edv-buchversand.de

Das Swift-Handbuch

Apps programmieren für alle Apple-Plattformen

» Hier geht's direkt zum Buch

DAS INHALTS-VERZEICHNIS

Dieses Inhaltsverzeichnis wird Ihnen von www.edv-buchversand.de zur Verfügung gestellt.

Inhalt

Vorwort ..... **XXIII**

Teil I: Swift **1**

1 Die Programmiersprache Swift **3**

1.1 Die Geschichte von Swift 3

1.2 Die Bedeutung von Swift im Apple-Kosmos 5

1.3 Das UI-Framework: SwiftUI 5

1.4 Was Sie als App-Entwickler brauchen 6

1.5 Programmieren für Beginner (und darüber hinaus): Playgrounds 8

1.6 Weitere wichtige Ressourcen 10

1.6.1 Apple-Developer-App 10

1.6.2 Apples Developer-Website 11

1.6.3 Swift.org 12

1.6.4 In eigener Sache 13

2 Grundlagen der Programmierung **14**

2.1 Grundlegendes 14

2.1.1 Swift Standard Library 14

2.1.2 print 16

2.1.3 Befehle und Semikolons 17

2.1.4 Operatoren 18

2.2 Variablen und Konstanten ... 19
2.2.1 Erstellen von Variablen und Konstanten ... 19
2.2.2 Variablen und Konstanten in der Konsole ausgeben ... 21
2.2.3 Type Annotation und Type Inference ... 21
2.2.4 Gleichzeitiges Erstellen und Deklarieren mehrerer Variablen und Konstanten ... 23
2.2.5 Namensrichtlinien ... 24
2.3 Kommentare ... 24

3 Schleifen und Abfragen ... 26
3.1 Schleifen ... 26
3.1.1 For-In ... 26
3.1.2 While ... 28
3.1.3 Repeat-While ... 29
3.2 Abfragen ... 30
3.2.1 If ... 30
3.2.2 Switch ... 35
3.2.3 Guard ... 39
3.3 Control Transfer Statements ... 40
3.3.1 Anstoßen eines neuen Schleifendurchlaufs mit continue ... 41
3.3.2 Verlassen der kompletten Schleife mit break ... 41
3.3.3 Labeled Statements ... 42

4 Typen in Swift ... 45
4.1 Integer ... 47
4.2 Fließkommazahlen ... 48
4.3 Bool ... 49
4.4 String ... 49
4.4.1 Erstellen eines Strings ... 49
4.4.2 Zusammenfügen von Strings ... 50
4.4.3 Character auslesen ... 51
4.4.4 Character mittels Index auslesen ... 52
4.4.5 Character entfernen und hinzufügen ... 54
4.4.6 Anzahl der Character zählen ... 55
4.4.7 Präfix und Suffix prüfen ... 55
4.4.8 String Interpolation ... 56

