

# Table of Contents

## Preface

## Part 1 – Kotlin Coroutines on Android

### Chapter 1: Introduction to Asynchronous Programming in Android

#### Technical requirements

#### Understanding asynchronous programming

#### Exploring threads, AsyncTasks, and Executors

#### Threads

#### Callbacks

#### AsyncTask

#### Executors

#### The new way to do it – coroutines and flows

#### Kotlin coroutines

#### Kotlin Flows

#### Summary

#### Further reading

### Chapter 2: Understanding Kotlin Coroutines

## Technical requirements

### Creating coroutines in Android

#### Exploring coroutine builders, scopes, and dispatchers

#### Coroutine builders

#### Coroutine scopes

#### Coroutine dispatchers

#### Understanding coroutine contexts and jobs

#### Coroutine contexts

#### Coroutine jobs

#### Exercise – using coroutines in an Android app

#### Summary

## *Chapter 3: Handling Coroutine Cancellations and Exceptions*

### Technical requirements

#### Canceling coroutines

#### Exercise 3.01 – canceling coroutines in an Android app

#### Managing coroutine timeouts

#### Catching exceptions in coroutines

## Exercise 3.02 – catching exceptions in your coroutines

### Summary

## Chapter 4: Testing Kotlin Coroutines

### Technical requirements

### Setting up an Android project for testing coroutines

### Unit testing suspending functions

### Testing coroutines

## Exercise 4.01 – adding tests to coroutines in an Android app

### Summary

### Further reading

## Part 2 – Kotlin Flows on Android

## Chapter 5: Using Kotlin Flows

### Technical requirements

### Using Flows in Android

### Creating Flows with Flow builders

### Using operators with Flows

### Collecting Flows with terminal operators

### Transforming Flows with Intermediate operators

## Buffering and combining flows

### Buffering Kotlin Flows

### Combining Flows

### Exploring StateFlow and SharedFlow

### Exercise 5.01 – Using Kotlin Flow in an Android app

### Summary

## Chapter 6: Handling Flow Cancelations and Exceptions

### Technical requirements

### Canceling Kotlin Flows

### Retrying tasks with Flow

### Catching exceptions in Flows

### Handling Flow completion

### Exercise 6.01 – Handling Flow exception in an Android app

### Summary

## Chapter 7: Testing Kotlin Flows

### Technical requirements

### Setting up an Android project for testing Flows

### Testing Kotlin Flows

## Testing Flows with Turbine

### Exercise 7.01 – Adding tests to Flows in an Android app

#### Summary

### Other Books You May Enjoy

[Support](#)   [Sign Out](#)

©2022 O'REILLY MEDIA, INC.   [TERMS OF SERVICE](#)   [PRIVACY POLICY](#)