

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



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 „Osnove Android Studio razvoja“.

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 potkrađu 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
```

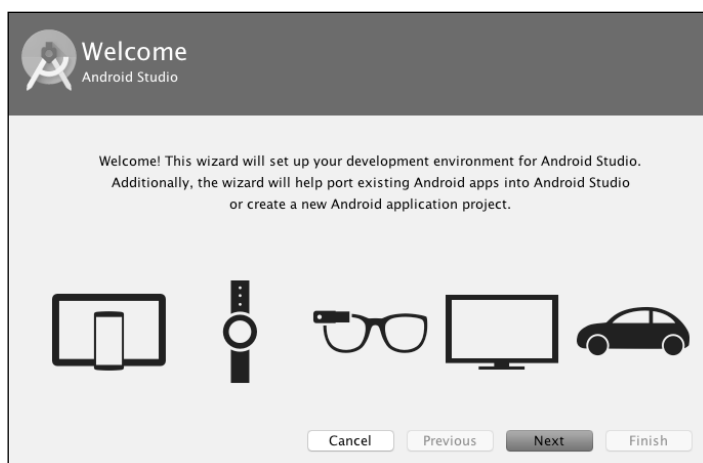
Na RedHat i Fedora 64-bitnim sistemima upotrebite sledeću komandu:

```
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.

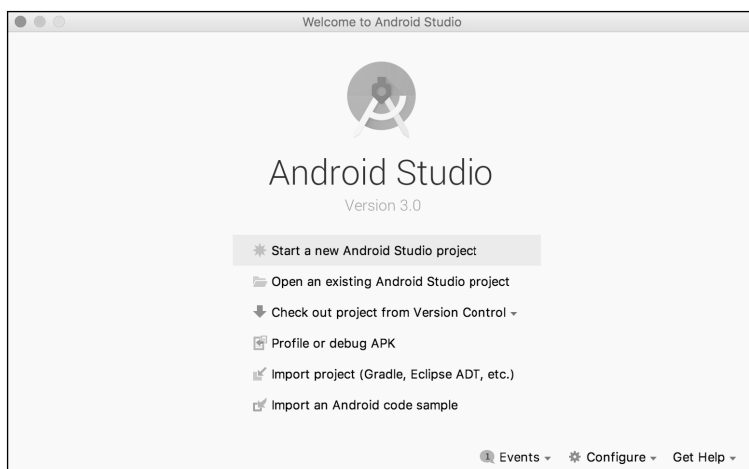
Setup Wizard može izgledati kao na slici 2-2, mada ovaj okvir za dijalog neće biti prikazan na svim platformama:



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 konfigurisanje 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:

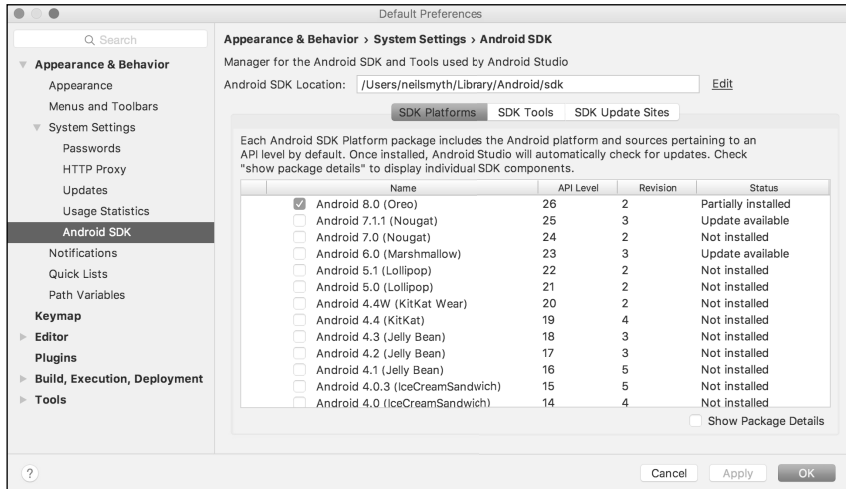


Slika 2-3

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.

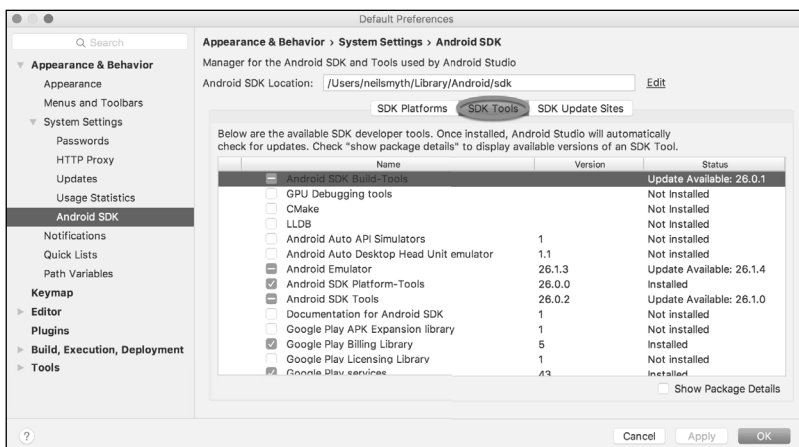
Takođe je moguće da će ažuriranja biti izlistana kao dostupna za najnoviji SDK. Da biste pristupili detaljnim informacijama o paketima koji su dostupni za ažuriranje, uključite opciju Show Package Details, koja se nalazi u donjem desnom uglu ekrana. Biće prikazane informacije koje su slične onima prikazanim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

	Name	API Level	Revision	Status
☐	Android TV Intel x86 Atom System Image	25	6	Not installed
☐	Android Wear for China ARM EABI v7a System Image	25	3	Not installed
☐	Android Wear for China Intel x86 Atom System Image	25	3	Not installed
☐	Android Wear ARM EABI v7a System Image	25	3	Not installed
☐	Android Wear Intel x86 Atom System Image	25	3	Not installed
☐	Google APIs ARM 64 v8a System Image	25	8	Not installed
☐	Google APIs ARM EABI v7a System Image	25	8	Not installed
☐	Google APIs Intel x86 Atom System Image	25	8	Not installed
☒	Google APIs Intel x86 Atom_64 System Image	25	6	Update Available: 8
▼ ☐	Android 7.0 (Nougat)			
☐	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 alatki 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.