4.5 Array ... 56
4.5.1 Erstellen eines Arrays ... 57
4.5.2 Zusammenfügen von Arrays ... 58
4.5.3 Inhalte eines Arrays leeren ... 59
4.5.4 Prüfen, ob ein Array leer ist ... 59
4.5.5 Anzahl der Elemente eines Arrays zählen ... 60
4.5.6 Zugriff auf die Elemente eines Arrays ... 60
4.5.7 Neue Elemente zu einem Array hinzufügen ... 61
4.5.8 Bestehende Elemente aus einem Array entfernen ... 62
4.5.9 Bestehende Elemente eines Arrays ersetzen ... 63
4.5.10 Alle Elemente eines Arrays auslesen und durchlaufen ... 63
4.6 Set ... 65
4.6.1 Erstellen eines Sets ... 65
4.6.2 Inhalte eines bestehenden Sets leeren ... 66
4.6.3 Prüfen, ob ein Set leer ist ... 66
4.6.4 Anzahl der Elemente eines Sets zählen ... 67
4.6.5 Element zu einem Set hinzufügen ... 67
4.6.6 Element aus einem Set entfernen ... 67
4.6.7 Prüfen, ob ein bestimmtes Element in einem Set vorhanden ist ... 68
4.6.8 Alle Elemente eines Sets auslesen und durchlaufen ... 68
4.6.9 Sets miteinander vergleichen ... 69
4.6.10 Neue Sets aus bestehenden Sets erstellen ... 71
4.7 Dictionary ... 73
4.7.1 Erstellen eines Dictionaries ... 73
4.7.2 Prüfen, ob ein Dictionary leer ist ... 75
4.7.3 Anzahl der Schlüssel-Wert-Paare eines Dictionaries zählen ... 75
4.7.4 Wert zu einem Schlüssel eines Dictionaries auslesen ... 75
4.7.5 Neues Schlüssel-Wert-Paar zu Dictionary hinzufügen ... 76
4.7.6 Bestehendes Schlüssel-Wert-Paar aus Dictionary entfernen ... 77
4.7.7 Bestehendes Schlüssel-Wert-Paar aus Dictionary verändern ... 77
4.7.8 Alle Schlüssel-Wert-Paare eines Dictionaries auslesen und durchlaufen ... 78
4.8 Tuple ... 79
4.8.1 Zugriff auf die einzelnen Elemente eines Tuples ... 80
4.8.2 Tuple und Switch ... 81

4.9 Optional . . . 84
4.9.1 Deklaration eines Optionals . . . 85
4.9.2 Zugriff auf den Wert eines Optionals . . . 86
4.9.3 Optional Binding . . . 88
4.9.4 Implicitly Unwrapped Optional . . . 90
4.9.5 Optional Chaining . . . 91
4.9.6 Optional Chaining über mehrere Eigenschaften und Funktionen 96
4.10 Any und AnyObject . . . 100
4.11 Type Alias . . . 100
4.12 Value Type versus Reference Type . . . 101
4.12.1 Reference Types auf Gleichheit prüfen . . . 103

5 Funktionen . . . 105
5.1 Funktionen mit Parametern . . . 106
5.1.1 Argument Labels und Parameter Names . . . 108
5.1.2 Default Value für Parameter . . . 110
5.1.3 Variadic Parameter . . . 111
5.1.4 In-Out-Parameter . . . 112
5.2 Funktionen mit Rückgabewert . . . 114
5.3 Function Types . . . 116
5.3.1 Funktionen als Variablen und Konstanten . . . 117
5.4 Verschachtelte Funktionen . . . 118

6 Enumerations, Structures und Classes . . . 120
6.1 Enumerations . . . 120
6.1.1 Enumerations und switch . . . 123
6.1.2 Associated Values . . . 124
6.1.3 Raw Values . . . 126
6.2 Structures . . . 128
6.2.1 Erstellen von Structures und Instanzen . . . 129
6.2.2 Eigenschaften und Funktionen . . . 130
6.3 Classes . . . 136
6.3.1 Erstellen von Klassen und Instanzen . . . 136
6.3.2 Eigenschaften und Funktionen . . . 137

6.4 Enumeration vs. Structure vs. Class . . . 139
6.4.1 Gemeinsamkeiten und Unterschiede . . . 139
6.4.2 Wann nimmt man was? . . . 140
6.5 self . . . 142

7 Eigenschaften und Funktionen von Typen . . . 144
7.1 Properties . . . 144
7.1.1 Stored Property . . . 145
7.1.2 Lazy Stored Property . . . 148
7.1.3 Computed Property . . . 150
7.1.4 Read-Only Computed Property . . . 154
7.1.5 Property Observer . . . 155
7.1.6 Property Wrapper . . . 159
7.1.7 Type Property . . . 169
7.2 Globale und lokale Variablen . . . 172
7.3 Methoden . . . 175
7.3.1 Instance Methods . . . 175
7.3.2 Type Methods . . . 178
7.4 Subscripts . . . 179
7.5 Self . . . 184

