

Android 9, Kotlin i Android Studio 3.2

U JEDNOJ KNJIZI

Neil Smyth



Android 9, Kotlin i Android Studio 3.2

NEIL SMYTH



Izdavač:



**kompjuter
biblioteka**

Obalskih radnika 4a, Beograd

Tel: 011/2520272

e-mail: kombib@gmail.com

internet: www.kombib.rs

Urednik: Mihailo J. Šolajić

Za izdavača, direktor:

Mihailo J. Šolajić

Autor: Neil Smyth

Prevod: Slavica Prudkov

Lektura: Miloš Jevtović

Slog : Zvonko Aleksić

Znak Kompjuter biblioteke:

Miloš Milosavljević

Štampa: „Pekograf“, Zemun

Tiraž: 500

Godina izdanja: 2018.

Broj knjige: 511

Izdanje: Prvo

ISBN: 978-86-7310-534-5

Kotlin / Android Studio 3.2 Development Essentials – Android 9 Edition

by Neil Smyth

2018

Copyright © 2018 by Neil Smyth, Payload Media, Inc.

All right reserved. No part of this book may be reproduced or transmitted in any form or by means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Autorizovani prevod sa engleskog jezika edicije u izdanju „Payload Media, Inc.“, Copyright © 2018.

Sva prava zadržana. Nije dozvoljeno da nijedan deo ove knjige bude reprodukovan ili snimljen na bilo koji način ili bilo kojim sredstvom, elektronskim ili mehaničkim, uključujući fotokopiranje, snimanje ili drugi sistem presnimavanja informacija, bez dozvole izdavača.

Zaštitni znaci

Kompjuter Biblioteka i „Payload Media“ su pokušali da u ovoj knjizi razgraniče sve zaštitne oznake od opisnih termina, prateći stil isticanja oznaka velikim slovima.

Autor i izdavač su učinili velike napore u pripremi ove knjige, čiji je sadržaj zasnovan na poslednjem (dostupnom) izdanju softvera. Delovi rukopisa su možda zasnovani na predizdanju softvera dobijenog od strane proizvođača. Autor i izdavač ne daju nikakve garancije u pogledu kompletnosti ili tačnosti navoda iz ove knjige, niti prihvataju ikakvu odgovornost za performanse ili gubitke, odnosno oštećenja nastala kao direktna ili indirektna posledica korišćenja informacija iz ove knjige.

CIP - Каталогизација у публикацији
Народна библиотека Србије, Београд



KRATAK SADRŽAJ

POGLAVLJE 1

Uvod 1

POGLAVLJE 2

Podešavanje Android Studio razvojnog okruženja 3

POGLAVLJE 3

Kreiranje primera Android aplikacije u Android Studiou 13

POGLAVLJE 4

Pregled korisničkog interfejsa razvojnog okruženja Android Studio 29

POGLAVLJE 5

Kreiranje Android virtuelnog uređaja (AVD) u razvojnog okruženju Android Studio . 37

POGLAVLJE 6

Upotreba i konfigurisanje Android Studio AVD emulatora 47

POGLAVLJE 7

Testiranje Android Studio aplikacija na fizičkom Android uređaju 57

POGLAVLJE 8

Osnove editora koda razvojnog okruženja Android Studio 63

POGLAVLJE 9

Pregled Android arhitekture 75

POGLAVLJE 10

Anatomija Android aplikacije 79

POGLAVLJE 11

Uvod u Kotlin 83

POGLAVLJE 12

Kotlin tipovi podataka, promenljive i vrednost null 87

POGLAVLJE 13

Kotlin operatori i izrazi. 97

POGLAVLJE 14

Kotlin kontrola toka 105

POGLAVLJE 15

Pregled Kotlin funkcija i lambda izraza 113

POGLAVLJE 16

Osnove objektno-orijentisanog programiranja u Kotlinu 119

POGLAVLJE 17

Uvod u Kotlin nasleđivanje i izradu potklasa 129

POGLAVLJE 18

Razumevanje „životnih ciklusa“ Android aplikacije i aktivnosti 135

POGLAVLJE 19

Obrada promena stanja Android aktivnosti 141

POGLAVLJE 20

Promene stanja Android aktivnosti kroz primer. 149

POGLAVLJE 21

Snimanje i vraćanje stanja Android aktivnosti 157

POGLAVLJE 22

Layout Editor alatka razvojnog okruženja Android Studio 163

POGLAVLJE 23

Vodič za upotrebu rasporeda ConstraintLayout
u razvojnem okruženju Android Studio 177

POGLAVLJE 24

Upotreba lanaca i razmera klase ConstraintLayout
u razvojnem okruženju Android Studio 195

POGLAVLJE 25

Uputstvo za klasu ConstraintLayout za Layout Editor alatku
razvojnog okruženja Android Studio 203

POGLAVLJE 26

Ručno dizajniranje XML rasporeda u razvojnem okruženju Android Studio 213

POGLAVLJE 27

Upravljanje ograničenjima upotrebom skupova ograničenja. 219

POGLAVLJE 28

Uputstvo za Android klasu ConstraintSet225

POGLAVLJE 29

Vodič za upotrebu funkcije Instant Run u razvojnom okruženju Android Studio233

POGLAVLJE 30

Pregled i primer Android funkcije za obradu događaja239

POGLAVLJE 31

Obrada događaja dodira i višestrukog dodira na Androidu247

POGLAVLJE 32

Detektovanje uobičajenih pokreta prstom upotrebom Android
Gesture Detector klase255

POGLAVLJE 33

Implementiranje prilagođenih pokreta prstom i prepoznavanje
primicanja i odmicanja prstiju na Androidu261

POGLAVLJE 34

Uvod u Android fragmente273

POGLAVLJE 35

Upotreba fragmenata u razvojnom okruženju Android Studio – primer281

POGLAVLJE 36

Moderna arhitektura Android aplikacije sa Jetpackom293

POGLAVLJE 37

Uputstvo za Android Jetpack komponentu ViewModel299

POGLAVLJE 38

Uputstvo za Android Jetpack LiveData komponentu307

POGLAVLJE 39

Pregled Android JetpackData Binding biblioteke311

POGLAVLJE 40

Uputstvo za Android Jetpack povezivanje podataka319

POGLAVLJE 41

Upotreba Android komponenata koje su svesne „životnog ciklusa“329

POGLAVLJE 42

Uputstvo za Android Jetpack svesnost „životnog ciklusa“.333

POGLAVLJE 43

Pregled Android Navigation Architecture Componenta339

POGLAVLJE 44

Uputstvo za Android Jetpack komponentu navigacije 347

POGLAVLJE 45

Kreiranje Overflow menija na Androidu i upravljanje njima 361

POGLAVLJE 46

Animacija korisničkog interfejsa pomoću Android Transitions radnog okvira 371

POGLAVLJE 47

Uputstvo za Android prelaze upotrebom metoda `beginDelayedTransition` 381

POGLAVLJE 48

Implementiranje Android prelaza scene – uputstvo 387

POGLAVLJE 49

Upotreba plutajućeg komandnog dugmeta i Snackbara 395

POGLAVLJE 50

Kreiranje interfejsa sa karticama upotrebom komponente `TabLayout` 405

POGLAVLJE 51

Upotreba vidžeta `RecyclerView` i `CardView` 417

POGLAVLJE 52

Uputstvo za Android komponente `RecyclerView` i `CardView` 423

POGLAVLJE 53

Uputstvo za primer podataka `Layout Editor` alatke 433

POGLAVLJE 54

Upotreba rasporeda `AppBar` i `Collapsing Toolbar` 443

POGLAVLJE 55

Implementiranje Android navigacione fioke 453

POGLAVLJE 56

Uputstvo za Android Studio šablon `Master/DetailFlow` 461

POGLAVLJE 57

Pregled Android namere 471

POGLAVLJE 58

Android eksplicitne namere – primenjeni primer 477

POGLAVLJE 59

Android implicitne namere – primenjeni primer 485

POGLAVLJE 60

Android objekti `Broadcast Intents` i `Broadcast Receivers` 493

POGLAVLJE 61

Pregled osnova programske niti i klase `AsyncTasks` 501

POGLAVLJE 62

Pregled Android pokrenutih i vezanih servisa 509

POGLAVLJE 63

Implementiranje Android pokrenutog servisa – primenjeni primer 515

POGLAVLJE 64

Android lokalni vezani servis – primenjeni primer 523

POGLAVLJE 65

Android udaljeni vezani servisi – primenjeni primer 531

POGLAVLJE 66

Uputstvo za Android obaveštenja 537

POGLAVLJE 67

Uputstvo za Android Direct Reply obaveštenja 551

POGLAVLJE 68

Uvod u Android podršku za više prozora 561

POGLAVLJE 69

Uputstvo za Android Studio režime sa više prozora Split-Screen i Freeform 569

POGLAVLJE 70

Pregled Android SQLite baza podataka 575

POGLAVLJE 71

Android Room Persistence biblioteka. 581

POGLAVLJE 72

Uputstvo za Android `TableLayout` i `TableRow` 591

POGLAVLJE 73

Uputstvo za Android Room bazu podataka i skladište 599

POGLAVLJE 74

Pristupanje skladištu u oblaku upotrebom Android Storage Access radnog okvira . 613

POGLAVLJE 75

Primer za Android Storage Access radni okvir 619

POGLAVLJE 76

Implementiranje video reprodukcije na Androidu
upotrebom klasa `VideoView` i `MediaController` 629

POGLAVLJE 77

Android Picture-in-Picture režim. 637

POGLAVLJE 78

Uputstvo za Android Picture-in-Picture režim643

POGLAVLJE 79

Snimanje video zapisa i fotografija na Androidu pomoću objekta `CameraIntent` . . .651

POGLAVLJE 80

Kreiranje zahteva za dozvolu izvršenja u Androidu657

POGLAVLJE 81

Snimanje i reprodukovanje audio fajlova na Androidu
upotrebom klasa `MediaPlayer` i `MediaRecorder`665

POGLAVLJE 82

Upotreba Google Maps Android API-ja u razvojnom okruženju Android Studio . . .677

POGLAVLJE 83

Štampanje pomoću Android Printing radnog okvira691

POGLAVLJE 84

Primer štampanja HTML-a i veb sadržaja na Androidu699

POGLAVLJE 85

Vodič za štampanje sopstvenog dokumenta na Androidu709

POGLAVLJE 86

Predstavljanje Android funkcije App Links723

POGLAVLJE 87

Uputstvo za Android Studio App Links.727

POGLAVLJE 88

Predstavljanje Android funkcije Instant Apps.739

POGLAVLJE 89

Uputstvo za Android funkciju Instant App.745

POGLAVLJE 90

Prilagođavanje Android Studio projekta za funkciju Instant App751

POGLAVLJE 91

Vodič za Android Studio alatu Profiler.757

POGLAVLJE 92

Uputstvo za Android biometrijsku proveru identiteta769

POGLAVLJE 93

Kreiranje, testiranje i slanje paketa Android aplikacije777

POGLAVLJE 94

Pregled Gradle sistema u Android Studiou795



SADRŽAJ

POGLAVLJE 1

Uvod	1
1.1 Preuzimanje primera koda	2
1.2 Povratne informacije	2
1.3 Štamparske greške	2

POGLAVLJE 2

Podešavanje Android Studio razvojnog okruženja	3
2.1 Sistemski zahtevi	3
2.2 Preuzimanje Android Studio paketa	3
2.3 Instaliranje Android Studioa	4
2.3.1 Instalacija na Windows	4
2.3.2 Instalacija na macOS	4
2.3.3 Instalacija na Linux	5
2.4 Android Studio Setup Wizard	5
2.5 Instaliranje dodatnih Android SDK paketa	6
2.6 Učinite dostupnom komandnu liniju Android SDK alati	9
2.6.1 Windows 7	9
2.6.2 Windows 8.1	10
2.6.3 Windows 10	11
2.6.4 Linux	11
2.6.5 macOS	11
2.7 Ažuriranje Android Studioa i SDK-a	11
2.8 Rezime	12

POGLAVLJE 3

Kreiranje primera Android aplikacije u Android Studiou	13
3.1 O projektu	13
3.2 Kreiranje novog Android projekta	13
3.3 Definisanje projekta i SDK podešavanja	15
3.4 Kreiranje aktivnosti	15
3.5 Modifikovanje primera aplikacije	17

3.6 Pregled rasporeda i fajlova resursa	25
3.7 Dodavanje interakcije	27
3.8 Rezime	28

POGLAVLJE 4

Pregled korisničkog interfejsa razvojnog okruženja Android Studio 29

4.1 Ekran dobrodošlice	29
4.2 Glavni prozor	30
4.3 Prozori alatke	31
4.4 Prečice na tastaturi Android Studioa	34
4.5 Switcher i pronalaženje nedavno otvaranih fajlova	34
4.6 Menjanje Android Studio teme	35
4.7 Rezime	36

POGLAVLJE 5

Kreiranje Android virtuelnog uređaja (AVD) u razvojnem okruženju Android Studio 37

5.1 O Android virtuelnim uređajima	37
5.2 Kreiranje novog AVD-a	38
5.3 Pokretanje emulatora	39
5.4 Pokretanje aplikacije u AVD-u	40
5.5 Run/Debug konfiguracije	41
5.6 Zaustavljanje pokrenute aplikacije	43
5.7 Kreiranje AVD-a iz komandne linije	43
5.8 Konfiguracioni fajlovi Android virtuelnog uređaja	44
5.9 Pomeranje i promena naziva Android virtuelnog uređaja	45
5.10 Rezime	45

POGLAVLJE 6

Upotreba i konfigurisanje Android Studio AVD emulatora 47

6.1 Okruženje emulatora	47
6.2 Opcije palete sa alatkama emulatora	48
6.3 Upotreba režima Zoom	49
6.4 Promena veličine prozora emulatora	49
6.5 Proširene kontrolne opcije	49
6.5.1 Location	50
6.5.2 Cellular	50
6.5.3 Camera	50
6.5.4 Battery	50
6.5.5 Phone	51
6.5.6 Directional Pad	51
6.5.7 Microphone	51
6.5.8 Fingerprint	51
6.5.9 Virtual Sensors	51
6.5.10 Snapshots	51
6.5.11 Screen Record	51
6.5.12 Google Play	51
6.5.13 Settings	52
6.5.14 Help	52
6.6 Upotreba snimaka	52

6.7 Podrška za prevlačenje i otpuštanje	54
6.8 Konfigurisanje simulacije otiska prsta	54
6.9 Rezime	56

POGLAVLJE 7

Testiranje Android Studio aplikacija na fizičkom Android uređaju 57

7.1 Pregled Android Debug Bridgea (ADB)	57
7.2 Omogućavanje ADB-a na Android uređajima	57
7.2.1 macOS ADB konfiguracija	58
7.2.2 Windows ADB konfiguracija	59
7.2.3 Linux adb konfiguracija	60
7.3 Testiranje adb konekcije	61
7.4 Rezime	62

POGLAVLJE 8

Osnove editora koda razvojnog okruženja Android Studio 63

8.1 Editor Android Studioa	63
8.2 Razdvajanje prozora editora	66
8.3 Završavanje koda	66
8.4 Završavanje iskaza	68
8.5 Informacije parametra	68
8.6 Nagoveštaji naziva parametra	69
8.7 Generisanje koda	69
8.8 Prelom koda	70
8.9 Brza pretraga dokumentacije	72
8.10 Ponovno formatiranje koda	72
8.11 Pronalaženje primera koda	73
8.12 Rezime	73

POGLAVLJE 9

Pregled Android arhitekture 75

9.1 Struktura Android programa	75
9.2 Linux Kernel	76
9.3 Android Runtime – ART	76
9.4 Android biblioteke	76
9.4.1 C/C++ biblioteke	77
9.5 Application Framework	78
9.6 Aplikacije	78
9.7 Rezime	78

POGLAVLJE 10

Anatomija Android aplikacije 79

10.1 Android Activity modul	79
10.2 Android fragmenti	80
10.3 Android Intent klase	80
10.4 Broadcast Intent klasa	80
10.5 Broadcast Receiver	80
10.6 Android Service	81

10.7 Content Provider	81
10.8 Application Manifest	81
10.9 Resursi aplikacije	81
10.10 Kontekst aplikacije	82
10.11 Rezime	82

POGLAVLJE 11

Uvod u Kotlin	83
11.1 Šta je Kotlin?	83
11.2 Kotlin i Java	83
11.3 Konvertovanje iz Java jezika u Kotlin	84
11.4 Kotlin i Android Studio	84
11.5 Eksperimentisanje sa Kotlinom	84
11.6 Znak tačka-zarez u Kotlinu	86
11.7 Rezime	86

POGLAVLJE 12

Kotlin tipovi podataka, promenljive i vrednost null	87
12.1 Kotlin tipovi podataka	87
12.1.1 Tipovi podataka celog broja	88
12.1.2 Tipovi podataka sa pokretnim zarezom	88
12.1.3 Logički tip podataka	88
12.1.4 Tip podataka karaktera	88
12.1.5 Tip podataka String	89
12.1.6 Izlazne sekvence	89
12.2 Izmenljive promenljive	90
12.3 Neizmenljive promenljive	90
12.4 Deklarisanje izmenljivih i neizmenljivih promenljivih	90
12.5 Tipovi podataka su objekti	91
12.6 Oznake tipa i utvrđivanje tipa	92
12.7 Tipovi koji prihvataju vrednost null	92
12.8 Operator bezbednog poziva	93
12.9 Not-Null tvrdnje	94
12.10 Tipovi koji prihvataju vrednost null i let funkcija	94
12.11 Operator Elvis	95
12.12 Eksplicitna konverzija tipa i provera tipa	96
12.13 Rezime	96

POGLAVLJE 13

Kotlin operatori i izrazi	97
13.1 Sintaksa izraza u Kotlinu	97
13.2 Osnovni operator dodele	97
13.3 Kotlin aritmetički operatori	98
13.4 Prošireni operatori dodele	98
13.5 Operatori inkrementiranja i dekrementiranja	99
13.6 Operatori jednakosti	99
13.7 Logički operatori	100
13.8 Operator opsega	101

13.9 Operatori nad bitovima.....	101
13.9.1 Inverzija nad bitovima.....	101
13.9.2 AND operator nad bitovima	102
13.9.3 OR operator nad bitovima	102
13.9.4 XOR operator nad bitovima	102
13.9.5 Operator nad bitovima pomaka ulevo.....	103
13.9.6 Operator nad bitovima pomaka udesno	103
13.10 Rezime	104

POGLAVLJE 14

Kotlin kontrola toka105

14.1 Kontrola toka ponavljanja u petlji.....	105
14.1.1 Kotlin for-in iskaz.....	105
14.1.2 Petlja while.....	106
14.1.3 Petlja do ... while	107
14.1.4 Izlazak iz petlji	107
14.1.5 Iskaz continue	108
14.1.6 Oznake break i continue	108
14.2 Uslovna kontrola toka	109
14.2.1 Upotreba izraza if	109
14.2.2 Upotreba iskaza if ... else	110
14.2.3 Upotreba izraza if ... else if	111
14.2.4 Upotreba iskaza when.....	111
14.3 Rezime	112

POGLAVLJE 15

Pregled Kotlin funkcija i lambda izraza113

15.1 Šta je funkcija?.....	113
15.2 Kako da deklarirate Kotlin funkciju.....	113
15.3 Pozivanje Kotlin funkcije.....	114
15.4 Funkcije sa jednim izrazom	114
15.5 Lokalne funkcije	114
15.6 Obrada vraćene vrednosti	115
15.7 Deklarisanje standardnih parametara funkcije	115
15.8 Promenljivi broj parametara funkcije	115
15.9 Lambda izrazi.....	116
15.10 Funkcije višeg reda.....	117
15.11 Rezime	118

POGLAVLJE 16

Osnove objektno-orijentisanog programiranja u Kotlinu119

16.1 Šta je objekat?.....	119
16.2 Šta je klasa?	119
16.3 Deklarisanje Kotlin klase	119
16.4 Dodavanje svojstava u klasu.....	120
16.5 Definisanje metoda	120
16.6 Deklarisanje i inicijalizovanje instance klase	121
16.7 Primarni i sekundarni konstruktori.....	121

16.8 Blokovi inicijalizatora	123
16.9 Pozivanje metoda i pristupanje svojstvima	123
16.10 Prilagođeni pristupni metodi	124
16.11 Ugneždene i unutrašnje klase	125
16.12 Prateći objekti	125
16.13 Rezime	128

POGLAVLJE 17

Uvod u Kotlin nasleđivanje i izradu potklase 129

17.1 Nasleđivanje, klase i potklase	129
17.2 Sintaksa za izradu potklase	129
17.3 Primer Kotlin nasleđivanja	130
17.4 Proširenje funkcionalnosti potklase	131
17.5 Menjanje vrednosti nasleđenih metoda	132
17.6 Dodavanje prilagođenog sekundarnog konstruktora	133
17.7 Uпотреba klase SavingsAccount	134
17.8 Rezime	134

POGLAVLJE 18

Razumevanje „životnih ciklusa“ Android aplikacije i aktivnosti 135

18.1 Android aplikacije i upravljanje resursima	135
18.2 Stanja Android procesa	135
Proces prednjeg plana	136
Vidljivi proces	136
Servisni proces	136
Pozadinski proces	136
Prazan proces	136
Viši prioritet	136
Niži prioritet	136
18.2.1 Proces prednjeg plana	136
18.2.2 Vidljivi proces	136
18.2.3 Servisni proces	136
18.2.4 Pozadinski proces	137
18.2.5 Prazan proces	137
18.3 Zavisnosti između procesa	137
18.4 „Životni ciklus“ aktivnosti	137
18.5 Stek aktivnosti	137
18.6 Stanja aktivnosti	138
18.7 Promene konfiguracije	139
18.8 Obrada promene stanja	139
18.9 Rezime	140

POGLAVLJE 19

Obrada promena stanja Android aktivnosti 141

19.1 Nove, nasuprot starih tehnika „životnog ciklusa“	141
19.2 Klase aktivnosti i fragmenata	141
19.3 Dinamičko stanje, nasuprot trajnog stanja	143
19.4 Android metodi „životnog ciklusa“	144

19.5 „Životni vek“	146
19.6 Isključivanje restartovanja konfiguracionih promena	146
19.7 Ograničenja metoda „životnog ciklusa“	147
19.8 Rezime	148

POGLAVLJE 20

Promene stanja Android aktivnosti kroz primer. 149

20.1 Kreiranje projekta primera promene stanja	149
20.2 Projektovanje korisničkog interfejsa	150
20.3 Promena vrednosti metoda „životnog ciklusa“ aktivnosti	151
20.4 Filtriranje panela Logcat	154
20.5 Pokretanje aplikacije	155
20.6 Eksperimentisanje sa aktivnošću	155
20.7 Rezime	156

POGLAVLJE 21

Snimanje i vraćanje stanja Android aktivnosti 157

21.1 Snimanje dinamičkog stanja	157
21.2 Standardna snimanja stanja korisničkog interfejsa	157
21.3 Klasa Bundle	158
21.4 Snimanje stanja	159
21.5 Vraćanje stanja	160
21.6 Testiranje aplikacije	161
21.7 Rezime	161

POGLAVLJE 22

Layout Editor alatka razvojnog okruženja Android Studio 163

22.1 Basic, nasuprot Empty Activity šablona	163
22.2 Layout Editor alatka Android Studioa	165
22.3 Design režim	165
22.4 Palette	166
22.5 Pan and Zoom	167
22.6 Prikazi Design i Layout	167
22.7 Text režim	168
22.8 Podešavanje atributa	169
22.9 Konfigurisanje omiljenih atributa	171
22.10 Konvertovanje prikaza	172
22.11 Prikazivanje primera podataka	173
22.12 Kreiranje definicije prilagođenog uređaja	174
22.13 Menjanje aktuelnog uređaja	175
22.14 Rezime	176

POGLAVLJE 23

Vodič za upotrebu rasporeda ConstraintLayout u razvojnem okruženju Android Studio 177

23.1 Prikazi Design i Layout	177
23.2 Autoconnect režim	179
23.3 Inference režim	179

23.4 Ručno manipulisanje ograničenjima	179
23.5 Dodavanje ograničenja u panelu Inspector	181
23.6 Brisanje ograničenja	181
23.7 Podešavanje odstupanja ograničenja	182
23.8 Razumevanje margina ograničenog rasporeda	183
23.9 Važnost suprotnih ograničenja i odstupanja	184
23.10 Konfigurisanje dimenzija vidžeta	186
23.11 Dodavanje vodica	188
23.12 Dodavanje barijera	189
23.13 Poravnanje i distribucija grupe vidžeta	191
23.14 Konvertovanje drugih rasporeda u raspored ConstraintLayout	193
23.15 Rezime	194

POGLAVLJE 24

Upotreba lanaca i razmera klase ConstraintLayout

u razvojnom okruženju Android Studio	195
24.1 Kreiranje lanca	195
24.2 Menjanje stila lanca	197
24.3 Stil lanca Spread Inside	198
24.4 Stil lanca Packed	199
24.5 Stil lanca Packed sa odstupanjem	199
24.6 Procenjeni lanac	199
24.7 Upotreba razmera	201
24.8 Rezime	202

POGLAVLJE 25

Uputstvo za klasu ConstraintLayout za Layout Editor alatku

razvojnog okruženja Android Studio	203
25.1 Primer Layout Editor alatke Android Studioa	203
25.2 Kreiranje nove aktivnosti	203
25.3 Pripremanje Layout Editor okruženja	205
25.4 Dodavanje vidžeta u korisnički interfejs	206
25.5 Dodavanje ograničenja	209
25.6 Testiranje rasporeda	211
25.7 Upotreba panela Layout Inspector	212
25.8 Rezime	212

POGLAVLJE 26

Ručno dizajniranje XML rasporeda u razvojnom okruženju Android Studio

26.1 Ručno kreiranje XML rasporeda	213
26.2 Ručni XML, nasuprot vizuelnog dizajna rasporeda	217
26.3 Rezime	217

POGLAVLJE 27

Upravljanje ograničenjima upotrebom skupova ograničenja.

