gewonnen. Viele Unternehmen suchen FP-versierte Entwicklerinnen und Entwickler, da auch Sprachen, die ursprünglich nicht für die funktionale Programmierung konzipiert waren, im Laufe der Zeit funktionale Merkmale angenommen haben. Dazu zählen zum Beispiel Java, JavaScript, Python und andere. Die Nachfrage beruht teilweise auf einer wahrgenommenen Verbesserung des Entwicklungsprozesses – anscheinend verspricht der funktionale Ansatz wenige Fehler und einen erweiterbaren und robusten Code. Ob das stimmt oder nicht – und ob ein größerer Anteil des in den nächsten Jahren geschriebenen Codes funktional sein wird –, muss sich noch zeigen. Im Moment sollten wir die funktionale Programmierung als eines von mehreren Paradigmen mit jeweils ihren eigenen Stärken und Schwächen ansehen.

## Für wen ist dieses Buch geeignet?

Dieses Buch richtet sich an alle Entwicklerinnen und Entwickler, unabhängig von ihrem Kenntnisstand in der funktionalen Programmierung. Sowohl Neulinge als auch erfahrene FP-Programmiererinnen werden nützliche Informationen finden. Das Buch behandelt die kategorientheoretischen Grundlagen der funktionalen Programmierung, die in anderen Büchern oft nicht thematisiert werden. Auch wenn Sie bereits Erfahrung mit FP haben, aber Ihr Verständnis der zugrunde liegenden Konzepte und Theorien vertiefen möchten, werden Sie viel Nützliches in diesem Buch finden.

## Wie ist dieses Buch aufgebaut?

Dieses Buch zeigt anhand verschiedener Programmiersprachen, wie funktionale Konstrukte unseren Code verbessern können. Für die Codebeispiele wird jedoch häufig Scala verwendet, da funktionale Konzepte in dieser Sprache besonders leicht ausgedrückt werden können.

Sie werden die Konzepte der funktionalen Programmierung leichter erlernen und begreifen, wenn sie auf natürliche Weise ausgedrückt werden, und genau das ermöglicht Scala. Eine kleine Einführung in Scala finden Sie in Anhang A, »Scala«.