8 Initialisierung . . . 185
8.1 Aufgabe der Initialisierung . . . 186
8.2 Erstellen eigener Initializer . . . 187
8.3 Initializer Delegation . . . 193
8.3.1 Initializer Delegation bei Value Types . . . 193
8.3.2 Initializer Delegation bei Reference Types . . . 194
8.4 Failable Initializer . . . 197
8.5 Required Initializer . . . 200
8.6 Deinitialisierung . . . 201

9 Closures . . . 203
9.1 Closures als Parameter von Funktionen . . . 204
9.2 Trailing Closures . . . 208
9.3 Escaping Closures . . . 209
9.4 Autoclosures . . . 210

10 Vererbung ... 212
10.1 Überschreiben von Eigenschaften und Funktionen einer Klasse ... 215
10.2 Überschreiben von Eigenschaften und Funktionen einer Klasse verhindern ... 218
10.3 Zugriff auf die Superklasse ... 219
10.4 Initialisierung und Vererbung ... 220
10.4.1 Zwei-Phasen-Initialisierung ... 221
10.4.2 Überschreiben von Initializern ... 227
10.4.3 Vererbung von Initializern ... 230
10.4.4 Required Initializer ... 230

11 Speicherverwaltung mit ARC ... 232
11.1 Strong Reference Cycles ... 235
11.1.1 Weak References ... 238
11.1.2 Unowned References ... 241
11.1.3 Weak Reference vs. Unowned Reference ... 244

12 Weiterführende Sprachmerkmale von Swift ... 245
12.1 Nested Types ... 245
12.2 Extensions ... 247
12.2.1 Computed Properties ... 248
12.2.2 Methoden ... 248
12.2.3 Initializer ... 249
12.2.4 Subscripts ... 251
12.2.5 Nested Types ... 252
12.3 Protokolle ... 252
12.3.1 Deklaration von Eigenschaften und Funktionen ... 254
12.3.2 Der Typ eines Protokolls ... 264
12.3.3 Protokolle und Extensions ... 266
12.3.4 Vererbung in Protokollen ... 271
12.3.5 Class-Only-Protokolle ... 272
12.3.6 Optionale Eigenschaften und Funktionen ... 273
12.3.7 Protocol Composition ... 276
12.3.8 Delegation ... 277
12.3.9 Übersicht diverser vorhandener Protokolle ... 280
12.4 Key-Path ... 282

13 Type Checking und Type Casting ... 286
13.1 Type Checking mit „is“ ... 289
13.2 Type Casting mit „as“ ... 290

14 Error Handling ... 293
14.1 Deklaration und Feuern eines Fehlers ... 294
14.2 Reaktion auf einen Fehler ... 297
14.2.1 Mögliche Fehler mittels Do-Catch auswerten ... 298
14.2.2 Mögliche Fehler in Optionals umwandeln ... 301
14.2.3 Mögliche Fehler weitergeben ... 302
14.2.4 Mögliche Fehler ignorieren ... 303

15 Generics ... 305
15.1 Generic Functions ... 306
15.2 Generic Types ... 309
15.3 Type Constraints ... 312
15.4 Associated Types ... 313

16 Nebenläufigkeit ... 317
16.1 Asynchronen Code schreiben und aufrufen ... 317
16.2 Mehrere asynchrone Funktionen parallel ausführen ... 320
16.3 Priorität von Tasks festlegen ... 322
16.4 Ergebnisse über Task zurückliefern und auswerten ... 323
16.5 Tasks abbrechen ... 326
16.6 Ausführung von Tasks verzögern ... 327
16.7 Parent- und Child-Tasks ... 328
16.8 Actors ... 330
16.9 Main Actor ... 332
16.10 Sendable ... 334

17 Dateien und Interfaces ... 336
17.1 Modules und Source Files ... 336
17.2 Access Control ... 337
17.2.1 Access Level ... 337
17.2.2 Explizite und implizite Zuweisung eines Access Levels ... 341
17.2.3 Besonderheiten ... 343