27.1 Kotlin kod, nasuprot fajlova XML rasporeda	219
27.2 Kreiranje prikaza	220
27.3 Atributi prikaza	220

27.4 Skupovi ograničenja	220
27.4.1 Uspostavljanje veza	220
27.4.2 Primena ograničenja na raspored	221
27.4.3 Ograničenja u „roditeljskom“ rasporedu	221
27.4.4 Ograničenja promene veličine	221
27.4.5 Odstupanja ograničenja	221
27.4.6 Ograničenja poravnanja	222
27.4.7 Kopiranje i primena skupova ograničenja	222
27.4.8 Lanci rasporeda ConstraintLayout	222
27.4.9 Vođice	223
27.4.10 Uklanjanje ograničenja	223
27.4.11 Skaliranje	223
27.4.12 Rotacija	223
27.5 Rezime	224

POGLAVLJE 28

Uputstvo za Android klasu ConstraintSet225

28.1 Kreiranje primera projekta u Android Studiou	225
28.2 Dodavanje prikaza u aktivnost	225
28.3 Podešavanja atributa prikaza	227
28.4 Kreiranje ID-ja prikaza	227
28.5 Konfigurisanje klase ConstraintSet	228
28.6 Dodavanje EditText prikaza	229
28.7 Konvertovanje piksela nezavisnih od gustine (dp) u piksele (px)	230
28.8 Rezime	232

POGLAVLJE 29

Vodič za upotrebu funkcije Instant Run u razvojnom okruženju Android Studio . . .233

29.1 Predstavljanje funkcije Instant Run	233
29.2 Razumevanje nivoa zamene funkcije Instant Run	234
29.3 Uključivanje i isključivanje funkcije Instant Run	234
29.4 Upotreba funkcije Instant Run	235
29.5 Uputstvo za funkciju Instant Run	235
29.6 Pokretanje hot zamene funkcije Instant Run	236
29.7 Pokretanje warm zamene funkcije Instant Run	236
29.8 Pokretanje cold zamene funkcije Instant Run	237
29.9 Dugme Run	237
29.10 Rezime	238

POGLAVLJE 30

Pregled i primer Android funkcije za obradu događaja239

30.1 Razumevanje Android događaja	239
30.2 Upotreba resursa android:onClick	240
30.3 „Osluškivači“ događaja i metodi povratnog poziva	240
30.4 Primer obrade događaja	241
30.5 Dizajniranje korisničkog interfejsa	241
30.6 „Osluškivač“ događaja i metod povratnog poziva	243
30.7 Upotreba događaja	244
30.8 Rezime	246

POGLAVLJE 31

Obrada događaja dodira i višestrukog dodira na Androidu 247

31.1 Presretanje događaja dodira	247
31.2 MotionEvent objekat	248
31.3 Razumevanje akcija dodira	248
31.4 Obrada višestrukih dodira	248
31.5 Primer aplikacije sa višestrukim dodirima	249
31.6 Dizajniranje korisničkog interfejsa aktivnosti	249
31.7 Implementiranje „osluškivača“ događaja dodira	250
31.8 Pokretanje primera aplikacije	253
31.9 Rezime	254

POGLAVLJE 32

Detektovanje uobičajenih pokreta prstom upotrebom Android

Gesture Detector klase 255

32.1 Implementiranje detekcije uobičajenog pokreta prstom	255
32.2 Kreiranje primera projekta za detekciju pokreta prstom	256
32.3 Implementiranje klase „osluškivača“	256
32.4 Kreiranje instance GestureDetectorCompat	258
32.5 Implementiranje metoda onTouchEvent()	259
32.6 Testiranje aplikacije	259
32.7 Rezime	260

POGLAVLJE 33

Implementiranje prilagođenih pokreta prstom i prepoznavanje

primicanja i odmicanja prstiju na Androidu 261

33.1 Android Gesture Builder aplikacije	261
33.2 Klasa GestureOverlayView	261
33.3 Detektovanje pokreta	262
33.4 Identifikovanje specifičnih pokreta prstom	262
33.5 Izgradnja i pokretanje aplikacije Gesture Builder	262
33.6 Kreiranje fajla pokreta prstom	262
33.7 Kreiranje primera projekta	263
33.8 Ekstrahovanje fajla pokreta prstom sa SD kartice	263
33.9 Dodavanje fajla gestures u projekat	264
33.10 Dizajniranje korisničkog interfejsa	264
33.11 Učitavanje fajla gestures	265
33.12 Registracija „osluškivača“ događaja	266
33.13 Implementiranje metoda onGesturePerformed	266
33.14 Testiranje aplikacije	267
33.15 Konfigurisanje objekta GestureOverlayView	268
33.16 Presretanje pokreta prstom	268
33.17 Detektovanje primicanja i odmicanja prstiju	268
33.18 Primer projekta pokreta primicanja i odmicanja prstom	269
33.19 Rezime	271

POGLAVLJE 34**Uvod u Android fragmente273**

34.1 Šta je fragment?	273
34.2 Kreiranje fragmenta	273
34.3 Dodavanje fragmenta u aktivnost upotrebom XML fajla rasporeda	274
34.4 Dodavanje fragmenata u kodu i upravljanje njima	276
34.5 Obrada događaja fragmenta	277
34.6 Implementiranje komunikacije fragmenta	278
34.7 Rezime	280

POGLAVLJE 35**Upotreba fragmenata u razvojnom okruženju Android Studio – primer281**

35.1 O primeru aplikacije fragmenta	281
35.2 Kreiranje primera projekta	281
35.3 Kreiranje rasporeda prvog fragmenta	282
35.4 Kreiranje klase prvog fragmenta	284
35.5 Kreiranje rasporeda drugog fragmenta	285
35.6 Dodavanje fragmenata u aktivnost	286
35.7 Omogućite komunikaciju Toolbar fragmenta sa aktivnostima	288
35.8 Omogućite komunikaciju aktivnosti sa fragmentom text	291
35.9 Testiranje aplikacije	292
35.10 Rezime	292

POGLAVLJE 36**Moderna arhitektura Android aplikacije sa Jetpackom293**

36.1 Šta je Android Jetpack?	293
36.2 "Stara" arhitektura	293
36.3 Moderna Android arhitektura	294
36.4 ViewModel komponenta	294
36.5 Komponenta LiveData	295
36.6 LiveData komponenta i Data Binding biblioteka	296
36.7 „Životni ciklusi“ Androida	296
36.8 Moduli skladišta	297
36.9 Rezime	298

POGLAVLJE 37**Uputstvo za Android Jetpack komponentu ViewModel299**

37.1 O projektu	299
37.2 Kreiranje primera projekta ViewModel komponente	299
37.3 Pregled projekta	300
37.3.1 Glavna aktivnost	300
37.3.2 Fragment sadržaja	301
37.3.3 Komponenta ViewModel	302
37.4 Dizajniranje rasporeda fragmenta	302
37.5 Implementiranje komponente View Model	304
37.6 Povezivanje fragmenta sa komponentom View Model	304
37.7 Modifikovanje fragmenta	305

37.8	Pristupanje podacima komponente ViewModel	305
37.9	Testiranje projekta	306
37.10	Rezime	306

POGLAVLJE 38

Uputstvo za Android Jetpack LiveData komponentu 307

38.1	LiveData – rekapitulacija	307
38.2	Dodavanje komponente LiveData u komponentu ViewModel	307
38.3	Implementiranje prijemnika	309
38.4	Rezime	310

POGLAVLJE 39

Pregled Android JetpackData Binding biblioteke 311

39.1	Pregled biblioteke Data Binding	311
39.2	Ključne komponente povezivanja podataka	311
39.2.1	Konfiguracija verzije projekta	312
39.2.2	Fajl rasporeda povezivanja podataka	312
39.2.3	Element data fajla rasporeda	313
39.2.4	Klase povezivanja	314
39.2.5	Konfiguracija promenljive povezivanja podataka	315
39.2.6	Izrazi povezivanja (jednosmerni)	315
39.2.7	Izrazi povezivanja (dvosmerni)	317
39.2.8	Povezivanje događaja i „osluškivača“	317
39.3	Rezime	318

POGLAVLJE 40

Uputstvo za Android Jetpack povezivanje podataka 319

40.1	Uklanjanje suvišnog koda	319
40.2	Omogućavanje povezivanja podataka	321
40.3	Dodavanje elementa layout	322
40.4	Dodavanje elementa data u fajl rasporeda	323
40.5	Upotreba klase povezivanja	324
40.6	Dodela instance ViewModel u promenljivu povezivanja podataka	325
40.7	Dodavanje izraza povezivanja	325
40.8	Dodavanje metoda konverzije	326
40.9	Dodavanje povezivanja „osluškivača“	327
40.10	Testiranje aplikacije	327
40.11	Rezime	328

POGLAVLJE 41

Upotreba Android komponenata koje su svesne „životnog ciklusa“ 329

41.1	Svesnost „životnog ciklusa“	329
41.2	Vlasnici „životnog ciklusa“	329
41.3	LifecycleObserver interfejs	330
41.4	Stanja i događaji „životnog ciklusa“	331
41.5	Rezime	332

POGLAVLJE 42**Uputstvo za Android Jetpack svesnost „životnog ciklusa“ 333**

42.1 Kreiranje primera projekta „životnog ciklusa“	333
42.2 Kreiranje prijemnika „životnog ciklusa“	333
42.3 Dodavanje prijemnika	335
42.4 Testiranje prijemnika	335
42.5 Kreiranje vlasnika „životnog ciklusa“	336
42.6 Testiranje prilagođenog vlasnika „životnog ciklusa“	338
42.7 Rezime	338

POGLAVLJE 43**Pregled Android Navigation Architecture Componenta 339**

43.1 Razumevanje navigacije	339
43.2 Deklarisanje hosta navigacije.....	341
43.3 Grafikon navigacije	342
43.4 Pristupanje upravljaču navigacije	344
43.5 Pokretanje akcije navigacije.....	344
43.6 Prosleđivanje argumenata	344
43.7 Rezime	346

POGLAVLJE 44**Uputstvo za Android Jetpack komponentu navigacije 347**

44.1 Kreiranje projekta NavigationDemo.....	347
44.2 Dodavanje navigacije u konfiguraciju izvršne verzije.....	347
44.3 Kreiranje fajla resursa grafikona navigacije	348
44.4 Deklarisanje hosta navigacije.....	349
44.5 Dodavanje destinacija navigacije	351
44.6 Dizajniranje rasporeda fragmenta destinacije.....	353
44.7 Dodavanje akcije u grafikon navigacije	354
44.8 Implementiranje interfejsa OnFragmentInteractionListener.....	356
44.9 Pokretanje akcije	356
44.10 Prosleđivanje podataka pomoću dodatnog modula Safeargs.....	357
44.11 Rezime	360

POGLAVLJE 45**Kreiranje Overflow menija na Androidu i upravljanje njima 361**

45.1 Meni Overflow.....	361
45.2 Kreiranje menija Overflow	361
45.3 Prikazivanje menija Overflow	363
45.4 Odgovor na selekcije stavke menija	363
45.5 Kreiranje grupa stavki koje mogu da se potvrde	363
45.6 Meniji i Menu Editor alatakaAndroid Studioa	365
45.7 Kreiranje primera projekta.....	366
45.8 Dizajniranje menija	366
45.9 Modifikovanje metoda onOptionsItemSelected()	368
45.10 Testiranje aplikacije.....	370
45.11 Rezime	370

POGLAVLJE 46**Animacija korisničkog interfejsa pomoću Android Transitions radnog okvira371**

46.1 Predstavljane Android prelaza i scena	371
46.2 Upotreba interpolatora sa prelazima	372
46.3 Upotreba prelaza scene	373
46.4 Prilagođene klase Transition i TransitionSets u kodu	374
46.5 Prilagođene klase Transitions i TransitionSets u XML fajlu	375
46.6 Upotreba interpolatora	376
46.7 Kreiranje prilagođenog interpolatora	378
46.8 Upotreba metoda beginDelayedTransition	379
46.9 Rezime	380

POGLAVLJE 47**Uputstvo za Android prelaze upotrebom metoda beginDelayedTransition381**

47.1 Kreiranje Android Studio projekta TransitionDemo	381
47.2 Priprema fajlova projekta	381
47.3 Implementiranje animacije beginDelayedTransition	382
47.4 Prilagođavanje prelaza	384
47.5 Rezime	386

POGLAVLJE 48**Implementiranje Android prelaza scene – uputstvo387**

48.1 Pregled projekta prelaza scene	387
48.2 Kreiranje Android Studio SceneTransitions projekta	387
48.3 Identifikovanje i priprema osnovnog kontejnera	387
48.4 Dizajniranje prve scene	388
48.5 Dizajniranje druge scene	389
48.6 Ulazak u prvu scenu	390
48.7 Učitavanje scene 2	390
48.8 Implementiranje prelaza	391
48.9 Dodavanje fajla prelaza	391
48.10 Učitavanje i upotreba skupa prelaza	392
48.11 Konfigurisanje dodatnih prelaza	393
48.12 Rezime	394

POGLAVLJE 49**Upotreba plutajućeg komandnog dugmeta i Snackbara395**

49.1 Material Design	395
49.2 Biblioteka Design	395
49.3 Plutajuće komandno dugme (FAB)	396
49.4 Snackbar	397
49.5 Kreiranje primera projekta	397
49.6 Pregled projekta	397
49.7 Menjanje plutajućeg komandnog dugmeta	399
49.8 Dodavanje klase ListView u raspored sadržaja	400
49.9 Dodavanje stavki u prikaz ListView	401
49.10 Dodavanje akcije u Snackbar	403
49.11 Rezime	404

POGLAVLJE 50**Kreiranje interfejsa sa karticama upotrebom komponente TabLayout 405**

50.1 Predstavljanje klase ViewPager	405
50.2 Pregled komponente TabLayout	405
50.3 Kreiranje projekta TabLayoutDemo	406
50.4 Kreiranje prvog fragmenta	406
50.5 Dupliranje fragmenata	407
50.6 Dodavanje komponente TabLayout i klase ViewPager	408
50.7 Kreiranje pager adaptera	409
50.8 Izvršavanje zadataka inicijalizacije	410
50.9 Testiranje aplikacije	413
50.10 Prilagođavanje komponente TabLayout	413
50.11 Prikazivanje ikonica na karticama	415
50.12 Rezime	416

POGLAVLJE 51**Upotreba vidžeta RecyclerView i CardView 417**

51.1 Pregled klase RecyclerView	417
51.2 Pregled klase CardView	419
51.3 Dodavanje biblioteka u projekat	421
51.4 Rezime	422

POGLAVLJE 52**Uputstvo za Android komponente RecyclerView i CardView 423**

52.1 Kreiranje CardDemo projekta	423
52.2 Uklanjanje plutajućeg komandnog dugmeta	423
52.3 Dodavanje biblioteka RecyclerView i CardView	424
52.4 Dizajniranje rasporeda prikaza CardView	424
52.5 Dodavanje objekta RecyclerView	425
52.6 Kreiranje RecyclerView adaptera	426
52.7 Dodavanje fajlova slike	428
52.8 Inicijalizacija komponente RecyclerView	428
52.9 Testiranje aplikacije	429
52.10 Odgovaranje na selekcije kartice	430
52.11 Rezime	432

POGLAVLJE 53**Uputstvo za primer podataka Layout Editor alatke 433**

53.1 Dodavanje primera podataka u projekat	433
53.2 Upotreba prilagođenih primera podataka	438
53.3 Rezime	442

POGLAVLJE 54**Upotreba rasporeda AppBar i Collapsing Toolbar 443**

54.1 Anatomija palete aplikacije	443
54.2 Primer projekta	444
54.3 Koordinisanje komponenta RecyclerView i Toolbar	445
54.4 Predstavljanje rasporeda Collapsing ToolbarLayout	447

54.5 Menjanje naslova i boje senke	450
54.6 Rezime	452

POGLAVLJE 55

Implementiranje Android navigacione fioke 453

55.1 Pregled navigacione fioke.....	453
55.2 Otvaranje i zatvaranje fioke.....	455
55.3 Odgovaranje na selekcije stavke fioke.....	455
55.4 Upotreba šablona Navigation Drawer Activity.....	456
55.5 Kreiranje projekta šablona navigacione fioke	457
55.6 Fajl resursa rasporeda šablona.....	457
55.7 Fajl resursa bojenja zaglavlja	457
55.8 Fajl resursa menija šablona.....	457
55.9 Kod šablona	458
55.10 Pokretanje aplikacije	459
55.11 Rezime	460

POGLAVLJE 56

Uputstvo za Android Studio šablon Master/DetailFlow 461

56.1 Šablon Master/Detail Flow	461
56.2 Kreiranje Master/Detail Flow aktivnosti	462
56.3 Anatomija šablona Master/Detail Flow	464
56.4 Modifikovanje šablona Master/Detail Flow	465
56.5 Menjanje modela sadržaja.....	465
56.6 Menjanje panela detail.....	467
56.7 Modifikovanje klase WebsiteDetailFragment	468
56.8 Modifikovanje klase WebsiteListActivity	469
56.9 Dodavanje manifest dozvola	469
56.10 Pokretanje aplikacije	470
56.11 Rezime	470

POGLAVLJE 57

Pregled Android namere 471

57.1 Pregled namera.....	471
57.2 Eksplicitne namere.....	472
57.3 Vraćanje podataka iz aktivnosti	473
57.4 Implicitne namere	474
57.5 Upotreba filtera namere.....	474
57.6 Provera dostupnosti namere	475
57.7 Rezime	476

POGLAVLJE 58

Android eksplicitne namere – primenjeni primer 477

58.1 Kreiranje primera aplikacije eksplicitne namere	477
58.2 Dizajniranje rasporeda korisničkog interfejsa za aktivnost ActivityA	477
58.3 Kreiranje klase druge aktivnosti	479
58.4 Dizajniranje rasporeda korisničkog interfejsa	

za aktivnost ActivityB	479
58.5 Pregled manifest fajla aplikacije	480
58.6 Kreiranje namere	481
58.7 Izdvajanje podataka namere	482
58.8 Pokretanje aktivnosti ActivityB kao podaktivnosti	482
58.9 Vraćanje podataka iz podaktivnosti	484
58.10 Testiranje aplikacije	484
58.11 Rezime	484

POGLAVLJE 59

Android implicitne namere – primenjeni primer 485

59.1 Kreiranje Android Studio primera projekta implicitne namere	485
59.2 Dizajniranje korisničkog interfejsa	485
59.3 Kreiranje implicitne namere	486
59.4 Dodavanje druge poklapajuće aktivnosti	487
59.5 Dodavanje objekta WebView u korisnički interfejs	487
59.6 Preuzimanje URL-a namere	488
59.7 Modifikovanje manifest fajla projekta MyWebView	489
59.8 Instaliranje paketa MyWebView na uređaju	490
59.9 Testiranje aplikacije	491
59.10 Rezime	492

POGLAVLJE 60

Android objekti Broadcast Intents i Broadcast Receivers 493

60.1 Pregled objekta Broadcast Intent	493
60.2 Pregled elemenata Broadcast Receiver	494
60.3 Preuzimanje rezultata iz neusmerene poruke	495
60.4 „Lepljivi“ Broadcast Intenti	496
60.5 Primer za Broadcast Intent	496
60.6 Kreiranje primera aplikacije	496
60.7 Kreiranje i slanje objekta Broadcast Intent	496
60.8 Kreiranje klase Broadcast Receiver	497
60.9 Registracija Broadcast Receiver klase	498
60.10 Testiranje primera neusmerenih poruka	499
60.11 „Osluškiivanje“ sistemskih neusmerenih poruka	499
60.12 Rezime	500

POGLAVLJE 61

Pregled osnova programske niti i klase AsyncTasks 501

61.1 Pregled programskih niti	501
61.2 Glavna programska nit aplikacije	501
61.3 Procedure programske niti	501
61.4 Osnovni primer klase AsyncTask	502
61.5 Kreiranje potklase za klasu AsyncTask	504
61.6 Testiranje aplikacije	507
61.7 Poništavanje zadatka	507
61.8 Rezime	508

POGLAVLJE 62**Pregled Android pokrenutih i vezanih servisa 509**

62.1 Pokrenuti servisi	509
62.2 Intent servis	510
62.3 Vezani servis	510
62.4 Anatomija servisa	510
62.5 Kontrolisanje opcija za restartovanje uništenih servisa	511
62.6 Deklarisanje servisa u manifest fajlu	511
62.7 Pokretanje servisa prilikom pokretanja sistema	512
62.8 Rezime	513

POGLAVLJE 63**Implementiranje Android pokrenutog servisa – primenjeni primer 515**

63.1 Kreiranje primera projekta	515
63.2 Kreiranje klase Service	515
63.3 Dodavanje servisa u manifest fajl	516
63.4 Pokretanje servisa	517
63.5 Testiranje IntentService primera	518
63.6 Upotreba klase Service	518
63.7 Kreiranje novog servisa	518
63.8 Modifikovanje korisničkog interfejsa	520
63.9 Pokretanje aplikacije	520
63.10 Kreiranje klase AsyncTask za zadatke servisa	521
63.11 Rezime	522

