

Auf einen Blick

Teil I

Kotlin 17

Teil II

Programmiertechniken 375

Teil III

App-Entwicklung 481

Teil IV

Backend und Server 661

Inhalt

Vorwort	13
---------------	----

TEIL I Kotlin

1 Hello World!	19
1.1 Über Kotlin	19
1.2 Installation	21
1.3 »Hello World!« mit und ohne IDE ausführen	24
2 Crashkurs	29
2.1 Elementare Syntaxregeln	29
2.2 Konventionen	41
2.3 Von Java zu Kotlin	42
3 Operatoren	47
3.1 Übersicht	47
3.2 Anmerkungen	49
3.3 Priorität von Operatoren	59
4 Variablenverwaltung	61
4.1 Variablen	61
4.2 Unveränderliche Variablen	63
4.3 Konstanten und Enumerationen	64
4.4 Der Umgang mit »null«	65
5 Datentypen	69
5.1 Die wichtigsten Kotlin-Datentypen	69
5.2 Zahlen	72
5.3 Bereiche (Ranges)	74
5.4 Zufallszahlen	76
5.5 Boolesche Ausdrücke	77

6	Zeichenketten	79
6.1	Die Klasse »Char«	80
6.2	Die Klasse »String«	82
6.3	Mit Zeichenketten arbeiten	85
6.4	Beispiel: Passwortqualität testen	94
6.5	Die Klasse »StringBuilder«	97
7	Datum und Uhrzeit	99
7.1	Neue Java-Klassen (»java.time«)	100
7.2	Herkömmliche Java-Klassen (»Date« und »Calendar«)	107
7.3	Die »Duration and Time Measurement API« (kotlin.time)	113
8	Listen, Sets, Maps und Arrays	117
8.1	Listen	118
8.2	Sets	132
8.3	Maps	135
8.4	Sequenzen	136
8.5	Arrays	140
9	Verzweigungen und Schleifen	147
9.1	Die »if«-Verzweigung	147
9.2	Die »when«-Verzweigung	150
9.3	Die »for«-Schleife	152
9.4	Die »while«-Schleife	154
9.5	Die »repeat«-Schleife	155
9.6	»break« und »continue«	156
9.7	Beispiele: Summen, Produkte, Duplikate und Co.	157
10	Funktionen	163
10.1	Funktionen definieren und aufrufen	164
10.2	Parameter	168
10.3	Rekursion	172
10.4	Lokale Funktionen	175
10.5	Die »main«-Funktion	176
10.6	Beispiel: Pfadsuche	177

11	Lambda-Ausdrücke und funktionale Programmierung	189
11.1	Hello Lambda!	190
11.2	Lambda-Ausdrücke und Funktionen für Fortgeschrittene	192
11.3	Beispiel: Funktionen erzeugen und verarbeiten	197
11.4	»map«, »filter«, »reduce« und Co.	199
11.5	Beispiel: Textanalyse mit Lambda-Ausdrücken	211
11.6	Beispiel: Dosenpyramide in drei Farben	213
11.7	Objekte verarbeiten (»apply«, »let«, »with« etc.)	218
11.8	Inline-Funktionen	223
12	Klassen und Objekte	227
12.1	Klassen	227
12.2	Der Konstruktor	233
12.3	Eigenschaften	240
12.4	Eigenschaften später initialisieren	245
12.5	Zugriffsteuerung	249
12.6	Beispiel: Bankkonto	251
12.7	Objekte ohne Klassen	253
12.8	Beispiel: Quiz	261
12.9	Enumerationen	266
13	Vererbung, Schnittstellen, Extensions	271
13.1	Vererbung	272
13.2	Vererbung (Konstruktoren, abstrakte Klassen, Polymorphie)	278
13.3	Die Klasse »Any«	284
13.4	Datenklassen	287
13.5	Beispiel: Schachfiguren	289
13.6	Schnittstellen	296
13.7	Beispiel: Geometrische Objekte mit Schnittstellen verarbeiten	303
13.8	Extensions	304
13.9	Receiver-Funktionen	309
13.10	Infix-Funktionen	312
13.11	Operator Overloading	313
13.12	Beispiel: Rechnen mit komplexen Zahlen	316
13.13	Delegation	317

14	Reflection, Generics und Annotationen	327
14.1	Reflection	328
14.2	Generics	330
14.3	Generische Typen: Zusatzregeln und Sonderfälle	333
14.4	Annotationen	344
14.5	Type-safe Builder	348
15	Exceptions	357
15.1	Fehlerabsicherung	358
15.2	Selbst Fehler auslösen (»throw«)	365
15.3	Arbeitstechniken	367
16	Pakete, Importe und Module	369
16.1	Pakete	369
16.2	Importe	370
16.3	Module	373