Teil II: Xcode **349**

18 Grundlagen, Aufbau und Einstellungen von Xcode **351**

18.1 Über Xcode 352

18.2 Arbeiten mit Xcode 353

18.2.1 Dateien und Formate eines Xcode-Projekts 353

18.2.2 Umgang mit Dateien und Ordnern 359

18.3 Der Aufbau von Xcode 362

18.3.1 Toolbar 362

18.3.2 Navigator 364

18.3.3 Editor 368

18.3.4 Inspectors 373

18.3.5 Debug Area 376

18.4 Einstellungen 377

18.4.1 General 377

18.4.2 Apple Accounts 378

18.4.3 Intelligence 378

18.4.4 Behaviors 379

18.4.5 Navigation 379

18.4.6 Notifications 380

18.4.7 Themes 381

18.4.8 Editing 381

18.4.9 Shortcuts 382

18.4.10 Source Control 383

18.4.11 Components 383

18.4.12 Locations 384

18.5 Projekteinstellungen 385

18.5.1 Einstellungen am Projekt 386

18.5.2 Einstellungen am Target 389

18.5.3 Einstellungen am Scheme 396

19 Dokumentation, Devices und Organizer ... 401
19.1 Dokumentation ... 401
19.1.1 Aufbau und Funktionsweise ... 402
19.1.2 Direktzugriff im Editor ... 406
19.2 Devices und Simulatoren ... 407
19.2.1 Simulatoren ... 409
19.2.2 Devices ... 411
19.3 Organizer ... 412

20 Debugging und Refactoring ... 414
20.1 Debugging ... 414
20.1.1 Einfache Konsolenausgaben ... 416
20.1.2 Logging ... 416
20.1.3 Arbeiten mit Breakpoints ... 421
20.1.4 Debug Navigator ... 427
20.2 Refactoring ... 430
20.3 Instruments ... 432

21 Tipps und Tricks für das effiziente Arbeiten mit Xcode ... 435
21.1 Code Snippets ... 435
21.2 Open Quickly ... 438
21.3 Related Items ... 438
21.4 Navigation über die Jump Bar ... 439
21.5 MARK, TODO und FIXME ... 440
21.6 Runtime-Bedingungen ... 441
21.7 Shortcuts für den Navigator ... 443
21.8 Clean Build ... 443
21.9 Coding Assistant ... 443
21.10 Playgrounds ... 445
21.10.1 Was sind Playgrounds? ... 445
21.10.2 Code schreiben und testen ... 448
21.10.3 Dateien hinzufügen ... 450
21.10.4 Kommentare und Dokumentation ... 452
21.10.5 Swift Playgrounds-App ... 455

Teil III: App-Entwicklung **459**

22 Grundlagen der App-Entwicklung **461**
22.1 Die Basis: SwiftUI 461
22.2 Bestandteile einer App 463
22.2.1 Umsetzung der Daten 464
22.2.2 Umsetzung der Ansichten 464
22.2.3 Weitere Frameworks 464
22.3 Das Design: Liquid Glass 465
22.4 Die Syntax von SwiftUI 466
22.5 Aufbau einer App 467
22.6 Das View-Protokoll 468
22.7 Aktualisierung von Views mittels Status 469
22.8 Grundlagen des Status 471
22.9 Anpassung von Views mittels Modifier 474
22.10 Gruppierung von Views mittels Containern 478
22.11 Praxis: Unsere erste App 479
22.11.1 Bestandteile des neuen Projekts 484
22.11.2 Änderung des Textes 487
22.11.3 Einsatz der Preview 488

23 Views in SwiftUI **491**
23.1 Textdarstellung und -bearbeitung 491
23.1.1 Text 491
23.1.2 TextField 502
23.1.3 SecureField 507
23.1.4 TextEditor 508
23.2 Bilder 511
23.2.1 Image-Instanz erstellen 513
23.2.2 Größe einer Image-Instanz ändern 516
23.3 Schaltflächen 519
23.3.1 Button 519
23.3.2 EditButton 524
23.3.3 PasteButton 525