POGLAVLJE 64**Android lokalni vezani servis – primenjeni primer 523**

64.1 Razumevanje vezanih servisa	523
64.2 Opcije interakcije vezanog servisa	523
64.3 Android Studio primer lokalnog vezanog servisa	524
64.4 Dodavanje vezanog servisa u projekat	524
64.5 Implementiranje objekta Binder	524
64.6 Povezivanje klijenta sa servisom	527
64.7 Završavanje primera	528
64.8 Testiranje aplikacije	529
64.9 Rezime	530

POGLAVLJE 65**Android udaljeni vezani servisi – primenjeni primer 531**

65.1 Komunikacija klijenta sa udaljenim servisom	531
65.2 Kreiranje primera aplikacije	531
65.3 Dizajniranje korisničkog interfejsa	532
65.4 Implementiranje udaljenog vezanog servisa	532
65.5 Konfigurisanje udaljenog servisa u manifest fajl	533
65.6 Pokretanje i povezivanje sa udaljenim servisom	534
65.7 Slanje poruke za udaljeni servis	535
65.8 Rezime	536

POGLAVLJE 66

Uputstvo za Android obaveštenja 537

66.1 Pregled obaveštenja	537
66.2 Kreiranje NotifyDemo projekta	539
66.3 Dizajniranje korisničkog interfejsa	539
66.4 Kreiranje druge aktivnosti	540
66.5 Kreiranje kanala obaveštenja.....	541
66.6 Kreiranje i izdavanje osnovnog obaveštenja.....	543
66.7 Pokretanje aktivnosti iz obaveštenja	545
66.8 Dodavanje akcija u obaveštenja	547
66.9 Grupna obaveštenja	548
66.10 Rezime	550

POGLAVLJE 67

Uputstvo za Android Direct Reply obaveštenja 551

67.1 Kreiranje projekta DirectReply	551
67.2 Dizajniranje korisničkog interfejsa	551
67.3 Kreiranje kanala obaveštenja.....	552
67.4 Izgradnja objekta RemoteInput.....	553
67.5 Kreiranje objekta PendingIntent	554
67.6 Kreiranje akcije odgovora.....	555
67.7 Primanje unosa direktnog odgovora	557
67.8 Ažuriranje obaveštenja	559
67.9 Rezime	560

POGLAVLJE 68

Uvod u Android podršku za više prozora 561

68.1 Režimi Split-Screen, Freeform i Picture-in-Picture	561
68.2 Aktiviranje režima za više prozora	562
68.3 Omogućavanje Freeform podrške.....	563
68.4 Provera Freeform podrške.....	563
68.5 Omogućavanje podrške za više prozora u aplikaciji	564
68.6 Specifikovanje atributa za više prozora.....	564
68.7 Detektovanje režima sa više prozora u aktivnosti	565
68.8 Primanje obaveštenja sa više prozora	566
68.9 Pokretanje aktivnosti u režimu sa više prozora	566
68.10 Konfigurisanje veličine i pozicije Freeform aktivnosti.....	567
68.11 Rezime	568

POGLAVLJE 69

Uputstvo za Android Studio režime sa više prozora Split-Screen i Freeform 569

69.1 Kreiranje projekta sa više prozora	569
69.2 Dizajniranje korisničkog interfejsa aktivnosti FirstActivity	569
69.3 Dodavanje druge aktivnosti.....	570
69.4 Pokretanje druge aktivnosti	570
69.5 Uključivanje režima sa više prozora	571
69.6 Testiranje podrške za više prozora.....	572
69.7 Pokretanje druge aktivnosti u drugom prozoru	573
69.8 Rezime	574

POGLAVLJE 70**Pregled Android SQLite baza podataka 575**

70.1 Razumevanje tabela baze podataka	575
70.2 Predstavljane šeme baze podataka	575
70.3 Kolone i tipovi podataka	576
70.4 Redovi baze podataka	576
70.5 Predstavljane primarnih ključeva	576
70.6 Šta je SQLite?	576
70.7 Structured Query Language (SQL)	577
70.8 Isprobavanje SQLitea na Android Virtual Deviceu (AVD)	577
70.9 Android Room Persistence biblioteka	579
70.10 Rezime	580

POGLAVLJE 71**Android Room Persistence biblioteka. 581**

71.1 Prisećanje na modernu arhitekturu aplikacije	581
71.2 Ključni elementi baze podataka Room persistence	582
71.2.1 Skladište	582
71.2.2 Room baza podataka	582
71.2.3 Data Access Object (DAO)	582
71.2.4 Entiteti	582
71.2.5 SQLite baza podataka	582
71.3 Razumevanje entiteta	583
71.4 Data Access Object	585
71.5 Room baza podataka	587
71.6 Skladište	588
71.7 Baze podataka u radnoj memoriji	589
71.8 Rezime	590

POGLAVLJE 72**Uputstvo za Android TableLayout****i TableRow 591**

72.1 Prikazi rasporeda TableLayout i TableRow	591
72.2 Kreiranje projekta Room baze podataka	592
72.3 Konvertovanje u LinearLayout raspored	592
72.4 Dodavanje prikaza TableLayout u korisnički interfejs	594
72.5 Konfigurisanje objekata TableRow	594
72.6 Dodavanje trake sa dugmadima u raspored	596
72.7 Dodavanje instance RecyclerView	596
72.8 Podešavanja margina rasporeda	598
72.9 Rezime	598

POGLAVLJE 73**Uputstvo za Android Room bazu podataka i skladište 599**

73.1 O RoomDemo projektu	599
73.2 Modifikovanje konfiguracije verzije	599
73.3 Izgradnja entiteta	600
73.4 Kreiranje Data Access Objecta	601

73.5 Dodavanje Room baze podataka	603
73.6 Dodavanje skladišta	604
73.7 Modifikovanje objekta ViewModel	607
73.8 Kreiranje rasporeda stavki proizvoda	608
73.9 Dodavanje RecyclerView adaptera	608
73.10 Priprema glavnog fragmenta	610
73.11 Dodavanje „osluškivača“ dugmadi	611
73.12 Dodavanje LiveData prijemnika	611
73.13 Inicijalizacija prikaza RecyclerView	612
73.14 Testiranje aplikacije RoomDemo	612
73.15 Rezime	612

POGLAVLJE 74

Pristupanje skladištu u oblaku upotrebom Android Storage Access radnog okvira . .613

74.1 Storage Access radni okvir	613
74.2 Upotreba Storage Access radnog okvira	614
74.3 Filtriranje liste fajlova birača	614
74.4 Obrada rezultata objekta Intent	615
74.5 Čitanje sadržaja fajla	616
74.6 Pisanje sadržaja u fajl	617
74.7 Brisanje fajla	617
74.8 Dobijanje trajnog pristupa fajlu	618
74.9 Rezime	618

POGLAVLJE 75

Primer za Android Storage Access radni okvir619

75.1 O primeru Storage Access radnog okvira	619
75.2 Kreiranje primera Storage Access radnog okvira	619
75.3 Dizajniranje korisničkog interfejsa	619
75.4 Deklarisanje kodova zahteva	620
75.5 Kreiranje novog fajla skladišta	621
75.6 Metod onActivityResult()	622
75.7 Snimanje u fajl skladišta	624
75.8 Otvaranje i čitanje fajla skladišta	626
75.9 Testiranje aplikacije za pristup skladištu	628
75.10 Rezime	628

POGLAVLJE 76

Implementiranje video reprodukovanja na Androidu upotrebom klasa VideoView i MediaController629

76.1 Predstavljanje Android VideoView klase	629
76.2 Predstavljanje Android MediaController klase	630
76.3 Kreiranje primera reprodukovanja video snimka	631
76.4 Dizajniranje rasporeda VideoPlayer	631
76.5 Konfigurisanje klase VideoView	632
76.6 Dodavanje dozvole za Internet	633
76.7 Dodavanje klase MediaController u klasu Video View	634
76.8 Podešavanje „osluškivača“ onPreparedListener	635
76.9 Rezime	636

POGLAVLJE 77**Android Picture-in-Picture režim637**

77.1 Funkcije režima Picture-in-Picture	637
77.2 Uključivanje režima Picture-in-Picture	638
77.3 Konfigurisanje parametara Picture-in-Picture	639
77.4 Ulaz u režim Picture-in-Picture.....	639
77.5 Detektovanje promena režima Picture-in-Picture	640
77.6 Dodavanje akcija Picture-in-Picture	640
77.7 Rezime	642

POGLAVLJE 78**Uputstvo za Android****Picture-in-Picture režim643**

78.1 Dodavanje Picture-in-Picture podrške u manifest fajl.....	643
78.2 Dodavanje dugmeta Picture-in-Picture	643
78.3 Ulazak u režim Picture-in-Picture	644
78.4 Detektovanje promena Picture-in-Picture režima	646
78.5 Dodavanje klase Broadcast Receiver.....	646
78.6 Dodavanje PiP akcije	647
78.7 Testiranje akcije Picture-in-Picture.....	650
78.8 Rezime	650

POGLAVLJE 79**Snimanje video zapisa i fotografija na Androidu pomoću objekta CameraIntent . . .651**

79.1 Provera podrške za kameru	651
79.2 Pozivanje namere za snimanje video zapisa	651
79.3 Pozivanje namere za snimanje fotografije.....	653
79.4 Kreiranje Android Studio projekta za snimanje video zapisa.....	653
79.5 Dizajniranje rasporeda korisničkog interfejsa.....	653
79.6 Provera kamere.....	654
79.7 Pokretanje namere za snimanje video zapisa	655
79.8 Obrada vraćanja namere	655
79.9 Testiranje aplikacije.....	656
79.10 Rezime	656

POGLAVLJE 80**Kreiranje zahteva za dozvolu izvršenja u Androidu657**

80.1 Razumevanje normalnih i opasnih dozvola	657
80.2 Kreiranje projekta primera dozvola	659
80.3 Provera dozvola.....	659
80.4 Zahtevanje dozvole u vreme izvršenja.....	661
80.5 Obezbeđivanje obrazloženja za zahtev za dozvolu	662
80.6 Testiranje dozvola aplikacije	664
80.7 Rezime	664

POGLAVLJE 81**Snimanje i reprodukovanje audio fajlova na Androidu upotrebom
klasa MediaPlayer i MediaRecorder665**

81.1 Reprodukovanje audio fajlova	665
81.2 Snimanje audio i video zapisa pomoću klase MediaRecorder	666
81.3 O primeru projekta	667
81.4 Kreiranje projekta AudioApp	667
81.5 Dizajniranje korisničkog interfejsa	667
81.6 Provera dostupnosti mikrofona	668
81.7 Izvršavanje inicijalizacije aktivnosti	669
81.8 Implementiranje metoda recordAudio()	670
81.9 Implementiranje metoda stopAudio()	671
81.10 Implementiranje metoda playAudio()	671
81.11 Konfigurisanje i zahtevanje dozvola	672
81.12 Testiranje aplikacije	675
81.13 Rezime	676

POGLAVLJE 82**Upotreba Google Maps Android API-ja u razvojnom okruženju Android Studio677**

82.1 Elementi Google Maps Android API-ja	677
82.2 Kreiranje Google Maps projekta	678
82.3 Preuzimanje programerskog potpisa	678
82.4 Testiranje aplikacije	679
82.5 Razumevanje geokodiranja i obrnutog geokodiranja	680
82.6 Dodavanje mape u aplikaciju	682
82.7 Traženje dozvole za aktuelnu lokaciju	682
82.8 Prikazivanje aktuelne lokacije korisnika	683
82.9 Menjanje tipa mape	685
82.10 Prikazivanje kontrola mape korisniku	686
82.11 Obrada interakcije pokreta prstom na mapi	686
82.11.1 Pokreti prstom za promenu uveličanja mape	686
82.11.2 Pokreti prstom za skrolovanje/pomeranje mape	687
82.11.3 Pokreti prstiju za nagib mape	687
82.11.4 Pokreti prstiju za rotiranje mape	687
82.12 Kreiranje oznaka mape	687
82.13 Kontrolisanje kamere mape	688
Rezime	690

POGLAVLJE 83**Štampanje pomoću Android Printing radnog okvira691**

83.1 Arhitektura Android štampanja	691
83.2 Dodatni moduli servisa za štampanje	691
83.3 Google Cloud Print	692
83.4 Skladištenje u Google Drive	692
83.5 Snimanje sadržaja kao PDF dokumenta	693
83.6 Štampanje sa Android uređaja	693
83.7 Opcije za ugradnju podrške za štampanje u Android aplikacije	694
83.7.1 Image Printing	694

83.7.2 Kreiranje i štampanje HTML sadržaja	695
83.7.3 Štampanje veb stranice	697
83.8 Rezime	698

POGLAVLJE 84

Primer štampanja HTML-a

i veb sadržaja na Androidu	699
----------------------------------	-----

84.1 Kreiranje primera aplikacije za štampanje HTML-a	699
84.2 Štampanje dinamičkog HTML sadržaja	699
84.3 Kreiranje primera štampanja veb stranice	702
84.4 Uklanjanje plutajućeg komandnog dugmeta	702
84.5 Dizajniranje rasporeda korisničkog interfejsa	702
84.6 Učitavanje veb stranice u objekat WebView	704
84.7 Dodavanje Print opcije menija	705
84.8 Rezime	708

POGLAVLJE 85

Vodič za štampanje sopstvenog dokumenta na Androidu	709
---	-----

85.1 Pregled štampanja sopstvenog dokumenta na Androidu	709
85.1.1 Prilagođeni adapteri za štampanje	709
85.2 Pripremanje projekta štampanja sopstvenog dokumenta	710
85.3 Kreiranje prilagođenog adaptera za štampanje	711
85.4 Implementiranje metoda povratnog poziva onLayout()	712
85.5 Implementiranje metoda povratnog poziva onWrite()	715
85.6 Provera stranice u rasponu	718
85.7 Iscrtavanje sadržaja na radnoj površini stranice	719
85.8 Pokretanje zadatka štampe	721
85.9 Testiranje aplikacije	722
85.10 Rezime	722

POGLAVLJE 86

Predstavljanje Android funkcije App Links	723
---	-----

86.1 Pregled Android funkcije App Links	723
86.2 App Link filteri namere	723
86.3 Obrada App Link namera	724
86.4 Povezivanje aplikacije sa veb sajtom	725
86.5 Rezime	726

POGLAVLJE 87

Uputstvo za Android Studio App Links.	727
--	-----

87.1 O primeru aplikacije	727
87.2 Šema baze podataka	727
87.3 Učitavanje i pokretanje projekta	728
87.4 Dodavanje mapiranja URL-a	729
87.5 Dodavanje filtera namere	732
87.6 Dodavanje koda za obradu namere	733
87.7 Testiranje funkcije App Links	736
87.8 Povezivanje linka aplikacije sa veb sajtom	737
87.9 Rezime	738

POGLAVLJE 88**Predstavljanje Android funkcije Instant Apps 739**

88.1 Pregled Android funkcije Instant App	739
88.2 Moduli funkcije Instant App.....	740
88.3 Struktura projekta Instant App	740
88.4 Dodatni moduli izgradnje aplikacije i funkcije	741
88.5 Instaliranje Instant Apps Development SDK-a	742
88.6 Rezime	744

POGLAVLJE 89**Uputstvo za Android funkciju Instant App 745**

89.1 Kreiranje projekta Instant App	745
89.2 Pregled projekta	746
89.3 Testiranje aplikacije koja može da se instalira	748
89.4 Testiranje funkcije Instant App	748
89.5 Pregled Instant App APK fajlova	749
89.6 Rezime	750

POGLAVLJE 90**Prilagođavanje Android Studio projekta za funkciju Instant App 751**

90.1 Početak rada	751
Kreiranje osnovnog modula funkcije	751
90.2 Dodavanje APK modula aplikacije	752
90.3 Dodavanje modula Instant App	754
90.4 Testiranje funkcije Instant App	755
90.5 Rezime	756

POGLAVLJE 91**Vodič za Android Studio
alatku Profiler 757**

91.1 Pristupanje Android Profiler alatki	757
91.2 Omogućavanje naprednog profilisanja	758
91.3 Prozor Android Profiler alatke	758
91.4 Sessions panel	759
91.5 CPU Profiler prozor	761
91.6 Memory Profiler prozor	764
91.7 Network Profiler prozor	766
91.8 Energy Profiler prozor	767
91.9 Rezime	768

POGLAVLJE 92**Uputstvo za Android biometrijsku proveru identiteta 769**

92.1 Pregled biometrijske provere identiteta	769
92.2 Kreiranje projekta biometrijske provere identiteta	770
92.3 Konfigurisanje provere identiteta otiskom prsta na uređaju	770
92.4 Dodavanje biometrijske dozvole u manifest fajl	771
92.5 Dizajniranje korisničkog interfejsa	771
92.6 Dodavanje pomoćnog metoda za poruke na ekranu	772

92.7 Provera podešavanja bezbednosti	772
92.8 Konfigurisanje povratnih poziva provere identiteta	773
92.9 Dodavanje klase CancellationSignal	774
92.10 Pokretanje klase BiometricPrompt	775
92.11 Testiranje projekta	775
92.12 Rezime	776

POGLAVLJE 93

Kreiranje, testiranje i slanje paketa Android aplikacije 777

93.1 Proces pripreme izdanja	777
93.2 Android App Bundle	777
93.3 Registracija Google Play Developer Console naloga	778
93.4 Konfigurisanje aplikacije u konzoli	779
93.5 Omogućavanje opcije Google Play App Signing	779
93.6 Uključivanje funkcije ProGuard	780
93.7 Kreiranje fajla Keystore	781
93.8 Kreiranje Android App Bundle paketa	783
93.9 Generisanje APK fajlova za test	784
93.10 Slanje paketa aplikacije u Google Play Developer Console	785
93.11 Istraživanje paketa aplikacije	786
93.12 Upravljanje izvršiocima testiranja	788
93.13 Slanje Instant App paketa	789
93.14 Slanje nove ispravke paketa aplikacije	790
93.15 Analiziranje fajla paketa aplikacije	792
93.16 Omogućavanje Google Play Signing opcije za postojeću aplikaciju	793
93.17 Rezime	794

POGLAVLJE 94

Pregled Gradle sistema u Android Studiou 795

94.1 Pregled Gradle sistema	795
94.2 Gradle i Android Studio	795
94.2.1 Razumna standardna podešavanja	795
94.2.2 Zavisnosti	796
94.2.3 Varijante izgradnje aplikacije	796
94.2.4 Unosi u manifest fajlu	796
94.2.5 Potpisivanje APK fajlova	796
94.2.6 ProGuard podrška	797
94.3 Gradle fajl izgradnje aplikacije najvišeg nivoa	797
94.4 Gradle fajlovi izgradnje aplikacije nivoa modula	798
94.5 Konfigurisanje podešavanja potpisa u fajlu izgradnje aplikacije	800
94.6 Pokretanje Gradle zadataka iz komandne linije	801
94.7 Rezime	802



UVOD

„Google“ je 2018. godine predstavio Android Jetpack zajednici programera. Projektovan da ubrza i olakša razvoj modernih i pouzdanih Android aplikacija, Jetpack se sastoji od skupa alati, biblioteka i arhitekturnih smernica. Glavni elementi Android Jetpacka se sastoje od Android Studio Integrated Development Environmenta (IDE), Android Architecture Componentsa i Modern App Architecture Guidelinesa - svi su opisani u ovom najnovijem izdanju knjige „Android 9, Kotlin i Android Studio 3.2“.

Potpuno ažurirana za Android Studio 3.2 i Android 9, ova knjiga treba da vas nauči veštinama potrebnim za razvoj aplikacija koje su zasnovane na Androidu korišćenjem Kotlin programskog jezika.

Knjiga započinje osnovama i sadrži opis potrebnih koraka za podešavanje Android razvojnog okruženja i okruženja za testiranje, a zatim je predstavljeno programiranje u Kotlinu, uključujući i tipove podataka, kontrolu toka, funkcije, lambda funkcije i objektno-orijentisano programiranje.

Pregled Android Studioa je uključen – u njemu su opisane oblasti kao što su prozori alati, editor koda i alatka LayoutEditor. Nakon uvoda o arhitekturi Androida, sledi detaljan pregled projektovanja Android aplikacija i korisničkih interfejsa upotrebom Android Studio okruženja.

Takođe postoje poglavlja u kojima se opisuju Android Architecture Components, uključujući modele prikaza, upravljanje „životnim ciklusom“ komponenata, pristup Room bazi podataka, navigaciju aplikacije, „žive“ podatke i vezivanje podataka.

Naprednije teme, kao što su intent klase, takođe su opisane, što uključuje obradu ekrana osetljivog na dodir, prepoznavanje pokreta, pristup kameri i reprodukovanje i snimanje video i audio zapisa. U ovom izdanju knjige takođe su obrađeni štampanje, prelazi i skladištenje fajlova na cloudu.

Koncepti projektovanja materijala su detaljno opisani, uključujući upotrebu plutajućih dugmadi akcije, Snackbars, interfejs sa karticama, prikaze na karticama, navigaciju i sakupljajuće trake sa alatcima.

Osim opisa osnovnih tehnika Android razvoja, knjiga uključuje i teme specifične za Google Play, kao što su implementiranje mapa pomoću Google Maps Android API-ja i slanje aplikacija na Google PlayDeveloper Consoleu.

Ostale funkcije Android Studioa 3.2 i Androida 9 su takođe opisane detaljno, uključujući Layout Editor, klase ConstraintLayout i ConstraintSet, ograničen lanac pravila i barijera, direktna obaveštenja o odgovorima i podršku za više prozora.

Poglavlja obuhvataju i napredne funkcije Android Studioa, kao što su App Links, Instant Apps, Android StudioProfiler i Gradle build konfiguracija.

Pod pretpostavkom da već imate iskustva u programiranju, da ste spremni da preuzmete Android Studio i AndroidSDK i da imate pristup Windows, Mac ili Linux sistemu i ideje za razvoj nekih aplikacija, spremni ste da započnete rad.

1.1 Preuzimanje primera koda

Izvorni kod i fajlovi Android Studio projekta za primere sadržane u ovoj knjizi dostupni su za preuzimanje na adresi

<https://www.ebookfrenzy.com/retail/as32kotlin/index.php>

Koraci za učitavanje projekta iz primera koda u Android Studio su sledeći:

1. Iz okvira za dijalog Welcome to Android Studio selektujte opciju Open an existing Android Studio project.
2. U okviru za dijalog za selekciju projekta potražite i selektujte direktorijum sa projektom koji će biti importovan i kliknite na OK.

1.2 Povratne informacije

Mi želimo da budete zadovoljni kupovinom ove knjige. Ako pronađete neke greške u knjizi ili imate neki komentar, pitanje ili problem, molimo vas da sa nama kontaktirate na adresi feedback@ebookfrenzy.com.

1.3 Štamparske greške

Iako se trudimo da obezbedimo tačnost sadržaja ove knjige, neizbežno je da se u knjizi koja obuhvata ovoliko oblasti i složenosti potkradu i neke greške i previdi. Svaka prepoznata greška u knjizi će biti istaknuta, zajedno sa rešenjem, na URL-u

<https://www.ebookfrenzy.com/errata/as32kotlin.html>

U slučaju da pronađete grešku koja nije izlistana u odeljku Errata, molimo vas da o tome obavestite naš tim tehničke podrške e-mailom na adresu feedback@ebookfrenzy.com. Taj tim postoji da bi vam pomogao pri rešavanju problema sa kojima se suočavate.