TEIL II Programmieretechniken

17	Asynchrone Programmierung	377
17.1	Hello Coroutines!	378
17.2	Koroutinen ausführen	382
17.3	Koroutinen abbrechen	390
17.4	Exceptions in asynchronem Code	395
17.5	Den Zugriff auf gemeinsame Daten synchronisieren	398
17.6	Suspending Functions	401
17.7	Asynchroner Code in Android-Apps	404
17.8	Beispiel: Effizient numerisch integrieren	407
18	Dateien verarbeiten (I/O, JSON, XML)	411
18.1	Umgang mit Dateien und Verzeichnissen	412
18.2	Textdateien lesen und schreiben	419
18.3	Download von Dateien	422

18.4	JSON und Serialisierung	422
18.5	JSON-Beispiel: Die »New-York-Times«-Bestseller	430
18.6	XML	432
19	Datenbankzugriff (Exposed)	437
19.1	Hello Exposed!	438
19.2	Verbindungsaufbau und Transaktionen	442
19.3	Data Access Objects (DAO)	448
19.4	DAO-Beispiel: Bücherdatenbank	455
19.5	SQL als Domain-specific Language (DSL)	462
20	JavaFX	473
20.1	Hello JavaFX!	473
20.2	TornadoFX	477

TEIL III App-Entwicklung

21	Hello Android!	483
21.1	Android Studio installieren	484
21.2	Das erste Projekt in Android Studio	486
21.3	Emulator einrichten	488
21.4	Smartphone einrichten	492
21.5	Aufbau eines Android-Studio-Projekts	494
21.6	Eigener Button, eigener Code	498
21.7	Personalisierung der App	505
22	App: Fahrenheit-Umrechner	509
22.1	Layout	510
22.2	Der Code zur Temperaturumrechnung	512
22.3	Debugging	518
22.4	Screenshots	522

23	Android Basics	525
23.1	Android-Grundlagen	525
23.2	Steuerelemente	527
23.3	Texte anzeigen und eingeben (»TextView«, »EditText«)	529
23.4	Buttons	533
23.5	ImageView	535
23.6	Layoutregeln	541
23.7	Listen und Tabellen (»RecyclerView«)	548
23.8	Preferences	561
23.9	Permissions	564
24	Aktivitäten, Fragmente und Menüs	573
24.1	Aktivitäten und Fragmente	574
24.2	Beispiel: App mit drei leeren Fragmenten	576
24.3	Datenaustausch zwischen Fragmenten	586
24.4	Beispiel: Datenaustausch zwischen drei Fragmenten	593
24.5	Einfache Menüs (Overflow Menu)	597
25	App: Währungsumrechner	603
25.1	Die Klasse »CurrencyCalculator«	604
25.2	Hauptaktivität	610
25.3	Fragment zur Währungsumrechnung (»MainFragment«)	611
25.4	Einstellungsfragment (»SetupFragment«)	616
25.5	Fragment mit App-Informationen (»AboutFragment«)	620
26	Jetpack Compose	623
26.1	Hello Compose!	625
26.2	Steuerelemente	635
26.3	Container	641
26.4	Listen	645
26.5	Theming	648
26.6	Aktivitäten und Fragmente	651
26.7	Beispiel: Fahrenheit-Umrechner	656

TEIL IV Backend und Server

27	Hello Server!	663
27.1	Hello Ktor!	665
27.2	Beispiel: URL-Verkürzer	670
27.3	Beispiel: URL-Verkürzer mit Datenbank-Server	675
28	Ktor-Programmiertechniken	681
28.1	Projekt- und Programmaufbau	682
28.2	Routing	684
28.3	Request und Response	690
28.4	HTML- und CSS-Dokumente zusammensetzen	694
28.5	REST-APIs	701
28.6	Authentifizierung	709
28.7	Ktor-Debugging	715
29	App: Evaluierungssystem (Backend)	719
29.1	Projektaufbau	720
29.2	Datenbankdesign	723
29.3	Der Datenbank-Code	725
29.4	Weboberfläche	735
29.5	Die REST-API	745
30	App: Evaluierungssystem (Client)	753
30.1	Den Ktor-Client in Android-Apps verwenden	755
30.2	Ktor-Client-Programmiertechniken	758
30.3	Projektaufbau der Evaluierungs-App	761
30.4	Liste der Evaluierungen anzeigen	766
30.5	Einmal-Login	772
30.6	Evaluierung durchführen	775
30.7	Evaluierungsergebnisse anzeigen	781

Inhalt

A	IntelliJ, Android Studio und Gradle	783
A.1	IntelliJ und Android Studio	783
A.2	Gradle	795
	Index	807