

# ATOMIC KOTLIN

Bruce  
Eckel

Svetlana  
Isakova

# Atomic Kotlin

**Bruce Eckel and Svetlana Isakova**

This book is for sale at <http://leanpub.com/AtomicKotlin>

This version was published on 2020-12-21



\* \* \* \* \*

This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

\* \* \* \* \*

© 2020 Mindview LLC

ISBN for EPUB version: 978-0-9818725-4-4

ISBN for MOBI version: 978-0-9818725-4-4

# Table of Contents

[Copyright](#)

## [Section I: Programming Basics](#)

[Introduction](#)

[Why Kotlin?](#)

[Hello, World!](#)

[var & val](#)

[Data Types](#)

[Functions](#)

[if Expressions](#)

[String Templates](#)

[Number Types](#)

[Booleans](#)

[Repetition with while](#)

[Looping & Ranges](#)

[The in Keyword](#)

[Expressions & Statements](#)

[Summary 1](#)

## [Section II: Introduction to Objects](#)

[Objects Everywhere](#)

[Creating Classes](#)

[Properties](#)

[Constructors](#)

[Constraining Visibility](#)

[Packages](#)

[Testing](#)

[Exceptions](#)

[Lists](#)

[Variable Argument Lists](#)

[Sets](#)

[Maps](#)

[Property Accessors](#)  
[Summary 2](#)

### [Section III: Usability](#)

[Extension Functions](#)  
[Named & Default Arguments](#)  
[Overloading](#)  
[when Expressions](#)  
[Enumerations](#)  
[Data Classes](#)  
[Destructuring Declarations](#)  
[Nullable Types](#)  
[Safe Calls & the Elvis Operator](#)  
[Non-Null Assertions](#)  
[Extensions for Nullable Types](#)  
[Introduction to Generics](#)  
[Extension Properties](#)  
[break & continue](#)

### [Section IV: Functional Programming](#)

[Lambdas](#)  
[The Importance of Lambdas](#)  
[Operations on Collections](#)  
[Member References](#)  
[Higher-Order Functions](#)  
[Manipulating Lists](#)  
[Building Maps](#)  
[Sequences](#)  
[Local Functions](#)  
[Folding Lists](#)  
[Recursion](#)

### [Section V: Object-Oriented Programming](#)

[Interfaces](#)  
[Complex Constructors](#)  
[Secondary Constructors](#)  
[Inheritance](#)

[Base Class Initialization](#)  
[Abstract Classes](#)  
[Upcasting](#)  
[Polymorphism](#)  
[Composition](#)  
[Inheritance & Extensions](#)  
[Class Delegation](#)  
[Downcasting](#)  
[Sealed Classes](#)  
[Type Checking](#)  
[Nested Classes](#)  
[Objects](#)  
[Inner Classes](#)  
[Companion Objects](#)

## [Section VI: Preventing Failure](#)

[Exception Handling](#)  
[Check Instructions](#)  
[The Nothing Type](#)  
[Resource Cleanup](#)  
[Logging](#)  
[Unit Testing](#)

## [Section VII: Power Tools](#)

[Extension Lambdas](#)  
[Scope Functions](#)  
[Creating Generics](#)  
[Operator Overloading](#)  
[Using Operators](#)  
[Property Delegation](#)  
[Property Delegation Tools](#)  
[Lazy Initialization](#)  
[Late Initialization](#)

## [Appendices](#)

[Appendix A: AtomicTest](#)  
[Appendix B: Java Interoperability](#)

# Copyright

## Atomic Kotlin

By Bruce Eckel, President, MindView, LLC, and Svetlana Isakova, JetBrains sro.

**Copyright ©2021, MindView LLC.**

**eBook ISBN 978-0-9818725-4-4**

**Print Book ISBN 978-0-9818725-5-1**

The eBook ISBN covers the Stepik and Leanpub eBook distributions, both available through *AtomicKotlin.com*.

**Please purchase this book through [www.AtomicKotlin.com](http://www.AtomicKotlin.com), to support its continued maintenance and updates.**

All rights reserved. Printed in the United States of America. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, see AtomicKotlin.com.

Created in Crested Butte, Colorado, USA, and Munich, Germany.

Text printed in the United States

**Ebook: Version 1.0, December 2020**

**First printing January 2021**

Cover design by Daniel Will-Harris, [www.Will-Harris.com](http://www.Will-Harris.com)

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations are printed with initial capital letters or in all capitals.

The Kotlin trademark belongs to [the Kotlin Foundation](http://the Kotlin Foundation). Java is a trademark or registered trademark of Oracle, Inc. in the United States and other countries. Windows is a registered trademark of Microsoft Corporation in the United States and other countries. All other product names and company names mentioned herein are the property of their respective owners.