PODEŠAVANJE ANDROID STUDIO RAZVOJNOG OKRUŽENJA

Pre nego što započnemo bilo kakav posao na razvoju Android aplikacije, prvi korak je da konfigurišemo računarski sistem da se ponaša kao razvojna platforma. To uključuje veliki broj koraka koji se sastoje od instalacije Android Studio Integrated Development Environmenta (IDE), koji uključuje Android Software Development Kit(SDK), Kotlin dodatni modul i OpenJDK Java razvojno okruženje.

U ovom poglavlju opisani su koraci za instalaciju komponenta potrebnih za razvoj Android aplikacije na Windows, macOS i Linux sistemima.

2.1 Sistemski zahtevi

Razvoj Android aplikacije može da se vrši na bilo kojem od sledećih tipova sistema:

- Windows 7/8/10 (32-bitni ili 64-bitni)
- macOS 10.10 ili noviji (samo sistemi zasnovani na Intelu)
- Linux sistemi sa verzijom 2.19 ili najnovijom GNU C bibliotekom (glibc)
- minimum 3 GB RAM-a (8 GB je najbolje)
- prosečno 4 GB slobodnog prostora na disku
- minimalna rezolucija ekrana 1.280x800

2.2 Preuzimanje Android Studio paketa

Veći deo posla koji je uključen u razvoj aplikacija za Android biće izvršen upotrebom Android Studio okruženja. Sadržaj i primeri u ovoj knjizi kreirani su u Android Studio verziji 3.2.

Međutim, Android Studio se često ažurira, pa će možda postojati i novije verzije kada ova knjiga bude izdata.

Najnovija verzija Android Studioa može da se preuzme sa glavne stranice za preuzimanje na URL-u <https://developer.android.com/studio/index.html>

Ako ova stranica obezbeđuje instrukcije za preuzimanje novije verzije Android Studioa, možda će postojati neke male razlike između ove knjige i softvera. Veb pretraga za Android Studio 3.2 treba da obezbedi opciju za preuzimanje starije verzije u slučaju da ove razlike predstavljaju problem.

2.3 Instaliranje Android Studioa

Kada je preuzet, koraci za instaliranje Android Studioa se razlikuju, u zavisnosti od operativnog sistema na kojem će instalacija biti izvršena.

2.3.1 Instalacija na Windows

Potražite izvršni fajl Android Studio instalacije (pod nazivom `android-studio-bundle-<version>.exe`) u prozoru Windows Explorera i dvostruko kliknite na njega da biste započeli proces instalacije i kliknite na dugme Yes u okviru za dijalog User Account Control, ako se pojavi.

Kada bude prikazan Android Studio setup wizard, izvršite potrebne korake na različitim ekranima da biste konfigurisali instalaciju i izabrali podešavanja u pogledu lokacije fajl sistema na koji bi Android Studio trebalo da bude instaliran i da definišete da li bi trebalo da on bude dostupan drugim korisnicima sistema ili ne. Kada se zatraži da selektujete komponente za instaliranje, uverite se da su selektovane opcije Android Studio, Android SDK i Android Virtual Device.

Iako ne postoje striktna pravila gde bi trebalo da bude instaliran Android Studio na sistemu, u ovoj knjizi ćemo pretpostaviti da je instalacija izvršena na lokaciji `C:\Program Files\Android\Android Studio`, a da su Android SDK paketi instalirani u poddirektorijum korisnika `AppData\Local\Android\sdk`. Kada su konfigurisane opcije, kliknite na dugme Install da biste započeli proces instalacije.

Na verzijama Windowsa sa Start menijem novoinstalirani Android Studio može da bude pokrenut iz unosa dodatog u taj meni u toku instalacije. Izvršni fajl može da bude prikazan u taskbar za lakši pristup, tako što ćete otvoriti direktorijum `Android Studio\bin`, kliknuti desnim tasterom miša na izvršni fajl i selektovati opciju Pin to Taskbar. Imajte na umu da je izvršni fajl obezbeđen u 32-bitnim (*studio*) i 64-bitnim (*studio64*) izvršnim verzijama. Ako pokrećete 32-bitni sistem, upotrebite *studio* izvršni fajl.

2.3.2 Instalacija na macOS

Android Studio for macOS se preuzima u formi disk imidž (`.dmg`) fajla. Kada je preuzet fajl `android-studio-ide-<version>.dmg`, pronađite ga u prozoru Findera i dvostruko kliknite na njega da biste ga otvorili, kao što je prikazano na slici 2-1.



Slika 2-1

Da biste instalirali paket, jednostavno prevucite ikonicu Android Studioa i otpustite je u direktorijum Application. Android Studio paket će tada biti instaliran u direktorijumu Application na sistemu; taj proces će, obično, trajati nekoliko minuta.

Da biste pokrenuli Android Studio, potražite izvršni fajl u direktorijumu Applications, koristeći prozor Finder, i dvostruko kliknite na njega.

Da biste ubuduće lakše pristupili alatki, prevucite ikonicu Android Studioa iz prozora Finder i otpustite je u dok.

2.3.3 Instalacija na Linux

Ako ste preuzeli Linux Android Studio paket, otvorite ga u prozoru terminala, promenite direktorijum za lokaciju gde će Android Studio biti instaliran i izvršite sledeću komandu:

```
unzip /<path to package>/android-studio-ide-<version>-linux.zip
```

Imajte na umu da će Android Studio biti instaliran u poddirektorijumu pod nazivom android-studio. Prema tome, ako pretpostavimo da je prethodna komanda izvršena u direktorijumu /home/demo, softverski paket će biti raspakovan u direktorijum /home/demo/android-studio.

Da biste pokrenuli Android Studio, otvorite prozor terminala, promenite direktorijum na android-studio/bin poddirektorijum i izvršite sledeću komandu:

```
./studio.sh
```

Kada pokrećete 64-bitni Linux sistem, potrebno je da instalirate neke 32-bitne biblioteke podrške da bi Android Studio bio pokrenut. Na Ubuntu sistemu obe biblioteke mogu da budu instalirane upotrebom sledeće komande:

```
sudo apt-get install libc6:i386 libncurses5:i386 libstdc++6:i386 lib32z1 libbz2-1.0:i386
```

Na RedHat i Fedora 64-bitnim sistemima upotrebite sledeću komandu:

```
sudo yum install zlib.i686 ncurses-libs.i686 bzip2-libs.i686
```

2.4 Android Studio Setup Wizard

Kada je Android Studio pokrenut prvi put nakon što je instaliran, biće prikazan okvir za dijalog koji obezbeđuje opcije za importovanje podešavanja iz prethodne verzije Android Studioa. Ako imate podešavanja iz prethodne verzije i želite da ih importujete u najnoviju instalaciju, selektujte odgovarajuće opcije i lokacije. Alternativno, izaberite da ne želite da importujete prethodna podešavanja i kliknite na dugme OK da biste nastavili rad.

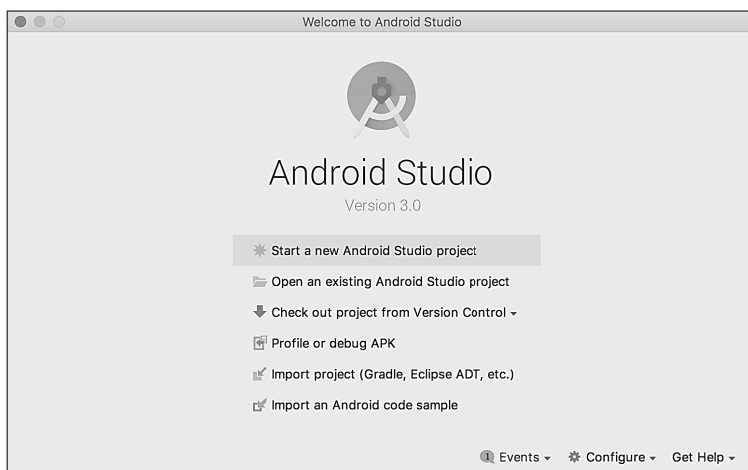
Setup Wizard može izgledati kao na slici 2-2, mada ovaj okvir za dijalog neće biti prikazan na svim platformama:



Slika 2-2

Ako se wizard prikaže, kliknite na dugme Next, izaberite opciju Standard installation i ponovo kliknite na Next.

Android Studio će nastaviti preuzimanje i konfiguirisanje najnovijeg Android SDK-a i nekih dodatnih komponenata i paketa. Kada je ovaj proces završen, kliknite na dugme Finish u okviru za dijalog Downloading Components i tada bi trebalo da se otvori ekran Welcome to Android Studio:

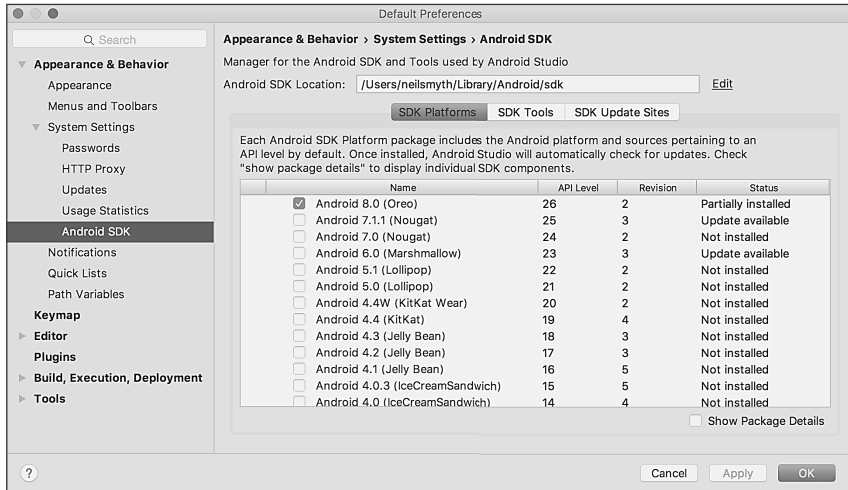


Slika 2-3

2.5 Instaliranje dodatnih Android SDK paketa

Do sada su izvršeni koraci za instaliranje Java programskog jezika, Android Studio IDE-a i aktuelnog skupa standardnih AndroidSDK paketa. Pre nego što nastavite, treba da utvrdite koji paketi su instalirani i da instalirate nedostajuće pakete ili ažuriranja.

Ovaj zadatak se izvršava pomoću Android SDK Settings ekrana, koji može da se otvori unutar Android Studio alatke selektovanjem opcije Configure -> SDK Manager unutar Android Studio welcome okvira za dijalog. Kada je pozvan, biće prikazan Android SDK ekran sa okvirom za dijalog sa podešenim standardnim opcijama, kao što je prikazano na slici 2-4.



Slika 2-4

Odmah nakon što je Android Studio instaliran prvi put, verovatno će biti instalirana samo najnovija verzija Android SDK-a. Da biste instalirali starije verzije, jednostavno selektujte polja za potvrđivanje koja odgovaraju tim verzijama i kliknite na dugme Apply.

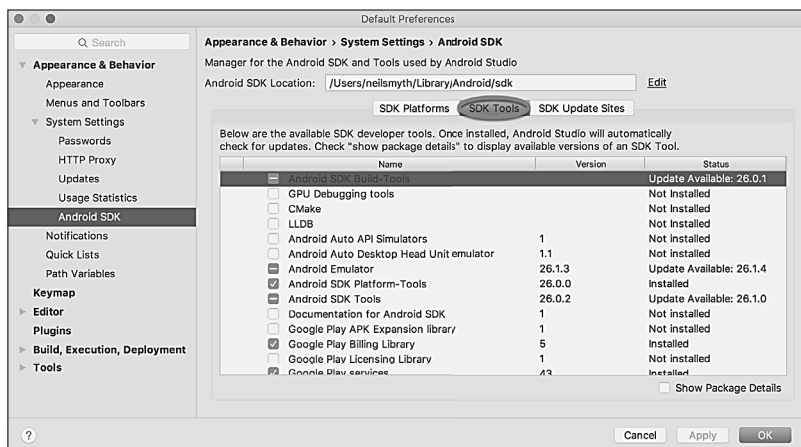
Takođe je moguće da će ažuriranja biti izlistana kao dostupna za najnoviji SDK. Da biste pristupili detaljnim informacijama o paketima koji su dostupni za ažuriranje, uključite opciju Show Package Details, koja se nalazi u donjem desnom uglu ekrana. Biće prikazane informacije koje su slične onima prikazanim na slici 2-5.

	Name	API Level	Revision	Status
<input type="checkbox"/>	Android TV Intel x86 Atom System Image	25	6	Not installed
<input type="checkbox"/>	Android Wear for China ARM EABI v7a System Image	25	3	Not installed
<input type="checkbox"/>	Android Wear for China Intel x86 Atom System Image	25	3	Not installed
<input type="checkbox"/>	Android Wear ARM EABI v7a System Image	25	3	Not installed
<input type="checkbox"/>	Android Wear Intel x86 Atom System Image	25	3	Not installed
<input type="checkbox"/>	Google APIs ARM 64 v8a System Image	25	8	Not installed
<input type="checkbox"/>	Google APIs ARM EABI v7a System Image	25	8	Not installed
<input type="checkbox"/>	Google APIs Intel x86 Atom System Image	25	8	Not installed
<input checked="" type="checkbox"/>	Google APIs Intel x86 Atom_64 System Image	25	6	Update Available: 8
▼ <input type="checkbox"/>	Android 7.0 (Nougat)			
<input type="checkbox"/>	Google APIs	24	1	Not installed

Slika 2-5

Na prethodnoj slici istaknuta je dostupnost ažuriranja. Da biste instalirali ažuriranja, uključite polje za potvrđivanje levo od naziva stavke i kliknite na dugme Apply.

Osim Android SDK paketa, instaliran je i veliki broj alati za izgradnju Android aplikacija. Da biste pregledali instalirane pakete i proverili ažuriranja, ostanite na ekranu SDK podešavanja i selektujte karticu SDK Tools, kao što je prikazano na slici 2-6.



Slika 2-6

Na ekranu Android SDK Tools uverite se da su sledeći paketi izlistani kao Installed u koloni Status:

- Android SDK Build-tools
- Android Emulator
- Android SDK Platform-tools
- Android SDK Tools
- Google Play Services
- Instant Apps Development SDK
- Intel x86 Emulator Accelerator (HAXM installer)
- ConstraintLayout for Android
- Solver for ConstraintLayout
- Android Support Repository
- Google Repository
- Google USB Driver (Windows only)

Ukoliko je bilo koji od navedenih paketa izlistan kao Not Installed ili zahteva ažuriranje, jednostavno selektujte polja za potvrđivanje pored tih paketa i kliknite na dugme Apply da biste pokrenuli proces instalacije.

Kada je instalacija završena, pogledajte listu paketa i uverite se da su selektovani paketi izlistani kao *Installed* u koloni *Status*. Ako je neki izlistan kao Not installed, uverite se da je selektovan i ponovo kliknite na dugme Apply.

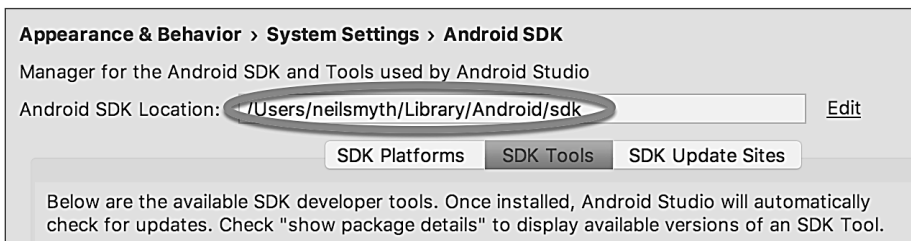
2.6 Učinite dostupnom komandnu liniju Android SDK alati

U većini slučajeva pozadinske alatke Android SDK-a će biti dostupne unutar Android Studio okruženja. Međutim, u nekim instancama će biti korisno da postoji mogućnost pozivanja tih alatki iz komandne linije ili prozora terminala. Da bi operativni sistem na kojem programirate mogao da pronađe ove alatke, potrebno je da ih dodate u PATH promenljivu okruženja sistema.

Bez obzira na operativni sistem, promenljiva PATH treba da bude konfigurisana da uključuje sledeće putanje (gde `<path_to_android_sdk_installation>` predstavlja lokaciju fajl sistema na koji je Android SDK instaliran):

```
<path_to_android_sdk_installation>/sdk/tools
<path_to_android_sdk_installation>/sdk/tools/bin
<path_to_android_sdk_installation>/sdk/platform-tools
```

Lokacija SDK-a na sistemu može da se identifikuje pokretanjem SDK Managera i pregledom polja Android SDK Location, koje se nalazi na vrhu panala sa podešavanjima, kao što je prikazano na slici 2-7.



Slika 2-7

Kada je identifikovana lokacija SDK-a, koraci za dodavanje lokacije u PATH promenljivu zavise od operativnog sistema.

2.6.1 Windows 7

1. Kliknite desnim tasterom miša na Computer na Start meniju desktopa i iz kontekstnog menija selektujte Properties.
2. Na panelu Properties selektujte link Advanced System Settings i u okviru za dijalog koji će se otvoriti kliknite na dugme Environment Variables....
3. U okviru za dijalog Environment Variables potražite promenljivu Path u listi System variables, selektujte je i kliknite na Edit.... Pronađite kraj znakovnog niza vrednosti aktuelne promenljive i dodajte putanju na kraj alatki Android platforme, koristeći znak tačka-zarez za razdvajanje putanje od prethodnih vrednosti. Na primer, ako pretpostavimo da je Android SDK instaliran u C:\Users\demo\AppData\Local\Android\sdk, na kraj aktuelne vrednosti Path će biti dodata sledeća putanja:

```
;C:\Users\demo\AppData\Local\Android\sdk\platform-tools; C:\Users\demo\AppData\Local\Android\sdk\tools; C:\Users\demo\AppData\Local\Android\sdk\tools\bin
```

4. Kliknite na OK u svakom okviru za dijalog i zatvorite kontrolni panel svojstava sistema.

Kada izvršite prethodne korake, potvrdite da je putanja pravilno podešena, tako što ćete otvoriti prozor Command Prompt (Start -> All Programs -> Accessories -> Command Prompt), a u odzivnik unesite:

```
echo %Path%
```

Vraćena vrednost promenljive path treba da uključuje putanje ka direktorijumima alatki Android SDK platforme. Potvrdite da je vrednost platform-tools tačna, tako što ćete pokrenuti adb alatku na sledeći način:

```
adb
```

Trebalo bi da alatka prikaže listu opcija komandne linije kada je izvršena.

Slično tome, proverite podešavanja putanje *tools*, tako što ćete pokrenuti alatku komandne linije AVD Manager:

```
avdmanager
```

Ukoliko je prikazana poruka slična sledećoj za jednu ili obe komande, verovatno je netačna putanja dodata u promenljivu okruženja Path:

```
'adb' is not recognized as an internal or external command,  
operable program or batch file.
```

2.6.2 Windows 8.1

1. Na početnom ekranu pomerite kursor u donji desni ugao ekrana i selektujte Search iz menija koji bude prikazan. U polje za pretragu unesite Control Panel. Kada je prikazana ikonica Control Panel u području rezultata, kliknite na nju da biste pokrenuli alatku na desktopu.
2. Unutar Control Panela upotrebite meni Category da biste promenili prikaz na Large Icons. Iz liste ikonica selektujte ikonicu System.
3. Pratite korake 2, 3 i 4 koji su opisani za Windows 7. Otvorite prozor komandne linije (pomerite kursor u donji desni ugao ekrana, selektujte opciju Search i unesite cmd u polje za pretragu). Selektujte *Command Prompt* iz rezultata pretrage.

Unutar Command Prompt prozora unesite:

```
echo %Path%
```

Vraćena vrednost promenljive Path treba da uključuje putanje ka direktorijumima alatki Android SDK platforme. Potvrdite da je vrednost *platform-tools* tačna, tako što ćete pokrenuti adb alatku na sledeći način:

```
adb
```

Alatka treba da ispiše listu opcija komandne linije kada je izvršena.

Slično tome, proverite podešavanja putanje *tools*, tako što ćete pokrenuti alatku komandne linije AVD Manager:

```
avdmanager
```

Ukoliko bude prikazana poruka slična sledećoj za jednu ili obe komande, najverovatnije je netačna putanja dodata u promenljivu okruženja Path:

```
'adb' is not recognized as an internal or external command,  
operable program or batch file.
```

2.6.3 Windows 10

Kliknite desnim tasterom miša na Start meni, selektujte System iz kontekstnog menija i kliknite na opciju *Advanced system settings* u prozoru System. Pratite korake 2, 3 i 4 koji su opisani za Windows 7.

2.6.4 Linux

Na Linuxu ova konfiguracija može da se podesi dodavanjem komande u fajl `.bashrc` u home direktorijumu (specifikacije se mogu razlikovati, u zavisnosti od određene Linux distribucije koja se koristi). Pod pretpostavkom da je Android SDK paket instaliran u direktorijum `/home/demo/Android/sdk`, linija eksportovanja u fajl `.bashrc` izgleda ovako:

```
export PATH=/home/demo/Android/sdk/platform-tools:/home/demo/Android/sdk/tools:/home/demo/Android/sdk/tools/bin:/home/demo/android-studio/bin:$PATH
```

Takođe imajte na umu da prethodna komanda dodaje direktorijum `android-studio/bin` u promenljivu `PATH`. Na taj način je omogućeno izvršenje skripta `studio.sh`, bez obzira na aktuelni direktorijum unutar prozora terminala.

2.6.5 macOS

Za modifikovanje promenljive okruženja `$PATH` na macOS sistemu može da se primeni veći broj tehnika. Nesumnjivo najčistiji metod je dodavanje novog fajla u direktorijum `/etc/paths.d`, koji sadrži putanje koje će biti dodate u promenljivu `$PATH`. Pod pretpostavkom da je lokacija instalacije Android SDK-a direktorijum `/Users/demo/Library/Android/sdk`, putanja može da bude konfigurisana kreiranjem novog fajla, pod nazivom `android-sdk`, u direktorijumu `/etc/paths.d`, koji sadrži sledeće linije:

```
/Users/demo/Library/Android/sdk/tools
/Users/demo/Library/Android/sdk/tools/bin
/Users/demo/Library/Android/sdk/platform-tools
```

Pošto je ovo sistemski direktorijum, potrebno je da upotrebite komandu `sudo` kada kreirate fajl - na primer:

```
sudo vi /etc/paths.d/android-sdk
```

2.7 Ažuriranje Android Studioa i SDK-a

S vremena na vreme se izdaju nove verzije Android Studioa i Android SDK-a. Nove verzije SDK-a su instalirane pomoću Android SDK Managera. Android Studio će vas, obično, obavestiti kada je spremno ažuriranje za instalaciju.

Da biste ručno proverili ažuriranja za Android Studio, kliknite na *Configure -> Check for Update* opciju unutar Android Studio welcome ekrana ili upotrebite opciju *Help -> Check for Update*, koja je dostupna unutar glavnog prozora Android Studioa.

2.8 Rezime

Pre nego što počnete da programirate aplikacije zasnovane na Androidu, prvo treba da podesite odgovarajuće razvojno okruženje. Ono se sastoji od Java Development Kita (JDK), Android SDK-a i Android Studio IDE-a. U ovom poglavlju smo opisali korake potrebne za instaliranje tih paketa na Windows, macOS i Linux.