23.4 Werteauswahl ... 527
23.4.1 Toggle ... 527
23.4.2 Slider ... 532
23.4.3 Stepper ... 539
23.4.4 Picker ... 544
23.4.5 DatePicker ... 548
23.4.6 MultiDatePicker ... 553
23.4.7 ColorPicker ... 555
23.5 Weitere Views ... 558
23.5.1 Label ... 558
23.5.2 ProgressView ... 561
23.5.3 Gauge ... 565

24 View-Layout ... 570
24.1 Stacks ... 570
24.1.1 HStack ... 570
24.1.2 VStack ... 574
24.1.3 ZStack ... 578
24.1.4 Stacks verschachteln ... 580
24.1.5 Lazy Stacks ... 581
24.2 Listen ... 584
24.2.1 List ... 584
24.2.2 ForEach ... 607
24.3 Grids ... 619
24.3.1 Grid ... 620
24.3.2 LazyHGrid und LazyVGrid ... 622
24.4 Table ... 635
24.4.1 Zellen selektieren ... 637
24.4.2 Sortierung ändern ... 638
24.5 Container-Views ... 639
24.5.1 Form ... 640
24.5.2 Section ... 642
24.5.3 Group ... 646
24.5.4 GroupBox ... 649
24.5.5 ViewThatFits ... 651

24.6 Weitere Views ... 654
24.6.1 ScrollView ... 654
24.6.2 OutlineGroup ... 659
24.6.3 DisclosureGroup ... 664
24.6.4 Spacer ... 669
24.6.5 Divider ... 673

25 Navigation ... 675
25.1 NavigationStack und NavigationLink ... 675
25.1.1 Titel in Navigation-Bar setzen ... 680
25.1.2 Eigene View zur Darstellung eines NavigationLink nutzen ... 685
25.1.3 Anzeige einer Ziel-View auf Basis von Daten ... 686
25.1.4 Programmatische Steuerung des Navigation-Stacks ... 690
25.2 NavigationSplitView ... 694
25.2.1 Verknüpfung von NavigationSplitView und List ... 695
25.2.2 Sichtbarkeit der Spalten steuern ... 697
25.2.3 Breite der Spalten anpassen ... 699
25.2.4 Verhalten von NavigationSplitView unter den verschiedenen Apple-Plattformen ... 700
25.3 TabView ... 705
25.4 HSplitView und VSplitView ... 712
25.5 Funktionen zur Präsentation von Views ... 713
25.5.1 Sheet einblenden ... 713
25.5.2 View über gesamtes Fenster legen ... 723
25.5.3 Popover einblenden ... 729

26 Weitere View-Konfigurationen ... 736
26.1 Toolbar ... 736
26.2 Alerts ... 746
26.3 Confirmation Dialog ... 749
26.4 Farben ... 751
26.5 View-Events ... 753
26.6 Eigene Modifier umsetzen und aufrufen ... 754

27 Status ... **757**
27.1 Property ... 759
27.2 State ... 762
27.3 Binding ... 763
27.4 ObservedObject ... 772
27.4.1 Datenmodell vorbereiten ... 773
27.4.2 Datenmodell in View einbinden ... 774
27.4.3 Auf Änderungen reagieren ... 777
27.5 StateObject ... 781
27.6 EnvironmentObject ... 783
27.7 @Observable-Makro und Bindable ... 789
27.8 Environment ... 791
27.9 SceneStorage ... 798
27.10 AppStorage ... 801
27.11 Source of Truth vs. Derived Value ... 802
27.12 Best Practices ... 806