The authors and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

Visit us at [www.AtomicKotlin.com](http://www.AtomicKotlin.com).

## Source Code

All the source code for this book is available as copyrighted freeware, distributed via [Github](https://github.com). To ensure you have the most current version, this is the official code distribution site. You may use this code in classroom and other educational situations as long as you cite this book as the source.

The primary goal of this copyright is to ensure that the source of the code is properly cited, and to prevent you from republishing the code without permission. (As long as this book is cited, using examples from the book in most media is generally not a problem.)

In each source-code file you find a reference to the following copyright notice:

```
// Copyright.txt
This computer source code is Copyright ©2021 MindView LLC.
All Rights Reserved.
```

```
Permission to use, copy, modify, and distribute this
computer source code (Source Code) and its documentation
without fee and without a written agreement for the
purposes set forth below is hereby granted, provided that
```

the above copyright notice, this paragraph and the following five numbered paragraphs appear in all copies.

1. Permission is granted to compile the Source Code and to include the compiled code, in executable format only, in personal and commercial software programs.

2. Permission is granted to use the Source Code without modification in classroom situations, including in presentation materials, provided that the book "Atomic Kotlin" is cited as the origin.

3. Permission to incorporate the Source Code into printed media may be obtained by contacting:

MindView LLC, PO Box 969, Crested Butte, CO 81224  
MindViewInc@gmail.com

4. The Source Code and documentation are copyrighted by MindView LLC. The Source code is provided without express or implied warranty of any kind, including any implied warranty of merchantability, fitness for a particular purpose or non-infringement. MindView LLC does not warrant that the operation of any program that includes the Source Code will be uninterrupted or error-free. MindView LLC makes no representation about the suitability of the Source Code or of any software that includes the Source Code for any purpose. The entire risk as to the quality and performance of any program that includes the Source Code is with the user of the Source Code. The user understands that the Source Code was developed for research and instructional purposes and is advised not to rely exclusively for any reason on the Source Code or any program that includes the Source Code. Should the Source Code or any resulting software prove defective, the user assumes the cost of all necessary servicing, repair, or correction.

5. IN NO EVENT SHALL MINDVIEW LLC, OR ITS PUBLISHER BE LIABLE TO ANY PARTY UNDER ANY LEGAL THEORY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS, OR FOR PERSONAL INJURIES, ARISING OUT OF THE USE OF THIS SOURCE CODE AND ITS DOCUMENTATION, OR ARISING OUT OF THE INABILITY TO USE ANY RESULTING PROGRAM, EVEN IF MINDVIEW LLC, OR ITS PUBLISHER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. MINDVIEW LLC SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOURCE CODE AND DOCUMENTATION PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, WITHOUT ANY ACCOMPANYING SERVICES FROM MINDVIEW LLC, AND MINDVIEW LLC HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

Please note that MindView LLC maintains a Web site which is the sole distribution point for electronic copies of the Source Code, where it is freely available under the terms stated above:

<https://github.com/BruceEckel/AtomicKotlinExamples>

If you think you've found an error in the Source Code,



please submit a correction at:  
<https://github.com/BruceEckel/AtomicKotlinExamples/issues>

You may use the code in your projects and in the classroom (including your presentation materials) as long as the copyright notice that appears in each source file is retained.

# SECTION I: PROGRAMMING BASICS

*There was something amazingly enticing about programming—***Vint Cerf**

This section is for readers who are learning to program. If you're an experienced programmer, skip forward to [Summary 1](#) and [Summary 2](#).

# Introduction

This book is for dedicated novices and experienced programmers.

You're a novice if you don't have prior programming knowledge, but "dedicated" because we give you just enough to figure it out on your own. When you're finished, you'll have a solid foundation in programming and in Kotlin.

If you're an experienced programmer, skip forward to [Summary 1](#) and [Summary 2](#), then proceed from there.

The "Atomic" part of the book title refers to atoms as the smallest indivisible units. In this book, we try to introduce only one concept per chapter, so the chapters cannot be further subdivided—thus we call them *atoms*.

## Concepts

All programming languages consist of features. You apply these features to produce results. Kotlin is powerful—not only does it have a rich set of features, but you can usually express those features in numerous ways.

If everything is dumped on you too quickly, you might come away thinking Kotlin is "too complicated."

This book attempts to prevent overwhelm. We teach you the language carefully and deliberately, using the following principles:

1. **Baby steps and small wins.** We cast off the tyranny of the chapter. Instead, we present each small step as an *atomic concept* or simply *atom*, which looks like a tiny chapter. We try to present only one new concept per atom. A typical atom contains one or more small, runnable pieces of code and the output it produces.
2. **No forward references.** As much as possible, we avoid saying, "These features are explained in a later atom."
3. **No references to other programming languages.** We do so only when necessary. An analogy to a feature in a language you don't understand isn't