KREIRANJE PRIMERA ANDROID APLIKACIJE U ANDROID STUDIOU

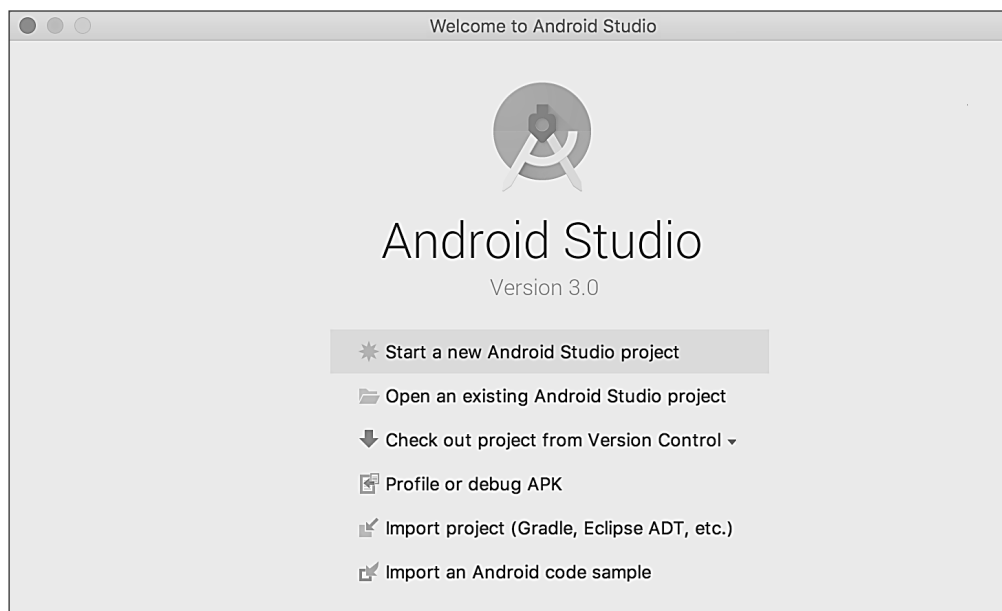
U prethodnim poglavljima ove knjige opisani su koraci za konfigurisanje okruženja koje je pogodno za Android aplikacije pomoću Android Studio IDE-a. Pre nego što nastavite učenje malo naprednijih tema, sada je dobro vreme da utvrdite da li su instalirani svi potrebni paketi i da li funkcionišu pravilno. Najbolji način da to postignete je da Android aplikaciju kreirate, kompajlirate i pokrenete. U ovom poglavlju ćemo opisati kreiranje jednostavnog projekta Android aplikacije pomoću Android Studioa. Kada je projekat kreiran, u sledećem poglavlju ćemo upotrebiti okruženje Android simulatora da bismo izvršili test pokretanja aplikacije.

3.1 O projektu

Projekat koji je kreiran u ovom poglavlju ima formu veoma jednostavnog kalkulatora za konverziju valuta (konvertuje samo dolare u evre i to radi procenom stope konverzije). Za projekat ćemo upotrebiti najosnovnije šablone Android Studio projekta. Ova jednostavnost nam omogućava da prikazemo neke ključne aspekte razvoja Android aplikacije, bez preteranog opterećenja početnika u pokušajima da predstavimo previše koncepata, kao što su preporučena arhitektura aplikacije i komponente Android arhitekture odjednom. Kada pratite uputstva iz ovog poglavlja, znajte da će sve tehnike i kod koji je upotrebljen u početnom primeru projekta biti opisani mnogo detaljnije u narednim poglavljima.

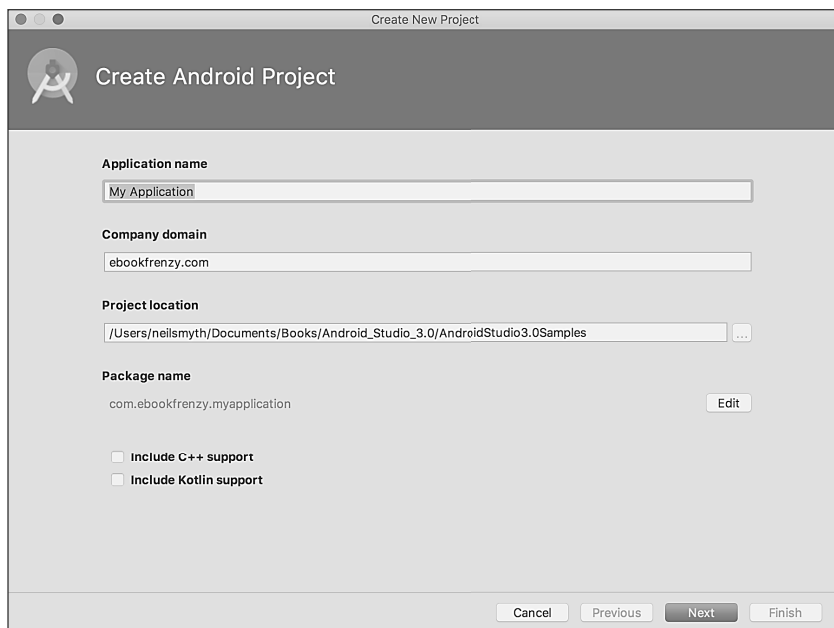
3.2 Kreiranje novog Android projekta

Prvi korak u procesu razvoja aplikacije je da kreirate novi projekat unutar Android Studio okruženja. Prema tome, započnite projekat otvaranjem Android Studioa, tako da se prikaže ekran „Welcome to Android Studio“, kao što je ilustrovano na slici 3-1.



Slika 3-1

Kada se otvori ovaj prozor, Android Studio je spreman i novi projekat može da bude kreiran. Radi kreiranja novog projekta, jednostavno kliknite na opciju **Start a new Android Studio project** da biste prikazali prvi ekran New Project wizarda, kao što je prikazano na slici 3-2.



Slika 3-2

3.3 Definisanje projekta i SDK podešavanja

U prozoru New Project podesite polje Application name na Android Sample. Naziv aplikacije je naziv po kojem će aplikacija biti referencirana i identifikovana unutar Android Studioa; osim toga, to je naziv koji bi se koristio ako bi završena aplikacija bila namenjena prodaji u Google Play prodavnici.

Polje Package Name se koristi za jedinstvenu identifikaciju aplikacije unutar ekosistema Android aplikacije. Iako može da se podesi na bilo koji znakovni niz, koji jedinstveno identifikuje aplikaciju, obično se koristi rezervisani URL naziva domena iza kojeg se nalazi naziv aplikacije. Na primer, ako je domen `www.mycompany.com`, a naziv aplikacije je `AndroidSample`, naziv paketa može da se specifikuje kao

```
com.mycompany.androidsample.
```

Ako nemate naziv domena, možete da unesete bilo koji drugi znakovni niz u polje Company Domain ili možete da upotrebite `example.com` za testiranje, mada će biti potrebno da, pre nego što aplikacija bude publikovana, taj naziv promenite u

```
com.example.androidsample
```

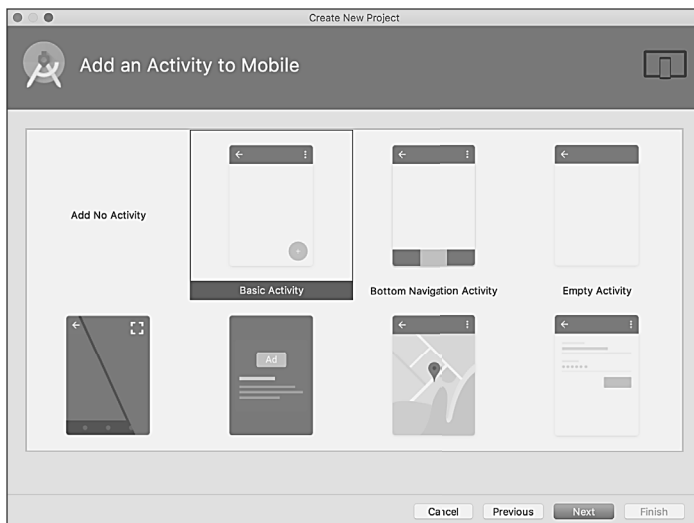
Podešavanje Project location će biti standardno podešavanje za lokaciju u direktorijumu `AndroidStudio-Projects`, koji se nalazi u home direktorijumu i može da bude promenjen ako kliknete na dugme desno od tekstualnog polja koje sadrži aktuelnu putanju.

Na kraju, uključite opciju `Include Kotlin support`.

Kliknite na `Next` da biste nastavili. Na ekranu form factors uključite opciju `Phone and Tablet` i podesite minimalno SDK podešavanje za API 26: Android 8.0 (Oreo). Ovo je SDK koji će biti upotrebljen u većini projekata kreiranih u ovoj knjizi, osim ako je potrebna funkcija dostupna samo u najnovijoj verziji. Iako Android Studio omogućava da budu selektovane starije verzije SDK-a, mnogo bezbednosnih funkcija i funkcija privatnosti ugrađenih u Android predstavljeno je tek nakon što je izdat API 25 SDK. Da bi bila poboljšana bezbednost aplikacije, Google Play prodavnica će, kako je „Google“ objavio, ove godine prihvatati samo aplikacije građene u API-ju 26 ili novijim. Ista ograničenja će se primenjivati i na ažuriranja postojećih aplikacija nakon oktobra ove godine. Pošto projekat nije namenjen za Google TV, Android Auto ili prenosive uređaje, ostavite opcije isključene pre nego što kliknete na `Next`. Podešavanje Instant Apps biće opisano kasnije u ovoj knjizi, pa se uverite da je isključena opcija `Include Android Instant App support`.

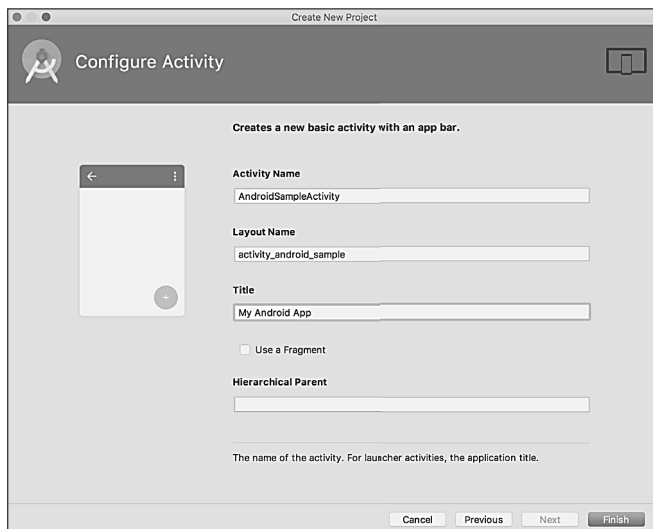
3.4 Kreiranje aktivnosti

Sledeći korak je da definišete tip početne aktivnosti koja će biti kreirana za aplikaciju. Raspon različitih tipova aktivnosti je dostupan kada razvijate Android aplikacije, od kojih će mnoge biti opisane detaljnije u narednim poglavljima. Za ovaj primer, međutim, jednostavno selektujte opciju za kreiranje `Basic Activity`. Ta opcija kreira šablon korisničkog interfejsa koji se sastoji od paleta aplikacije, menija, područja sadržaja i jednog plutajućeg dugmeta akcije.



Slika 3-3

Kada je selektovana opcija Basic Activity, kliknite na Next. Na posljednjem ekranu (slika 3-4) unesite za naziv aktivnosti naslov `AndroidSampleActivity`. Aktivnost će se sastojati od jednog rasporeda ekrana korisničkog interfejsa - za ovaj primer mu dodelite naziv `activity_android_sample`. Na kraju, unesite *My Android App* u polje Title, kao što je prikazano na slici 3-4.

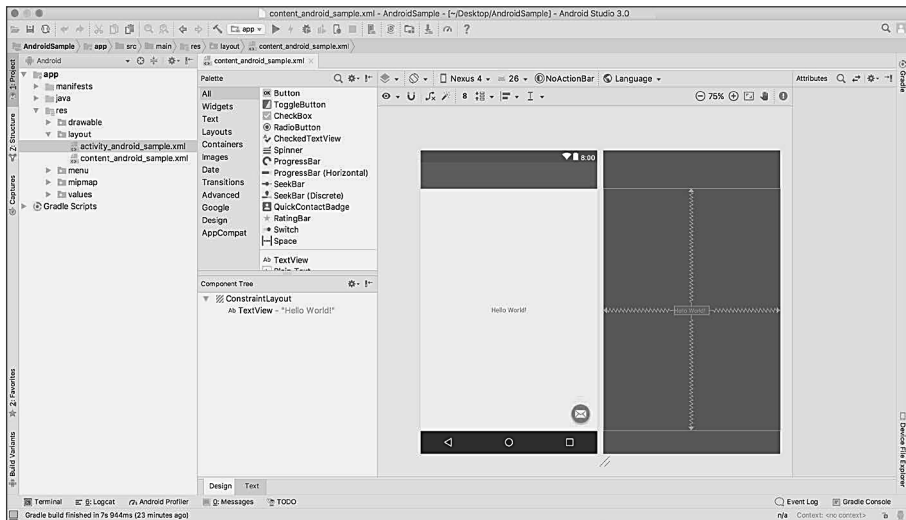


Slika 3-4

Kliknite na Finish da biste pokrenuli proces kreiranja projekta.

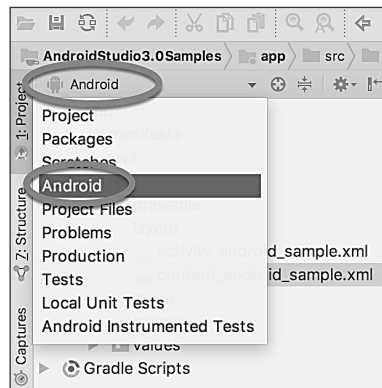
3.5 Modifikovanje primera aplikacije

Sada je Android Studio kreirao minimalni primer projekta aplikacije i otvoren je glavni prozor.



Slika 3-5

Novokreirani projekat i reference sa povezanim fajlovima su izlistani u prozoru alatke Project, koji se nalazi na levoj strani glavnog prozora projekta. Prozor alatke Project ima veliki broj režima u kojima informacije mogu da budu prikazane. Prema standardnom podešavanju, ovaj panel će biti u *Android* režimu. Ovo podešavanje se kontroliše menijem na vrhu panela, kao što je istaknuto na slici 3-6. Ako se panel trenutno ne nalazi u režimu Android, upotrebite meni da biste promenili režim.

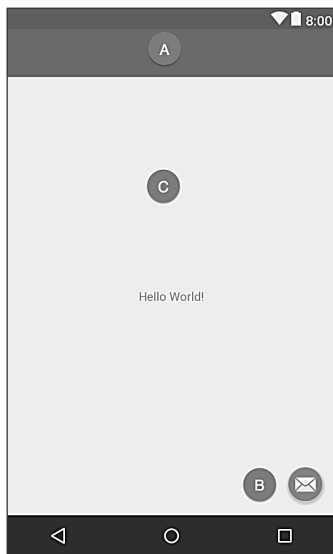


Slika 3-6

Primer projekta kreiran je kada selektujete opciju za kreiranje aktivnosti koja se sastoji od korisničkog interfejsa sa oznakom „Hello World!“ kada je aplikacija izvršena.

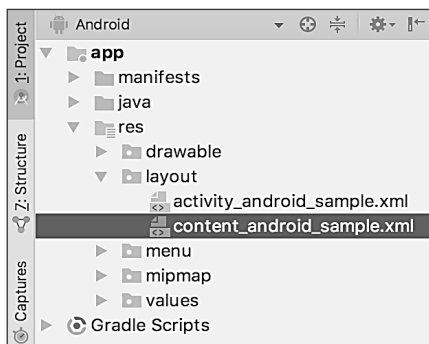
Sledeći korak u ovom uputstvu je da modifikujete korisnički interfejs aplikacije da prikazuje veći objekat prikaza teksta sa porukom drugačijom od one koju prikazuje Android Studio.

Dizajn korisničkog interfejsa za aktivnost je uskladišten u fajlu *activity_android_sample.xml*, koji se nalazi u hijerarhiji fajlova projekta u direktorijumu *app -> res -> layout*. Ovaj fajl rasporeda uključuje paletu aplikacije (poznatu i kao paleta akcija) koja se prikazuje duž vrha ekrana uređaja (označena je sa A na slici 3-7) i plutajuće dugme akcije (e-mail dugme označeno je sa B). Osim ovih stavki, fajl rasporeda *activity_android_sample.xml* sadrži referencu za drugi fajl u kome se nalazi sadržaj rasporeda (označen sa C).



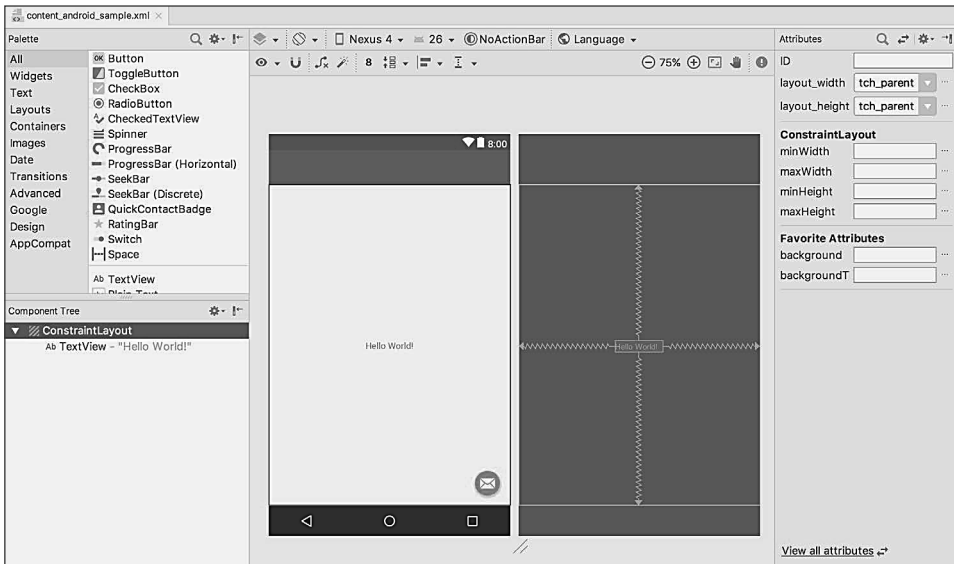
Slika 3-7

Prema standardnom podešavanju, raspored sadržaja je unutar fajla *content_android_sample.xml*, a on se nalazi unutar ovog fajla koji se menja u kreiranom rasporedu aktivnosti. Koristeći prozor alatke Project, locirajte ovaj fajl kao što je ilustrovano na slici 3-8.




Slika 3-8

Kada je fajl lociran, dvostruko kliknite na njega da biste ga učitali u alatku Layout Editor korisničkog interfejsa, koja će biti prikazana u centru panela glavnog prozora Android Studioa.

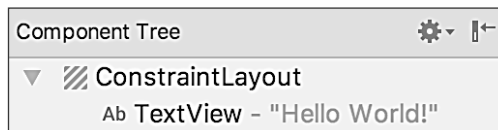


Slika 3-9

U paleti sa alatima na vrhu prozora Layout Editor nalazi se meni (podešen na Nexus 4 na gornjoj slici), koji je reflektovan u vizuelnoj reprezentaciji uređaja unutar panela Layout Editor. Širok raspon opcija drugih uređaja dostupan je za selekciju kada kliknete na ovaj meni.

Da biste promenili orijentaciju prikaza uređaja između pejzažne i portretne, jednostavno upotrebite padajuću meni koji se nalazi odmah levo od menija za selekciju uređaja; prikazana je  ikonica.

Kao što može da se vidi na ekranu uređaja, raspored sadržaja već uključuje naslov koji prikazuje „Hello World!“ poruku. Na levoj strani panela nalazi se paleta koja sadrži različite kategorije komponentata korisničkog interfejsa, koje mogu da se upotrebe za konstruisanje korisničkog interfejsa, kao što su dugmad, oznake i tekstualna polja. Međutim, nisu sve komponente korisničkog interfejsa vidljive korisniku. Jedna takva kategorija se sastoji od rasporeda. Android podržava različite rasporede koji obezbeđuju različite nivoe kontrole za način pozicioniranja i upravljanja vidljivim komponentama korisničkog interfejsa. Iako je teško na osnovu pregleda vizuelne reprezentacije korisničkog interfejsa reći, aktuelni dizajn je kreiran pomoću rasporeda *ConstraintLayout*. To možemo da potvrdimo ako pregledamo informacije u panelu *Component Tree*, koji se, prema standardnom podešavanju, nalazi u donjem levom uglu panela Layout Editor i prikazan je na slici 3-10.



Slika 3-10

Kao što možemo da vidimo iz hijerarhije stabla komponente, raspored korisničkog interfejsa se sastoji od *ConstraintLayout* „roditelja“ sa jednim „potomkom“ u formi objekta *TextView*.

Pre nego što nastavite rad, potvrdite da je režim Autoconnect Layout Editora uključen. To znači da će, dok se komponente dodaju u raspored, Layout Editor automatski dodati ograničenja da bi bio siguran

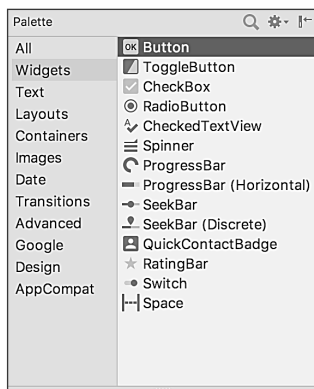
da su komponente tačno pozicionirane za različite veličine ekrana i orijentaciju uređaja (ova tema će biti detaljnije opisana u narednim poglavljima). Dugme Autoconnect se prikazuje kao ikonica magneta u paleti sa alatkama Layout Editor. Kada je opcija isključena, magnet će biti prevučen dijagonalnom linijom (slika 3-11). Ako je potrebno, ponovo uključite režim Autoconnect, tako što ćete kliknuti na ovo dugme.



Slika 3-11

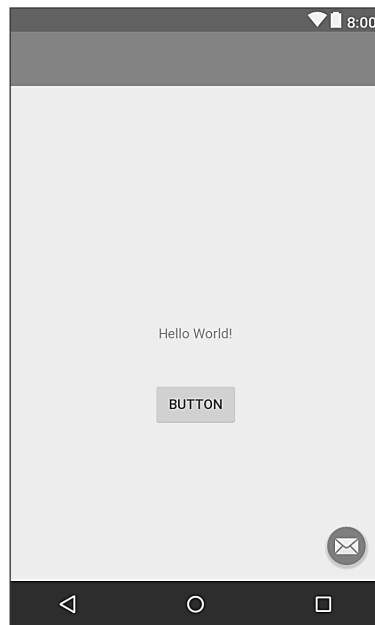
Sledeći korak u modifikovanju aplikacije je da dodate komponente u raspored; prva će biti dugme koje korisnik treba da pritisne da bi pokrenuo konverziju valute.

Panel Palette se sastoji od dve kolone. Leva kolona sadrži listu kategorija komponenata prikaza, a desna lista komponente koje se nalaze unutar aktuelno selektovane kategorije. Na slici 3-12 prikaz Button je aktuelno selektovan unutar kategorije Buttons.



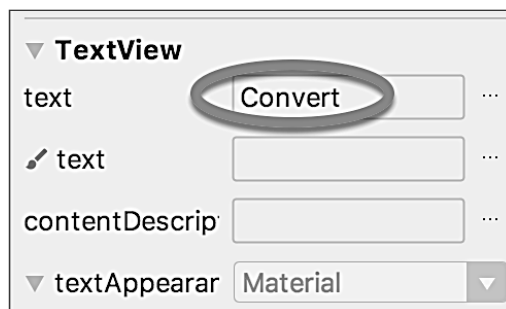
Slika 3-12

Kliknite i prevucite objekat Button iz liste Buttons i otpustite ga u horizontalni centar korisničkog interfejsa, tako da bude pozicioniran ispod postojećeg vidžeta TextView.