28 Datenhaltung ... **812**
28.1 UserDefaults ... 812
28.1.1 UserDefaults und SwiftUI ... 814
28.2 SwiftData ... 815
28.2.1 Grundlegende Funktionsweise von SwiftData ... 816
28.2.2 Erstellen des eigenen Datenmodells ... 816
28.2.3 Model-Container erzeugen ... 818
28.2.4 Elemente im ModelContext verwalten ... 819
28.2.5 Zusammenspiel mit der Preview ... 824
28.3 Core Data ... 826
28.3.1 Grundlegende Funktionsweise von Core Data ... 827
28.3.2 Grundlegende Elemente beim Einsatz von Core Data ... 827
28.3.3 Einen Core Data Stack erstellen ... 830
28.3.4 Ein Managed Object Model erstellen ... 832
28.3.5 Grundlegende Core-Data-Operationen ... 843
28.3.6 Core Data mit SwiftUI ... 845

29 Weitere Projektkonfigurationen ... 853
29.1 Cross-Platform-Entwicklung ... 853
29.1.1 Neue Targets hinzufügen ... 854
29.1.2 Target-Zuweisung ... 856
29.1.3 Plattform im Code prüfen ... 858
29.1.4 Funktionen auf Verfügbarkeit prüfen ... 859
29.2 Mehrsprachigkeit ... 861
29.2.1 Grundlagen ... 861
29.2.2 Verschiedene Sprachen einer App testen ... 869
29.3 Asset Catalogs ... 870

30 Preview und Library ... 875
30.1 Preview ... 875
30.1.1 Funktionsweise der Preview ... 877
30.1.2 Konfiguration der Preview ... 877
30.1.3 Preview ausführen und anhalten ... 879
30.2 Library ... 881

Teil IV: Source Control und Testing ... 883

31 Source Control ... 885
31.1 Basisfunktionen und -begriffe der Source Control ... 885
31.2 Source Control in Xcode ... 887
31.2.1 Bestehendes Projekt klonen ... 889
31.2.2 Lokale Änderungen committen ... 890
31.2.3 Lokale Änderungen verwerfen ... 893
31.2.4 Pull und Push ... 893
31.2.5 Aktuelle Branches vom Repository laden ... 894
31.2.6 Git-Repository mit neuem Xcode-Projekt erzeugen ... 894
31.2.7 Optische Source-Control-Hervorhebungen im Editor ... 895
31.2.8 Zugriff auf GitHub, GitLab und Bitbucket ... 897
31.3 Source Control Navigator ... 898
31.4 Code Review Mode ... 900

32 Testing ... **903**
32.1 Unit-Tests ... 903
32.1.1 Unit-Tests mit Swift Testing ... 908
32.1.2 Unit-Tests mit XCTest ... 918
32.1.3 Neue Test-Datei erstellen ... 922
32.1.4 Ausführen von Unit-Tests ... 924
32.2 Performancetests ... 926
32.3 UI-Tests ... 928
32.3.1 Klassen für UI-Tests ... 930
32.3.2 Aufbau von UI-Test-Klassen ... 932
32.3.3 Automatisches Erstellen von UI-Tests ... 932
32.3.4 Einsatz von UI-Tests ... 933

Teil V: Veröffentlichung von Apps ... **935**

33 Veröffentlichung im App Store ... **937**
33.1 Das Apple Developer Portal ... 939
33.1.1 Zertifikate, App IDs und Provisioning Profiles ... 942
33.1.2 Code Signing ... 960
33.2 App Store Connect ... 965
33.2.1 Apps für den App Store vorbereiten und verwalten ... 967
33.2.2 Apps erstellen, hochladen und einreichen ... 970
33.3 App Store Review Guidelines ... 973

34 Das Business Model für Ihre App ... **975**
34.1 Geschäftsmodelle ... 975
34.1.1 Free Model ... 976
34.1.2 Freemium Model ... 976
34.1.3 Subscription Model ... 976
34.1.4 Paid Model ... 977
34.1.5 Paymium Model ... 978
34.2 App Bundles ... 978
34.3 Veröffentlichung außerhalb des App Store ... 980
34.3.1 Das Apple Developer Enterprise Program ... 981

35 TestFlight 983
35.1 TestFlight in App Store Connect 983
35.2 TestFlight im App Store 985

Stichwortverzeichnis 988