Slika 3-13

Sledeći korak je da promenite tekst koji komponenta Button prikazuje. Na desnoj strani područja dizajna nalazi se panel Attributes. On prikazuje atribute dodeljene aktuelno selektovanoj komponenti u rasporedu. Unutar ovog panela potražite svojstvo text i promenite aktuelnu vrednost sa „Button“ na „Convert“, kao što je prikazano na slici 3-14.

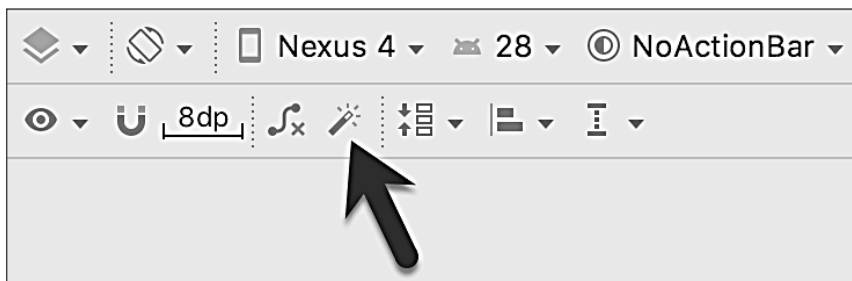


Slika 3-14

Da biste promenili svojstvo text za komponentu, korisna prečica je da na nju dvostruko kliknete. Automatski će u panelu Attributes biti prikazan atribut, selektovan i spreman za editovanje.

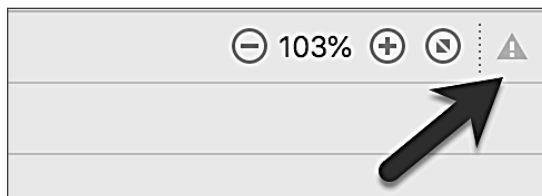
Drugo svojstvo text, pored kojeg se nalazi ikonica sa ključem, omogućava da svojstvo text bude podešeno, što će biti prikazano samo unutar alatke Layout Editor, ali ne u toku izvršenja. To je korisno za testiranje načina na koji će se vizuelna komponenta i raspored ponašati sa različitim podešavanjima, bez potrebe da se aplikacija ponovo pokreće.

U slučaju da je sistem Autoconnect neuspešan prilikom podešavanja svih konekcija rasporeda, kliknite na dugme Infer constraints (slika 3-15) da biste dodali nedostajuća ograničenja u raspored.



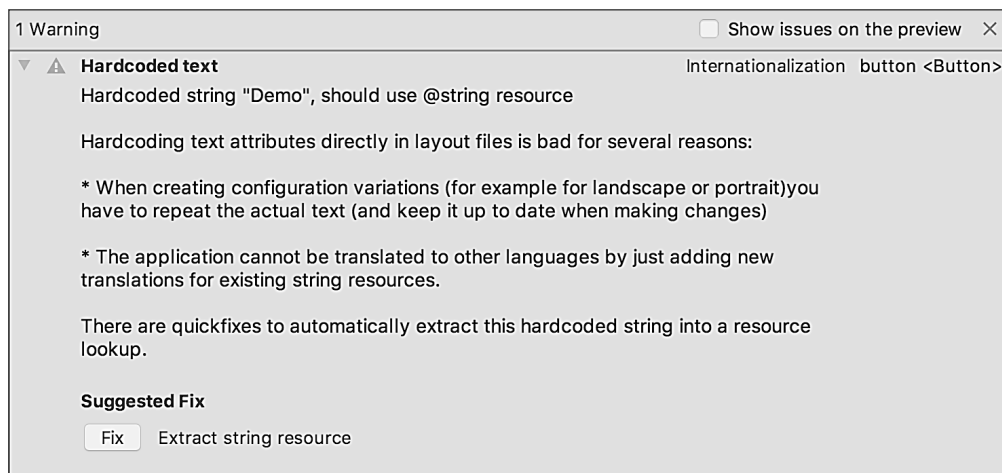
Slika 3-15

Sada je važno da objasnim dugme upozorenja koje se nalazi u gornjem desnom uglu alatke Layout Editor, kao što je prikazano na slici 3-16. Ono ukazuje na potencijalne probleme u rasporedu. Za detalje o problemima kliknite na to dugme.



Slika 3-16

Kada je dugme kliknuto, biće prikazan panel (slika 3-17) koji opisuje prirodu problema i nudi neke moguće korektivne mere.



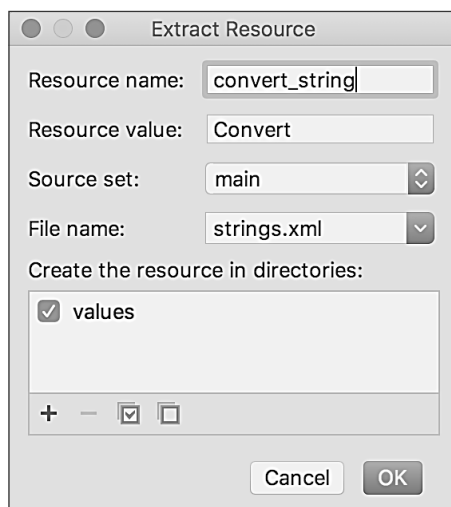
Slika 3-17

Trenutno, jedino izlistano upozorenje je sledeće:

Hardcoded string "Convert", should use @string resource

Ova I18N poruka informiše nas da postoji potencijalni problem koji se odnosi na buduću internacionalizaciju projekta (I18N je naziv koji je nastao na osnovu reči internationalization, koja počinje sa I, a završava se sa N i ima 18 slova između). Upozorenje nas podseća da, kada razvijamo Android aplikacije, atributi i vrednosti, kao što su tekstualni nizovi, treba da budu sačuvani u formi resursa kad god je to moguće, čime se omogućava da se promene izgleda aplikacije izvršavaju modifikovanjem fajlova resursa, umesto menjanjem izvornog koda aplikacije. To može posebno biti vredno kada prevodimo korisnički interfejs na drugačiji jezik. Ako se ceo tekst u korisničkom interfejsu nalazi u jednom fajlu resursa, fajl može da se prosledi prevodiocu koji će izvršiti prevod i vratiti preveden fajl za uključivanje u aplikaciju. Na taj način može da se upotrebi više jezika bez potrebe da se izvršava promena izvornog koda. U ovom primeru mi ćemo kreirati novi resurs, pod nazivom *demonstring*, i dodeliti mu znakovni niz „Demo“.

Kliknite na dugme Fix na panelu Issue Explanation da biste prikazali panel *Extract Resource* (slika 3-18). Unutar ovog panela promenite polje Resource name na *convert_string*, a polje Resource value ostavite podešeno na Convert, pre nego što kliknete na dugme OK.



Slika 3-18

I znakovni niz može da se dodeli resursu kada je unet u panel Attributes. To uključuje klik na dugme koje prikazuje tri tačke desno od polja svojstva u panelu Attributes i selektovanje opcije *Add new resource -> New String Value...* u okviru za dijalog Resources, koji će se otvoriti. Međutim, u praksi, često je brže jednostavno podesiti vrednosti direktno u poljima panela Attributes za svaki vidžet u rasporedu, nego izvršavati zadatak sekvencijalno kroz listu u okviru za dijalog sa upozorenjima za ekstrakovanje potrebnih resursa kada je raspored završen.

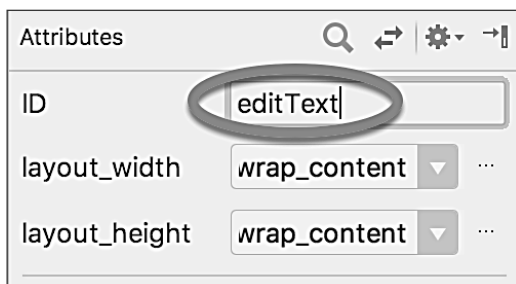
Sledeći vidžet koji ćemo dodati je EditText, u koji će korisnik uneti vrednost dolara koja će biti konvertovana. Iz palete vidžeta selektujte kategoriju Text i kliknite i prevucite komponentu Number (Decimal) na raspored da bude centrirana horizontalno i pozicionirana iznad postojećeg vidžeta TextView. Kada je vidžet selektovan, upotrebite prozor alatki Attributes da biste podesili svojstvo *hint* na „dollars“.

Dodajte nedostajuća ograničenja rasporeda, tako što ćete kliknuti na dugme *Infer constraints*. Sada raspored treba da izgleda kao na slici 3-19.



Slika 3-19

Kod napisan kasnije u ovom poglavlju treba da pristupi vrednosti dollar koju unese korisnik u polje EditText. Kod će to izvršiti referenciranjem ID-a koji je dodeljen vidžetu u rasporedu korisničkog interfejsa. Standardni ID koji vidžetu dodeljuje Android Studio može da se prikaže i promeni unutar prozora Attributes kada je vidžet selektovan u rasporedu, kao što je prikazano na slici 3-20.

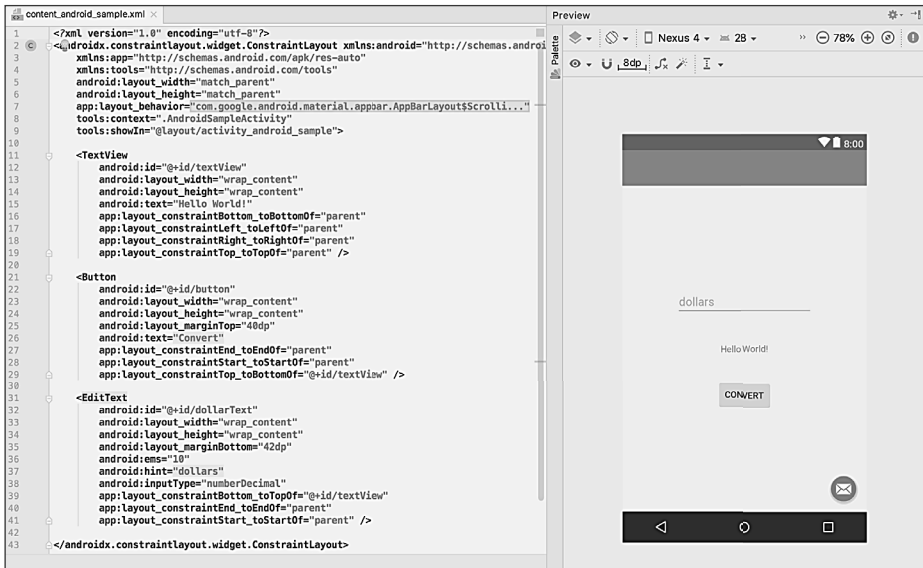


Slika 3-20

Promenite ID na *dollarText*, pre nego što nastavite rad.

3.6 Pregled rasporeda i fajlova resursa

Pre nego što preduzmete sledeći korak, pogledajte neke interne aspekte dizajna korisničkog interfejsa i obrade resursa. U prethodnom odeljku smo izvršili neke promene u korisničkom interfejsu modifikovanjem fajla *content_android_sample.xml* pomoću alatke Layout Editor. U stvari, alatka Layout Editor samo obezbeđuje jasan način za editovanje pozadinskog XML sadržaja fajla. U praksi, ne postoji razlog zašto ne biste mogli da modifikujete XML direktno da biste izvršili promene korisničkog interfejsa; u nekim primerima to je možda brže nego upotreba alatke Layout Editor. Na dnu panela Layout Editor nalaze se dve kartice, označene sa *Design* i *Text*. Da biste se prebacili na XML prikaz, jednostavno selektujte karticu Text, kao što je prikazano na slici 3-21.



Slika 3-21

Kao što možete da vidite iz strukture XML fajla, korisnički interfejs se sastoji od komponente *ConstraintLayout*, koja je „roditelj“ objekta *Button*. Takođe možete da uočite da je svojstvo *text* za objekat *Button* podešeno na demostring resurs. Iako se razlikuju po složenosti i sadržaju, svi rasporedi korisničkog interfejsa su strukturirani na ovaj hijerarhijski način zasnovan na XML-u.

Jedna od najmoćnijih funkcija Android Studioa može da se pronade sa desne strane panela XML editing. Ako panel nije vidljiv, prikažite ga, tako što ćete selektovati dugme *Preview*, koje se nalazi duž desne ivice prozora Android Studioa. To je panel *Preview*, koji prikazuje aktuelno vizuelno stanje rasporeda. Kada su izvršene promene u XML rasporedu, one će biti reflektovane u panelu *Preview*. Raspored takođe može da bude modifikovan vizuelno unutar tog panela, a promene će biti prikazane u XML listi. Da biste videli ovo u akciji, modifikujte XML raspored, tako što ćete promeniti boju pozadine komponente *ConstraintLayout* na nijansu crvene na sledeći način:

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://
schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
```

```

    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context=".AndroidSampleActivity"
    tools:showIn="@layout/activity_android_sample"
    android:background="#ff2438" >
.
.
</android.support.constraint.ConstraintLayout>

```

Vidite da se boja u prikazu menja u realnom vremenu da bi se poklopila sa novim podešavanjem u XML fajlu. Takođe vidite da se prikazuje mali crveni kvadrat u levoj margini (naziva se i međustubačni razmak) XML editora pored linije koja sadrži podešavanje boje. To je vizuelni trag za činjenicu da je crvena boja podešena u svojstvu. Promenite vrednost boje na #a0□28 i videćete da će i mali kvadrat u margini i prikaz postati zeleni.

Na kraju, upotrebite prikaz Project da biste locirali fajl *app -> res -> values -> strings.xml* i dvostruko kliknite na njega da biste ga učitali u editor. Trenutno, XML izgleda ovako:

```

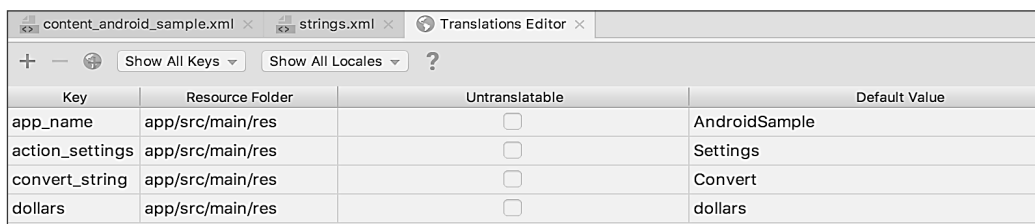
<resources>
    <string name="app_name">AndroidSample</string>
    <string name="action_settings">Settings</string>
    <string name="convert_string">Convert</string>
    <string name="dollars">dollars</string>
</resources>

```

Kao demonstraciju resursa u akciji, promenite vrednost znakovnog niza koji je trenutno dodeljen resursu demostring na „Convert to Euros“, a zatim se vratite u prozor Layout Editor, tako što ćete selektovati karticu za fajl rasporeda u panelu editora. Vidite da je raspored preuzeo novu izvornu vrednost za znakovni niz.

Postoji i brži način za pristup vrednosti resursa referenciranog u XML fajlu. Pomoću alatke Layout Editor u režimu Text kliknite na podešavanje svojstva *@string/convert_string*, tako da ono bude istaknuto, a zatim pritisnite Ctrl-B na tastaturi (Cmd-B na macOS-u). Android Studio će otvoriti fajl *strings.xml* i u njemu prikazati liniju u kojoj je deklarisan resurs. Upotrebite ovu mogućnost za preuzimanje resursa znakovnog niza nazad u originalni tekst „Demo“.

Znakovni nizovi resursa takođe mogu da budu editovani pomoću Android Studio Translations Editora. Da biste otvorili ovaj editor, kliknite desnim tasterom miša na fajl *app -> res -> values -> strings.xml* i selektujte opciju Open Editor. U glavnom panelu prozora Android Studioa biće prikazan Translation Editor.



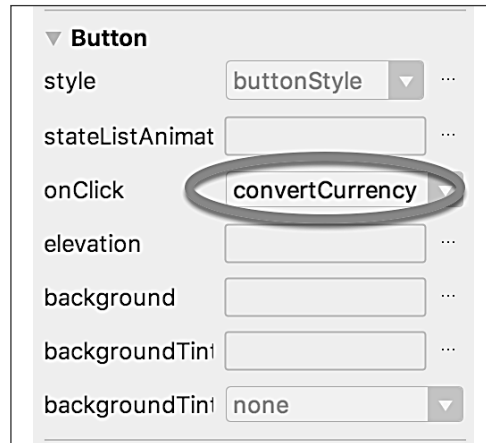
Key	Resource Folder	Untranslatable	Default Value
app_name	app/src/main/res	<input type="checkbox"/>	AndroidSample
action_settings	app/src/main/res	<input type="checkbox"/>	Settings
convert_string	app/src/main/res	<input type="checkbox"/>	Convert
dollars	app/src/main/res	<input type="checkbox"/>	dollars

Slika 3-22

Ovaj editor omogućava da znakovni nizovi koji su dodeljeni ključevima resursa budu editovani i da se njima upravlja za prevod na više jezika.

3.7 Dodavanje interakcije

Finalni korak u ovom primeru projekta je da učinimo aplikaciju interaktivnom, pa da se, kada korisnik unese vrednost dolara u polje EditText i klikne na dugme za konvertovanje, konvertovana vrednost eura prikazuje u polju TextView. To uključuje implementaciju obrade nekog događaja u vidžetu Button. Konkretno, objekat Button treba da bude konfigurisan tako da je pozvan metod u kodu aplikacije kada je pokrenut događaj onClick. Obrada događaja može da bude implementirana na više različitih načina, a opisana je detaljno u narednom poglavlju. Vratite editor rasporeda u režim Design, selektujte vidžet Button u editoru rasporeda, pogledajte prozor alatke Attributes i specifikujte metod pod nazivom convertCurrency, kao što je prikazano na sledećoj slici.



Slika 3-23

Zatim, dvostruko kliknite na fajl AndroidSampleActivity.kt da biste ga učitali u editor koda i dodajte kod za metod convertCurrency u fajl klase, tako da izgleda kao što je ovde prikazano; imajte na umu da je potrebno da importujete i neke dodatne Android pakete:

```
package com.ebookfrenzy.androidsample

import android.os.Bundle
import com.google.android.material.snackbar.Snackbar
import android.support.v7.app.AppCompatActivity;
import android.view.Menu
import android.view.MenuItem

import kotlinx.android.synthetic.main.activity_android_sample.*
import kotlinx.android.synthetic.main.content_android_sample.*

class AndroidSampleActivity : AppCompatActivity() {
    .
    .
    fun convertCurrency(view: View) {
```

```
if (dollarText.text.isEmpty()) {  
  
    val dollarValue = dollarText.text.toString().toFloat()  
  
    val euroValue = dollarValue * 0.85f  
  
    resultText.text = euroValue.toString()  
} else {  
    resultText.text = "No Value"  
}  
}  
  
.  
.  
}
```

Metod započinje proverom svojstva text prikaza dollarText EditText da bi se uverio da nije prazno (drugim rečima, da je korisnik uneo vrednost dolara). Ako vrednost nije uneta, u prikazu resultText biće prikazan znakovni niz „No Value“. Ako je vrednost dolara uneta, konvertovana je u vrednost sa pokretnim zarezom i izračunata je ekvivalentna vrednost eura. Ova vrednost pokretnog zareza je konvertovana u znakovni niz i predstavljena je u prikazu resultText. Ako vam je nešto od ovoga nejasno, ne brinite, jer će ovi koncepti biti opisani mnogo detaljnije u narednim poglavljima.

Projekat je sada završen i spreman za pokretanje, a zadatak će biti izvršen u sledećem poglavlju, nakon što kreiramo sesiju AVD simulatora za namenu testiranja.

3.8 Rezime

Veliki broj koraka je uključen u podešavanje Android razvojnog okruženja. Kada ih izvršite, uradite primer projekta da biste bili sigurni da je okruženje pravilno instalirano i konfigurisano. U ovom poglavlju smo kreirali jednostavnu aplikaciju, a zatim smo upotreбили Android Studio Layout Editor alatku za modifikovanje rasporeda korisničkog interfejsa. Na taj način smo istražili važnost upotrebe resursa, posebno u slučaju vrednosti znakovnog niza, i ukratko smo predstavili temu rasporeda. Zatim smo pregledali pozadinski XML koji se koristi za skladištenje dizajna korisničkog interfejsa Android aplikacija.

Iako je korisno imati mogućnost pregleda rasporeda unutar Android Studio Layout Editor alatke, ne postoji zamena za testiranje aplikacije kompajliranjem i pokretanjem.

Na kraju, dodat je događaj onClick u objekat Button, koji je povezan sa metodom implementiranim za ekstrakovanje korisničkog unosa iz komponente EditText konvertovanjem vrednosti iz dolara u euro, a zatim prikazivanjem rezultata u polju TextView.

Sada je aplikacija spremna za testiranje - potrebni koraci za podešavanje simulatora za testiranje će biti opisani detaljno u sledećem poglavlju.



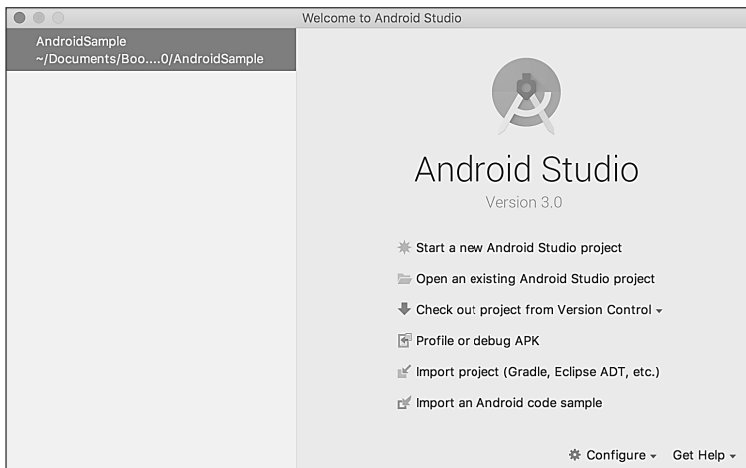
PREGLED KORISNIČKOG INTERFEJSA RAZVOJNOG OKRUŽENJA ANDROID STUDIO

Iako je primamljivo „zaroniti“ u primenu primera aplikacije koju smo kreirali u prethodnom poglavlju, to uključuje aspekte korisničkog interfejsa Android Studioa koje je najbolje opisati unapred.

Android Studio je moćno i funkcijama bogato razvojno okruženje koje je u velikoj meri intuitivno za upotrebu. Sada ćete upoznati raspored i organizaciju korisničkog interfejsa Android Studioa, što će znatno smanjiti potrebu učenja u narednim poglavljima knjige. Imajući to na umu, u ovom poglavlju ćemo pregledati različite oblasti i komponente koje čine Android Studio okruženje.

4.1 Ekran dobrodošlice

Ekran dobrodošlice (slika 4-1) prikazan je uvek kada je otvoren Android Studio bez otvorenog projekta (otvoreni projekti mogu biti zatvoreni bilo kada selektovanjem opcije menija File -> Close Project). Ako je Android Studio prethodno bio zatvoren, dok je projekat bio i dalje otvoren, alatka će zaobići ekran dobrodošlice kada se otvori sledećeg puta i automatski će otvoriti prethodno aktivan projekat.



Slika 4-1

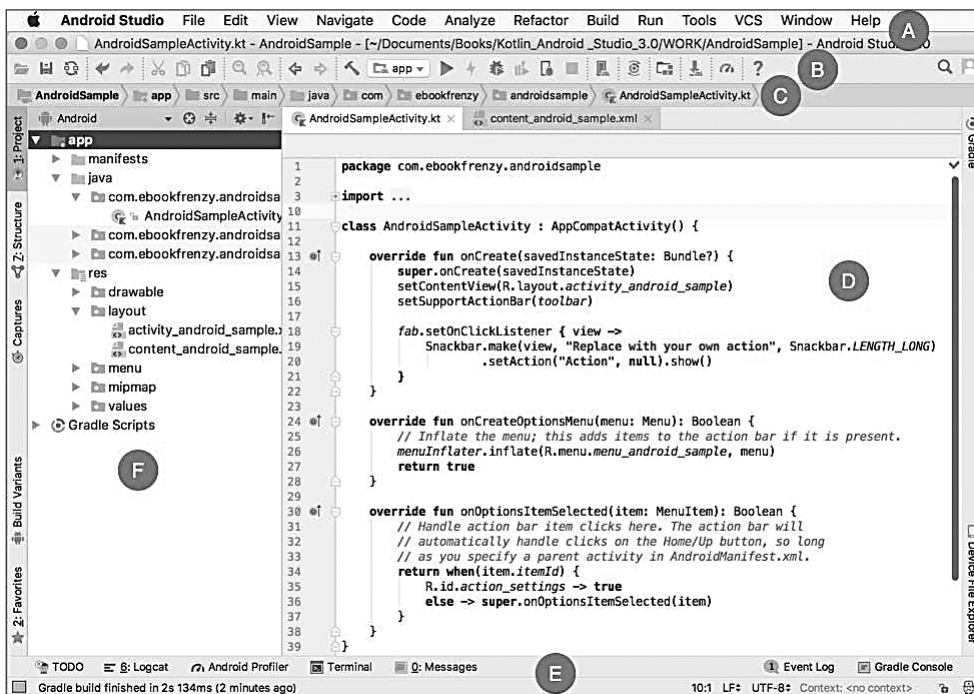
Osim liste nedavno otvorenih projekata, meni Quick Start obezbeđuje raspon opcija za izvršavanje zadataka, kao što su otvaranje, kreiranje i importovanje projekata, zajedno sa pristupom projektima

koji su pod kontrolnom verzijom. Osim toga, meni *Configure* na dnu prozora obezbeđuje pristup SKD Manageru, zajedno sa ogromnim nizom podešavanja i opcija konfiguracije. Pregled ovih opcija će brzo otkriti da ne postoji aspekt Android Studioa koji ne može da bude konfigurisan i „skrojen“ za specifične potrebe.

Meni *Configure* takođe uključuje opciju za proveru da li je dostupno ažuriranje Android Studioa za preuzimanje.

4.2 Glavni prozor

Kada je kreiran novi projekat ili je otvoren postojeći, biće prikazan glavni prozor Android Studioa. Kada je istovremeno otvoreno više projekata, svakom će biti dodeljen poseban glavni prozor. Precizna konfiguracija prozora će varirati, u zavisnosti od toga koje su alatke i paneli prikazani kada je poslednjeg puta projekat bio otvoren, ali će podsećati na ekran sa slike 4-2.



Slika 4-2

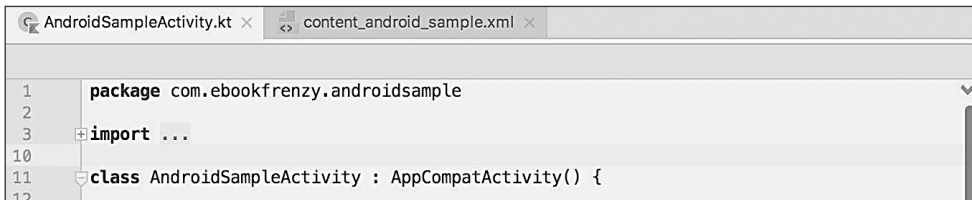
Različiti elementi glavnog prozora mogu biti rezimirani na sledeći način:

A – linija menija – Sadrži različite menije za izvršavanje zadataka unutar Android Studio okruženja.

B – paleta sa alatkama (selekcija prečica za često izvršavane zadatke) – Dugmad palete sa alatkama obezbeđuju brži pristup za selektovanu grupu akcija linije menija. Paleta sa alatkama može da se prilagodi, tako što ćete kliknuti desnim tasterom miša na nju i selektovati opciju *Customize Menus and Toolbars*....

C – traka za navigaciju – Obezbeđuje pogodan način za pomeranje fajlova i direktorijuma koji čine projekat. Nakon klika na elemenat u traci za navigaciju, otvoriće se padajući meni sa listom poddirektorijuma i fajlova na konkretnoj lokaciji, spremnih za selekciju. Obezbeđuje se alternativa za prozor *Project*.

D – prozor editora – Prikazuje sadržaj fajla na kojem programer trenutno radi. Međutim, ono što je prikazano na ovoj lokaciji je tema konteksta. Na primer, kada editujete kod, biće prikazan editor koda. Kada radite u fajlu rasporeda korisničkog interfejsa, biće prikazana alatka Layout Editor korisničkog interfejsa. Kada je otvoreno više fajlova, svaki fajl je predstavljen karticom koja se nalazi duž gornje ivice editora, kao što je prikazano na slici 4-3.



Slika 4-3

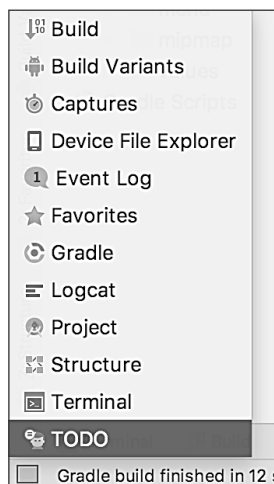
E – statusna traka – Prikazuje informacione poruke o projektu i aktivnostima Android Studioa, a dugme menija sa alatkama se nalazi u donjem levom uglu prozora. Ako postavite kursor na stavke u statusnoj traci, biće prikazan opis konkretnog polja. Mnoga polja su interaktivna i omogućavaju korisniku da klikne da bi izvršio zadatke ili dobio detaljnije informacije o statusu.

F – prozor Project alatke – Obezbeđuje hijerarhijski prikaz strukture fajlova projekta, omogućavajući kretanje do specifičnih fajlova i direktorijuma. Paleta sa alatkama može da se upotrebi za prikaz projekta na više različitih načina. Standardno podešavanje je prikaz *Android*; ovaj režim se primarno koristi u ostatku ove knjige.

Prozor Project alatke je samo jedan od prozora alatki koje su dostupne u Android Studio okruženju.

4.3 Prozori alatke

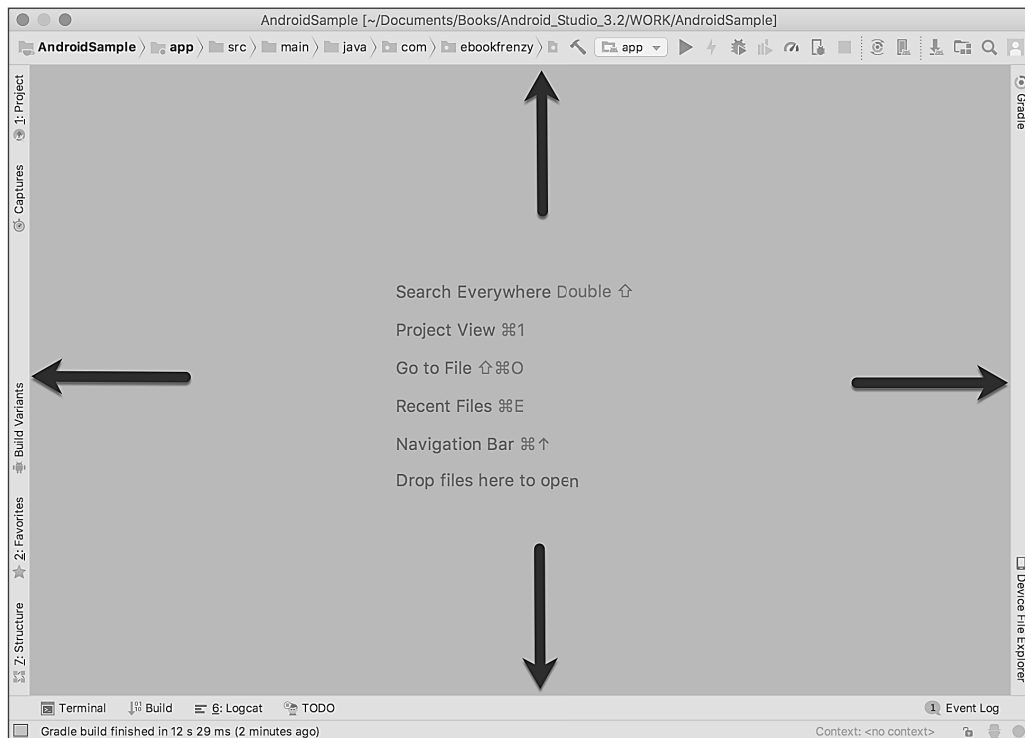
Osim prozora alatke Project, Android Studio uključuje i veliki broj drugih prozora, koji su, kada su uključeni, prikazani duž donje strane i sa obe strane glavnog prozora. Meniju za brz pristup prozoru alatke možete da pristupite tako što ćete postaviti kursor miša iznad dugmeta koje se nalazi u donjem levom uglu statusne trake (slika 4-4), bez klika mišem.



Slika 4-4

Selektovanje stavke iz menija za brz pristup će izazvati prikaz prozora odgovarajuće alatke unutar glavnog prozora.

Alternativno, skup *traka prozora alatke* može da se prikaže klikom na ikonicu menija za brz pristup u statusnoj traci. Ove trake će biti prikazane duž leve, desne i donje ivice glavnog prozora (kao što je prikazano strelicama na slici 4-5) i sadrže dugme za prikazivanje i sakrivanje svakog prozora alatke. Kada se trake prozora alatke prikažu, drugi klik na dugme u statusnoj traci će ih sakriti.

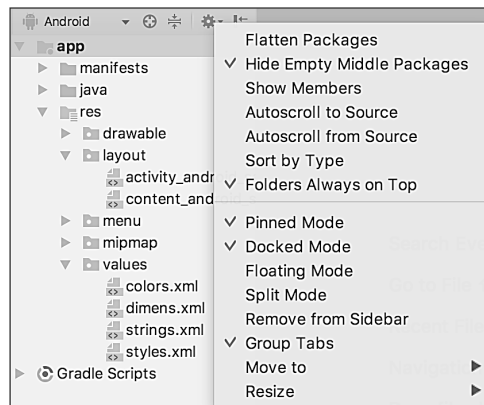


Slika 4-5

Klik na dugme će prikazati prozor odgovarajuće alatke, dok će drugi klik sakriti prozor. Dugmad označena brojevima (na primer, 1: Project) ukazuju da prozor alatke takođe može da bude prikazan pritiskom na taster Alt na tastaturi (ili na taster Command za macOS), zajedno sa odgovarajućim brojem.

Lokacija dugmeta u traci prozora alatke ukazuje na stranu prozora na kojoj će prozor biti prikazan. Ove pozicije mogu da budu promenjene ako kliknete na dugme i prevučete ga na drugu lokaciju u drugoj traci prozora alatke.

Svaki prozor alatke ima sopstvenu paletu sa alatkama duž gornje ivice. Dugmad unutar ovih paleta sa alatkama variraju od alatke do alatke, mada svi prozori alatke sadrže opciju podešavanja, predstavljenu ikonicom zupčanika, koja omogućava da budu promenjeni razni aspekti prozora. Na slici 4-6 prikazan je meni podešavanja za prozor alatke Project view. Na primer, opcije su dostupne za odvajanje prozora iz doka, tako da on pluta van okvira glavnog prozora Android Studioa, i za pomeranje i promenu veličine panela alatke.



Slika 4-6

Svi prozori takođe uključuju dugme na desnoj strani palete sa alatka, koje obezbeđuje dodatni način sakrivanja prozora alatke iz prikaza. Pretraživanje stavki unutar prozora alatke može da bude izvršeno ako prikažete prozor tako što ćete kliknuti na njega, a zatim ukucati termin za pretragu (na primer, naziv fajla u prozoru Project view). Polje za pretraživanje će biti prikazano u paleti sa alatka prozora i biće istaknute stavke koje se poklapaju sa pretragom.

Android Studio ima mnogo prozora alatki, a najčešće se koriste:

Project – Project prikaz obezbeđuje pregled strukture fajlova koji čine projekat i omogućava brzu navigaciju između fajlova. Generalno, kada se u Project view prozoru dvostruko klikne na fajl, on će biti učitao u odgovarajuću alatku za editovanje.

Structure – Alatka Structure obezbeđuje prikaz visokog nivoa strukture izvornog fajla koji je trenutno prikazan u editoru. Ova informacija uključuje listu stavki, kao što su klase, metodi i promenljive u fajlu. Selektovanjem stavke iz strukturne liste u prozoru editora će biti prikazana njena lokacija u izvornom fajlu.

Captures – Prozor alatke Captures obezbeđuje pristup fajlovima podataka performanse koje generišu alatke za praćenje unutar Android Studioa.

Favorites – U listu favorita mogu da se dodaju različite stavke projekta. Na primer, kliknite desnim tasterom miša na fajl u prikazu projekta, što će obezbediti pristup opciji menija *Add to Favorites*. I metod u izvornom fajlu može da bude dodao kao favorit, tako što ćete kliknuti na njega desnim tasterom miša u prozoru alatke Structure. Svemu što je dodato u listu Favorites možete da pristupite kroz prozor alatke Favorites.

Build Variants – Prozor alatke Build Variants obezbeđuje da brzo konfigurišete različite ciljne verzije za projekat aktuelne aplikacije (na primer, različite verzije za ispravljanje grešaka i verzije izdanja aplikacije ili više verzija za kategorije različitih uređaja).

TODO – Ova alatka obezbeđuje mesto za pregled stavki koje treba da se završe u projektu. Android Studio kompajlira tu listu skeniranjem izvornih fajlova koji čine projekat da bi potražio komentare koji se poklapaju sa specifikovanim obrascima TODO. Ovi obrasci mogu da se pregledaju i promene selektovanjem opcije *File -> Settings... (Android Studio -> Preferences... na macOS-u)* i otvaranjem stranice TODO koja je izlistana ispod Editora.

Messages – Prozor alatke Messages snima rezultat iz Gradle build sistema (Gradle je pozadinski sistem koji Android Studio koristi za izgradnju različitih delova projekata u aplikacijama) i može biti koristan za identifikaciju problema izgradnje prilikom kompajliranja projekata aplikacije.

Logcat – Prozor alatke Logcat obezbeđuje pristup za praćenje evidencije iz pokrenute aplikacije, osim opcija za snimanje ekrana i video snimaka aplikacije, i za zaustavljanje procesa restartovanja.

Terminal – Obezbeđuje pristup prozoru terminala na sistemu na kojem je pokrenut Android Studio. Na Windows sistemima to je Command Prompt interfejs, dok na Linux i macOS sistemima preuzima oblik Terminal prompta.

Build – Prozor alatke Build prikazuje informacije o procesu izgradnje dok je projekt kompajliran i spa-kovan i prikazuje detalje grešaka sa kojima se susrećete.

Run – Prozor alatke Run postaje dostupan kada je aplikacija pokrenuta i obezbeđuje prikaze rezultata pokretanja, zajedno sa opcijama za zaustavljanje i restartovanje procesa pokretanja. Ako se aplikacija ne instalira uspešno i ako je pokrenuta na uređaju ili simulatoru, ovaj prozor će, obično, obezbediti informacije dijagnostike koje se odnose na problem.

Logcat – Prozor alatke Logcat obezbeđuje pristup evidenciji praćenja iz pokrenute aplikacije i opcije za snimanje ekrana i video snimaka aplikacije, kao i zaustavljanje i restartovanje procesa.

Event Log – Prozor Event Log prikazuje poruke koje se odnose na događaje i aktivnosti izvršene unutar Android Studioa. Na primer, u ovom prozoru će biti prikazan izveštaj o uspešnoj izgradnji projekta ili činjenica da je aplikacija pokrenuta.

Gradle – Prozor alatke Gradle obezbeđuje prikaz za Gradle zadatke koji čine konfiguraciju izgradnje projekta. U prozoru su izlistani zadaci koji su uključeni u kompajliranje različitih elemenata projekta u izvršnoj aplikaciji. Kliknite desnim tasterom miša na najviši nivo Gradle zadatka i selektujte opciju menija Open Gradle Config da biste učitali fajl Gradle verzije za aktuelni projekat u editoru. Gradle će biti opisan detaljnije kasnije u ovoj knjizi.

Android Profiler – Prozor alatke Android Profiler obezbeđuje alatke za praćenje u realnom vremenu i analizu za identifikaciju problema performanse unutar pokrenutih aplikacija, uključujući CPU i upotrebu memorije i mreže. Ova opcija postaje dostupna kada je aplikacija pokrenuta.

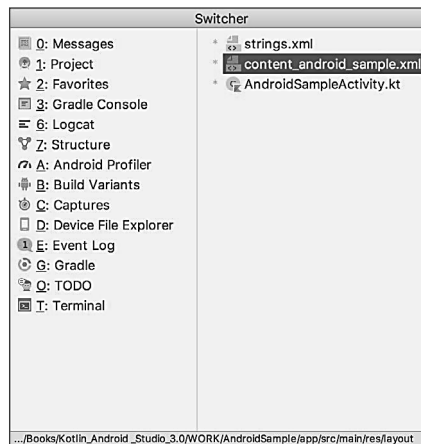
Device File Explorer – Prozor alatke Device File Explorer obezbeđuje direktan pristup za fajl sistem aktuelno povezanih Android uređaja ili simulatora, omogućavajući pretraživanje fajl sistema i kopiranje fajlova u lokalni fajl sistem.

4.4 Prečice na tastaturi Android Studioa

Android Studio uključuje veliki broj prečica na tastaturi koje su projektovane da štede vreme pri izvršavanju uobičajenih zadataka. Celu listu prečica na tastaturi možete da vidite i odštampate unutar prozora Android Studio projekta, tako što ćete selektovati opciju menija *Help -> Keymap Reference*.

4.5 Switcher i pronalaženje nedavno otvaranih fajlova

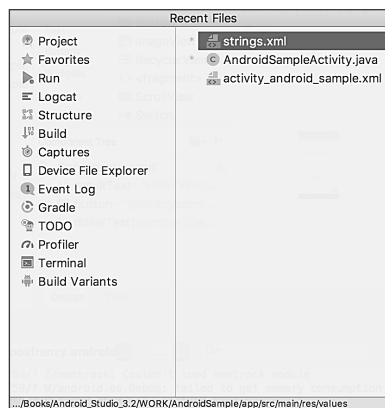
Još jedan koristan mehanizam za navigaciju unutar glavnog prozora Android Studioa uključuje upotrebu *Switchera*. Možete da mu pristupite pomoću prečice Ctrl-Tab - otvoriće se panel switchera koji lista prozore alatki i nedavno otvarane fajlove (slika 4-7).



Slika 4-7

Kada je prikazan, Switcher će ostati vidljiv, sve dok taster Ctrl ostane pritisnut. Uzastopnim pritiskanjem tastera Tab, dok držite pritisnut taster Ctrl, kružićete kroz razne opcije selekcije, dok otpuštanje Ctrl tastera izaziva selektovanje aktuelno istaknute stavke koja će biti prikazana u glavnom prozoru.

Osim u Switcheru, pronalaženje nedavno otvorenih fajlova obezbeđeno je u panelu Recent Files (slika 4-8). Ovom panelu možete da pristupite i pomoću prečice na tastaturi Ctrl-E (Cmd-E na macOS-u). Kada je panel prikazan, možete da upotrebite kursor za selektovanje opcije ili, alternativno, možete da upotrebite tastere sa strelicama da biste skrolovali kroz nazive fajlova i opcije prozora alatke. Pritiskom na taster Enter selektovaćete aktuelno istaknutu stavku.

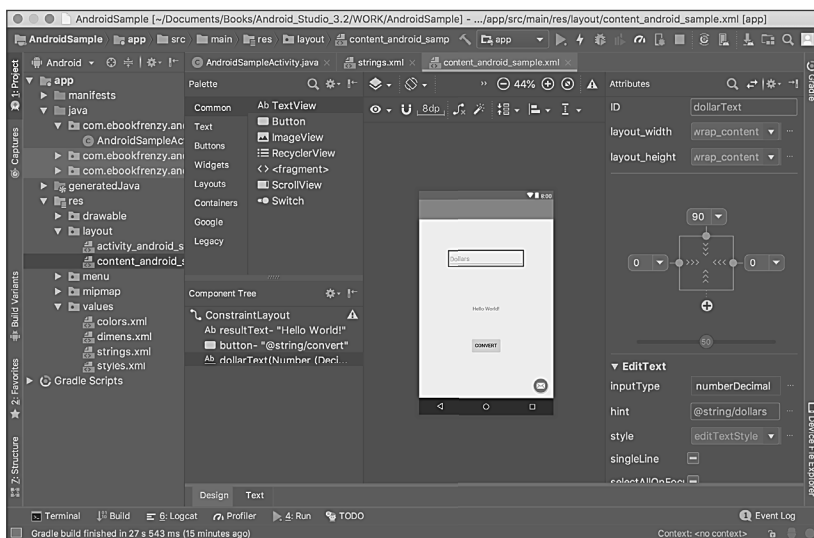


Slika 4-8

4.6 Menjanje Android Studio teme

Uopštena tema Android Studio okruženja može da bude promenjena sa ekrana dobrodošlice korišćenjem opcije Configure -> Settings ili pomoću opcije menija File -> Settings... (*Android Studio -> Preferenc-es...* na macOS-u) u glavnom prozoru.

Kada je prikazan okvir za dijalog sa podešavanjima, selektujte opciju *Appearance* u panelu sa leve strane, a zatim promenite podešavanje menija *Theme*, pre nego što kliknete na dugme *Apply*. Dostupne teme će zavisići od platforme, ali, obično, uključuju opcije kao što su *Light*, *IntelliJ*, *Windows*, *Default* i *Darcula*. Na slici 4-9 prikazan je primer glavnog prozora sa selektovanom temom *Darcula*.



Slika 4-9

4.7 Rezime

Primarni elementi Android Studio okruženja se sastoje od ekrana dobrodošlice i glavnog prozora. Svakom otvorenom projektu je dodeljen glavni prozor, koji se sastoji od linije menija, palete sa alatka, oblasti za editovanje i projektovanje, statusne trake i kolekcije prozora alatke. Prozori alatke se prikazuju sa leve, desne i donje strane glavnog prozora i može im se pristupiti upotrebom menija za brz pristup, koji se nalazi u statusnoj traci, ili korišćenjem opcionih traka prozora alatke.

Postoji veoma malo akcija unutar Android Studioa koje ne mogu da budu pokrenute pomoću prečice na tastaturi. Listi standardnih prečica na tastaturi možete da pristupite bilo kada unutar glavnog prozora Android Studioa.



KREIRANJE ANDROID VIRTUELNOG UREĐAJA (AVD) U RAZVOJNOM OKRUŽENJU ANDROID STUDIO

Da biste kreirali Android aplikacije u Android Studio, potrebno je da kompajlirate i pokrenete aplikaciju više puta. Android aplikaciju možete da testirate tako što ćete je instalirati i pokrenuti na fizičkom uređaju ili u simuliranom okruženju *Android virtuelnog uređaja (AVD)*. Pre nego što upotrebite AVD, morate da ga kreirate i konfigurišete da se poklapa sa specifikacijama određenog modela uređaja. Cilj u ovom poglavlju je, prema tome, da opišemo korake uključene u kreiranje takvog virtuelnog uređaja, koristeći model telefona Nexus 5X kao referentni primer.

5.1 O Android virtuelnim uređajima

AVD-ovi su, u stvari, emulatori, koji omogućavaju da Android aplikacije budu testirane, bez potrebe da se instaliraju na fizički Android uređaj. AVD može da bude konfigurisani za simuliranje različitih funkcija hardvera, uključujući i opcije, kao što su veličina ekrana, kapacitet memorije i prisutnost nekih funkcija, kao što su kamera, podrška za GPS navigaciju ili merač ubrzanja. Kao deo standardne instalacije Android Studioa, instaliran je veliki broj šablona emulatora koji omogućavaju AVD-ovima da budu konfigurisani za različite uređaje. Dodatni šabloni mogu da budu učitani ili se mogu kreirati prilagođene konfiguracije koje se poklapaju sa nekim fizičkim Android uređajem, specifikovanjem svojstava, kao što su tip procesora, kapacitet memorije i veličina i gustina piksela ekrana. Proverite online dokumentaciju za uređaj da biste otkrili da li su dostupne definicije emulatora za preuzimanje i instalaciju u AVD okruženju.

Kada je pokrenut, AVD će biti prikazan kao prozor koji sadrži okruženje simuliranog Android uređaja. Na primer, na slici 5-1 prikazana je AVD sesija koja je konfigurisana za simuliranje Google Nexus 5X modela.

Novi AVD-ovi su kreirani i njima se upravlja pomoću Android Virtual Device Managera, koji može da se upotrebi u režimu komandne linije, ili korišćenjem grafičkog korisničkog interfejsa.



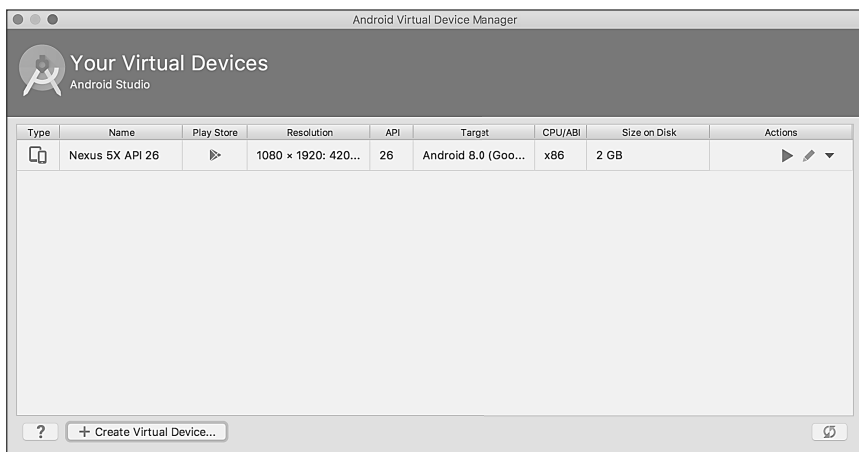
Slika 5-1

5.2 Kreiranje novog AVD-a

Da biste testirali ponašanje aplikacije u odsustvu fizičkog uređaja, potrebno je da kreirate AVD za konfiguraciju specifičnog Android uređaja.

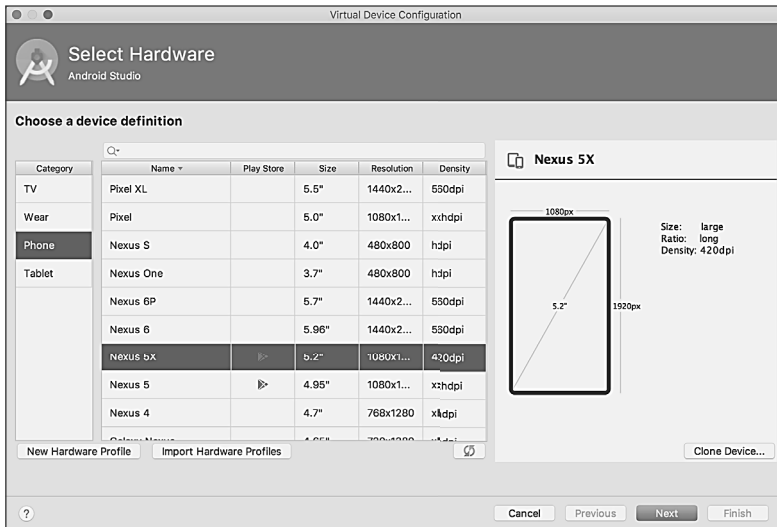
Da biste kreirali novi AVD, prvi korak je da pokrenete AVD Manager. To možete da uradite unutar Android Studio okruženja selektovanjem opcije menija `Tools -> Android -> AVD Manager` unutar glavnog prozora.

Kada je otvorena, alatka će biti prikazana kao na slici 5-2 ako su kreirane postojeće AVD instance.



Slika 5-2

Radi dodavanja AVD-a, najpre kliknite na dugme `Create Virtual Device` da biste otvorili okvir za dijalog *VirtualDevice Configuration*.



Slika 5-3

Unutar okvira za dijalog izvršite sledeće korake da biste kreirali emulator kompatibilan sa uređajem Nexus 5X:

1. Iz panela Category selektujte opciju Phone da biste prikazali listu dostupnih Android tabled AVD šablona.
2. Selektujte opciju Nexus 5X i kliknite na Next.
3. Na ekranu System Image selektujte najnoviju verziju Androida (u vreme pisanja ove knjige to je API nivoa 28, Android 9.0 with Google Play) za x86 ABI. Ako sistemski imidž još nije instaliran, biće prikazan link Download pored polja Release Name. Kliknite na ovaj link da biste preuzeli i instalirali sistemski imidž, pre nego što ga selektujete. Ako imidž koji vam je potreban nije izlistan, kliknite na kartice x86 images i Other images da biste prikazali liste alternativa.
4. Kliknite na Next da biste nastavili i unesite opisni naziv (na primer, Nexus 5X API 28) u polje Name ili, jednostavno, prihvatite standardni naziv.
5. Kliknite na Finish da biste kreirali AVD.
6. Kada je AVD kreiran, AVD Manager možete zatvoriti. Ako ubuduće budu potrebne modifikacije, jednostavno ponovo otvorite AVD Manager, selektujte AVD iz liste i kliknite na ikonicu olovke u koloni Actions reda uređaja u AVD Manageru.

5.3 Pokretanje emulatora

Da biste izvršili probno pokretanje novokreiranog AVD emulatora, jednostavno selektujte emulator iz AVD Managera i kliknite na dugme za pokretanje (zeleni trougao u koloni Actions). Emulator će biti prikazan u novom prozoru i zapoćeće proces pokretanja. Vreme koje je mu potrebno da se pokrene će zavisi od konfiguracije AVD-a i sistema na kojem se pokreće. U slučaju da je vreme pokretanja predugo, nemojte oklevati da ostavite emulator pokrenut. Sistem će detektovati da je već pokrenut i povećaće ga kada se aplikacije pokrenu, pa ćete na taj način uštedeti značajno vreme za pokretanje.

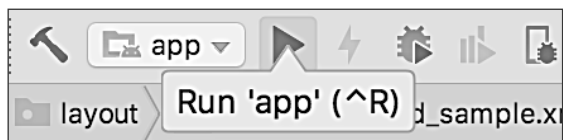
Prema standardnom podešavanju, emulator se otvara u portretnoj orijentaciji. Važno je da imate na umu da ova i druge standardne opcije mogu da budu promenjene. Unutar AVD Managera selektujte novi

Nexus 5X unos i kliknite na ikonicu olovke u koloni Actions reda uređaja. Na ekranu konfiguracije pronađite odeljak Startup and orientation i promenite podešavanje za orijentaciju. Zatvorite i restartujte sesiju emulatora da biste videli efekte promene. Više detalja o emulatoru pročitajte u Poglavlju 6, „Upotreba i konfigurisanje Android Studio AVD emulatora“.

Da biste uštedeli vreme u sledećem odeljku ovog poglavlja, ostavite emulator pokrenut pre nego što nastavite rad.

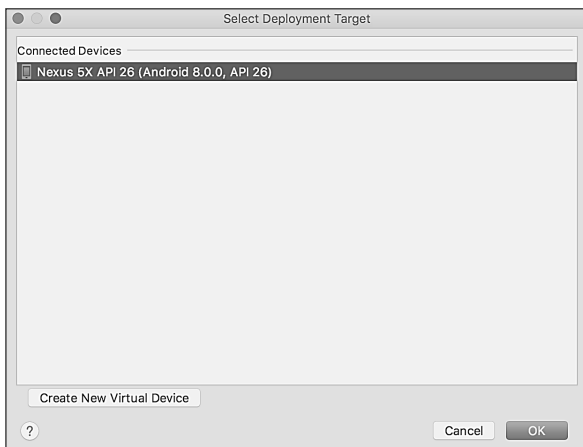
5.4 Pokretanje aplikacije u AVD-u

Sada, kada je konfigurisan AVD emulator, primer Android Sample aplikacije koju smo kreirali u prethodnom poglavlju može da bude kompajliran i pokrenut. Kada je Android Sample projekat učitao u Android Studio, jednostavno kliknite na dugme Run (prikazano je zelenim trougлом, a nalazi se u paleti sa alatkama Android Studioa, kao što je predstavljeno na slici 5-4), selektujte opciju menija *Run -> Run* 'app' ili upotrebite prečicu na tastaturi Ctrl-R.



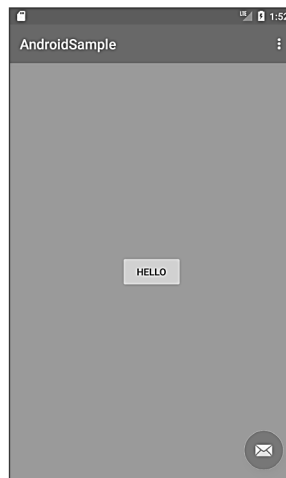
Slika 5-4

Prema standardnom podešavanju, Android Studio će odgovoriti na pokretanje zahteva prikazom okvira za dijalog *Select Deployment Target*, koji obezbeđuje opciju za izvršenje aplikacije na AVD instanci, koja je već pokrenuta, ili za pokretanje nove AVD sesije, koje je specifično za ovu aplikaciju. Na slici 5-5 izlistan je prethodno kreirani Nexus 5X AVD kao pokrenuti uređaj, što je rezultat izvršenih koraka u prethodnom odeljku. Kada je ovaj uređaj selektovan u okviru za dijalog, kliknite na OK da biste instalirali i pokrenuli aplikaciju na emulatoru.



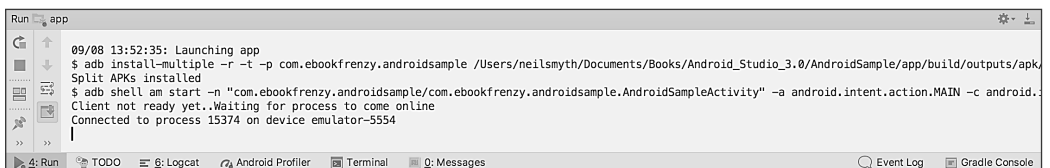
Slika 5-5

Kada je aplikacija instalirana i pokrenuta, korisnički interfejs za klasu Android Sample Activity će biti prikazan unutar emulatora.



Slika 5-6

Ukoliko se aktivnost ne pokrene automatski, proverite da li se ikonica za pokretanje prikazuje u ostalim aplikacijama na emulatoru. Ako se prikazuje, jednostavno kliknite na nju da biste pokrenuli aplikaciju. Kada započne proces pokretanja, otvoriće se prozor alatke Run i Logcat. Prozor alatke Run će prikazati dijagnostičke informacije kada je paket aplikacije instaliran i pokrenut. Na slici 5-7 prikazan je prozor alatke Run sa rezultatom uspešnog pokretanja aplikacije.



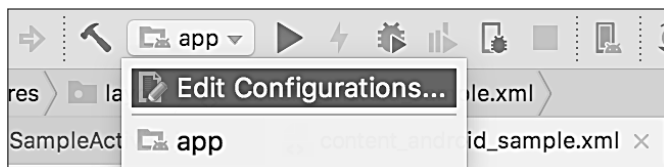
Slika 5-7

Ako se u toku procesa pokretanja suočite sa problemima, prozor alatke Run će obezbediti informacije koje će vam pomoći da otkrijete uzrok problema.

Pod pretpostavkom da se aplikacija učitava u emulator i pokreće kao što se i očekuje, slobodno možete da potvrdite da je Android razvojno okruženje pravilno instalirano i konfigurisano.

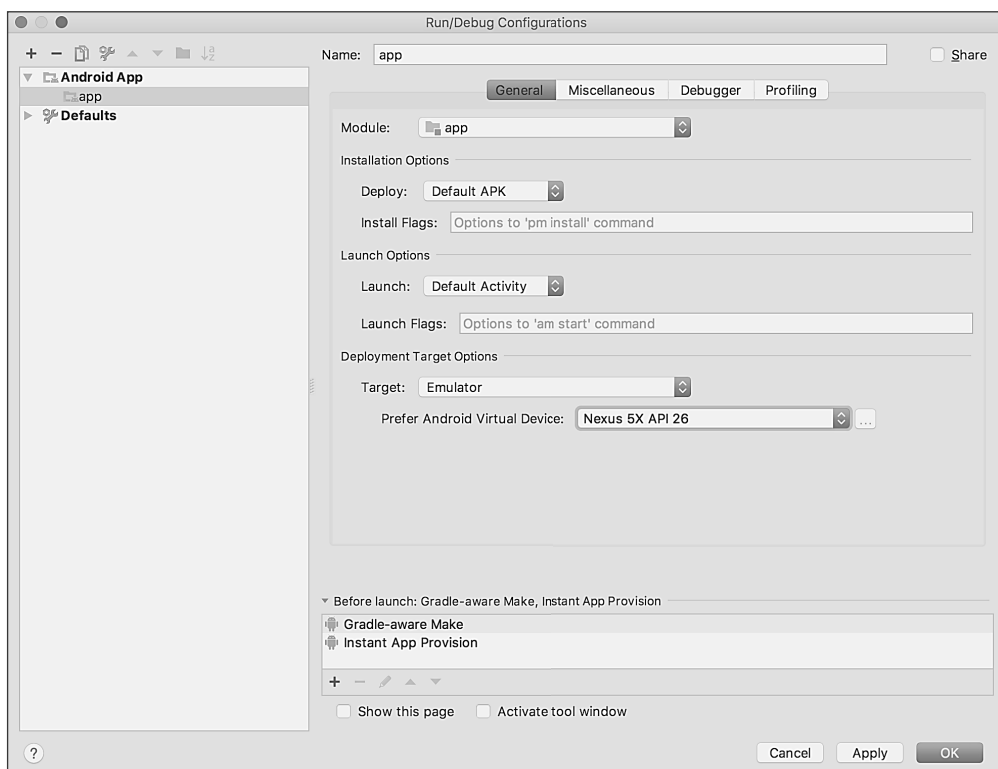
5.5 Run/Debug konfiguracije

Određeni projekat može da bude konfigurisan tako da specifični uređaj ili emulator budu upotrebljeni automatski uvek kada je projekat pokrenut unutar Android Studioa. Zahvaljujući tome, izbeći ćete potrebu da vršite selekciju iz birača uređaja kada izvršavate aplikaciju. Da biste pregledali i modifikovali Run/Debug konfiguraciju, kliknite na dugme levo od dugmeta za pokretanje u paleti sa alatkama Android Studioa i selektujte opciju *Edit Configurations...* iz rezultirajućeg menija.



Slika 5-8

U okviru za dijalog *Run/Debug Configurations* aplikacija može da bude konfigurisana da uvek koristi izabrani emulator, tako što ćete selektovati Emulator iz menija *Target*, koji se nalazi u odeljku *Deployment Target Options*, i selektovati emulator iz padajućeg menija. Na primer, na slici 5-9 prikazana je Android Sample aplikacija konfigurisana da se prema standardnom podešavanju pokreće na prethodno kreiranom Nexus 5X emulatoru.

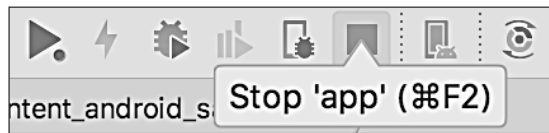


Slika 5-9

Obavezno promenite podešavanje menija *Target* nazad na *Open Select Deployment Target Dialog* režim, pre nego što pređete na sledeće poglavlje ove knjige.

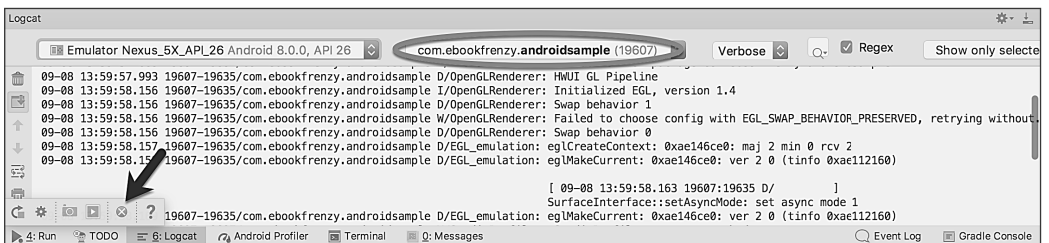
5.6 Zaustavljanje pokrenute aplikacije

Da biste zaustavili pokrenutu aplikaciju, jednostavno kliknite na dugme Stop u glavnoj paleti sa alat-kama, kao što je prikazano na slici 5-10.



Slika 5-10

Aplikacija može da bude isključena i upotrebom prozora alatke Logcat. Prvo prikažite taj prozor, koristeći dugme trake prozora koja će postati dostupna kada je aplikacija pokrenuta. Kada se otvori prozor alatke Logcat, selektujte androidsample, koji je istaknut na slici 5-11.



Slika 5-11

Kada je proces selektovan, zaustavite ga, tako što ćete kliknuti na crveno dugme Terminate Application u paleti sa alatkama levo od liste procesa, kao što je prikazano strelicom na prethodnoj slici.

5.7 Kreiranje AVD-a iz komandne linije

Kao što je prethodno opisano, osim upotrebom grafičkog korisničkog interfejsa, takođe je moguće kreirati novi AVD direktno iz komandne linije. To možete da uradite upotrebom alatke avdmanager, zajedno sa nekim opcijama komandne linije. Kada je pokrenuta, alatka će zatražiti dodatne informacije pre kreiranja novog AVD-a.

Pod pretpostavkom da je sistem konfigurisan tako da je Android SDK direktorijum tools uključen u promenljivu okruženja PATH, lista dostupnih ciljeva za novi AVD može da se dobije unosom sledeće komande u terminal ili prozor Command:

```
avdmanager list targets
```

Rezultat iz ove komande će sadržati listu Android SDK verzija koje su dostupne na sistemu. Na primer:

```
Available Android targets:
```

```
-----
```

```
id: 1 or "android-28"
    Name: Android API 28
    Type: Platform
    API level: 28
```

```

Revision: 3
-----
id: 2 or "android-26"
  Name: Android API 26
  Type: Platform
  API level: 26
  Revision: 1

```

Alatka avdmanager takođe omogućava da nove AVD instance budu kreirane iz komandne linije. Na primer, da biste kreirali novi AVD pod nazivom Nexus9, koristeći ciljni ID za Android API level 26 uređaj koji koristi x86 ABI, možete da upotrebite sledeću komandu:

```
avdmanager create avd -n Nexus9 -k "system-images;android-26;google_apis;x86"
```

Alatka android će kreirati novi AVD za specifikacije potrebne za osnovni Android 8 uređaj, obezbeđujući i opciju za kreiranje prilagođene konfiguracije koja se poklapa sa specifikacijama određenog uređaja, ako je potrebno. Kada je novi AVD kreiran iz komandne linije, možda neće biti prikazan u alatki Android Device Manager, dok ne kliknete na dugme Refresh.

Osim kreiranja novih AVD-ova, veliki broj drugih zadataka može da bude izvršen iz komandne linije. Na primer, listu dostupnih AVD-ova možete da dobijete korišćenjem list avd argumenata komandne linije:

```

Available Android Virtual Devices:
  Name: Pixel_XL_API_28_No_Play
  Device: pixel_xl (Google)
  Path: /Users/neilsmyth/.android/avd/Pixel_XL_API_28_No_Play.avd
  Target: Google APIs (Google Inc.)
        Based on: Android API 28 Tag/ABI: google_apis/x86
  Skin: pixel_xl_silver
  Sdcard: 512M

```

Da biste izbrisali postojeći AVD, jednostavno upotrebite opciju delete na sledeći način:

```
avdmanager delete avd -n <avd name>
```

5.8 Konfiguracioni fajlovi Android virtuelnog uređaja

Prema standardnom podešavanju, fajlovi povezani sa AVD-om su sačuvani u poddirektorijumu .android/avd direktorijuma home za korisnika - njegova struktura je sledeća (<avd name> je zamenjen nazivom koji je dodeljen AVD-u):

```

<avd name>.avd/config.ini
<avd name>.avd/userdata.img
<avd name>.ini

```

Fajl *config.ini* sadrži podešavanja konfiguracije uređaja, kao što su dimenzije prikaza i memorija specifikovana u toku procesa kreiranja AVD-a. Ova podešavanja mogu da budu promenjena direktno unutar konfiguracionog fajla i AVD će ih prihvatiti kada sledećeg puta bude otvoren.

Fajl `<avd name>.ini` sadrži referencu za ciljni Android SDK i putanju ka AVD fajlovima. Imajte na umu da promena vrednosti za `image.sysdir` u fajlu `config.ini` takođe treba da bude reflektovana u ciljnoj vrednosti ovog fajla.

5.9 Pomeranje i promena naziva Android virtuelnog uređaja

Aktuelni naziv ili lokacija AVD fajlova mogu da budu promenjeni iz komandne linije upotrebom argumenta *move* avd alatke *avdmanager*. Na primer, da biste promenili naziv AVD-a pod nazivom Nexus9 na Nexus9B, možete da upotrebite sledeću komandu:

```
avdmanager move avd -n Nexus9 -r Nexus9B
```

Da biste fizički promenili lokaciju fajlova koji su povezani sa AVD-om, upotrebite sledeću komandu:

```
avdmanager move avd -n <avd name> -p <path to new location>
```

Na primer, da biste pomerili AVD iz aktuelne lokacije u fajl sistemu u direktorijum `/tmp/Nexus9Test`, upotrebite sledeći kod:

```
avdmanager move avd -n Nexus9 -p /tmp/Nexus9Test
```

Imajte na umu da krajnji direktorijum ne mora da postoji pre izvršenja komande za pomeranje AVD-a.

5.10 Rezime

Tipičan proces razvoja aplikacije prati ciklus kodiranja, kompajliranja i pokretanja u okruženju testiranja. Android aplikacije mogu da budu testirane na fizičkom Android uređaju ili upotrebom emulatora Android virtuelnog uređaja (AVD). AVD-ovi su kreirani i njima se upravlja upotrebom AVD Manager alatke (ona može da se upotrebi kao alatka komandne linije) ili upotrebom grafičkog korisničkog interfejsa. Kada kreirate AVD za simuliranje specifičnog modela Android uređaja, važno je da virtuelni uređaj bude konfigurisan hardverskim specifikacijama koje se poklapaju sa onima na fizičkom uređaju.