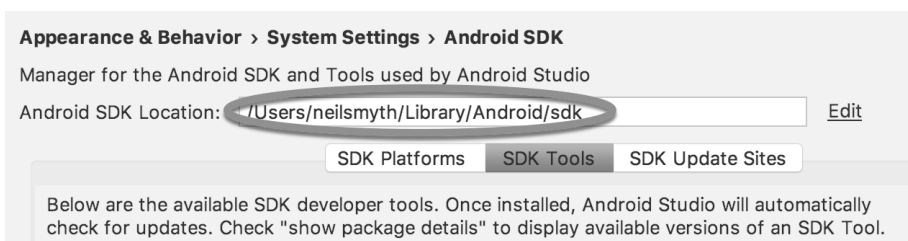
2.6 Učinite dostupnom komandnu liniju Android SDK alati

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 alati 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 panela 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 alati 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 alati 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 alati 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 podese 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

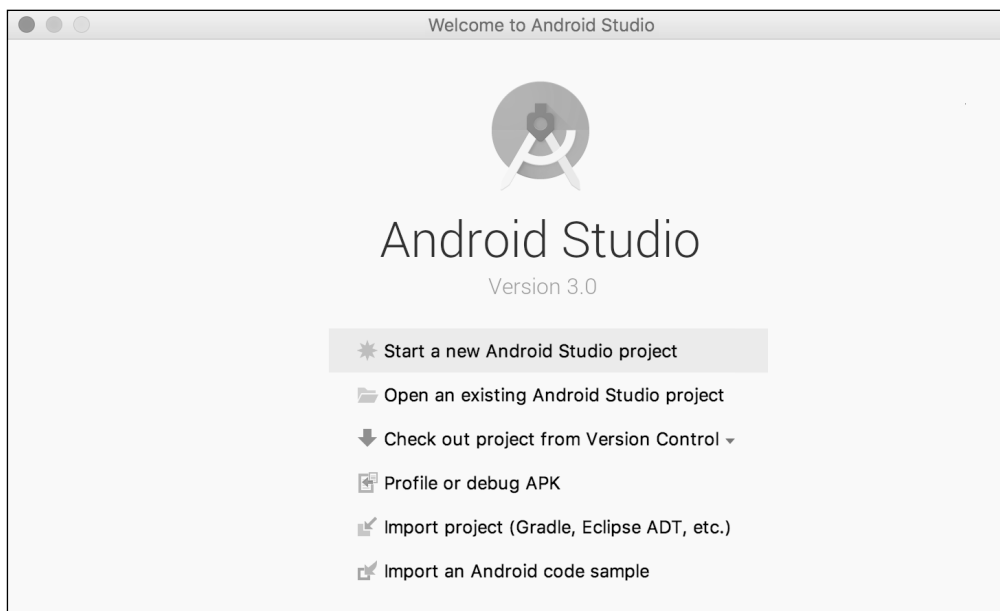
U prethodnim poglavljima ove knjige opisani su koraci za konfigurisanje okruženja koje je pogodno za Android aplikacije pomoću Android Studio IDE-a. Pre nego što nastavite učenje malo naprednijih tema, sada je dobro vreme da utvrdite da li su instalirani svi potrebni paketi i da li funkcionišu pravilno. 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 0 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.

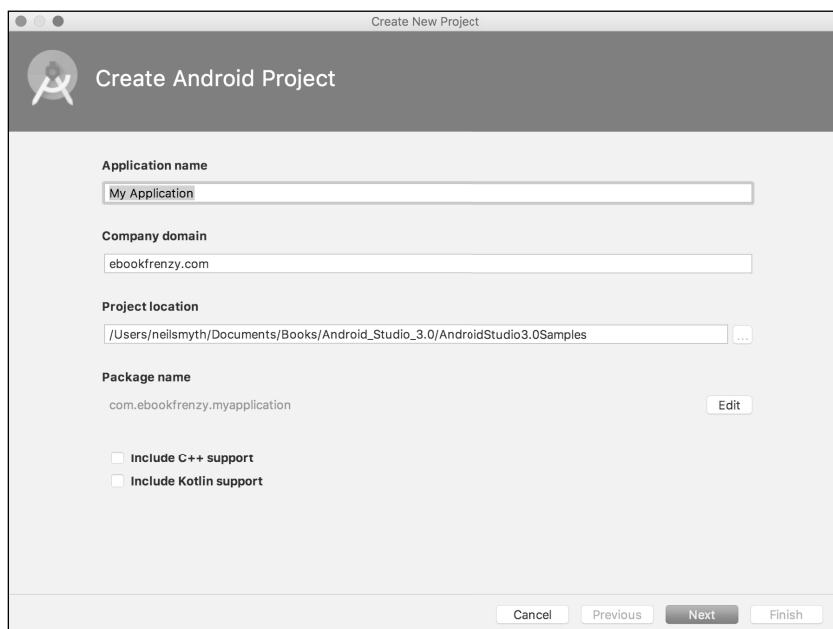
3.2 Kreiranje novog Android projekta

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
```

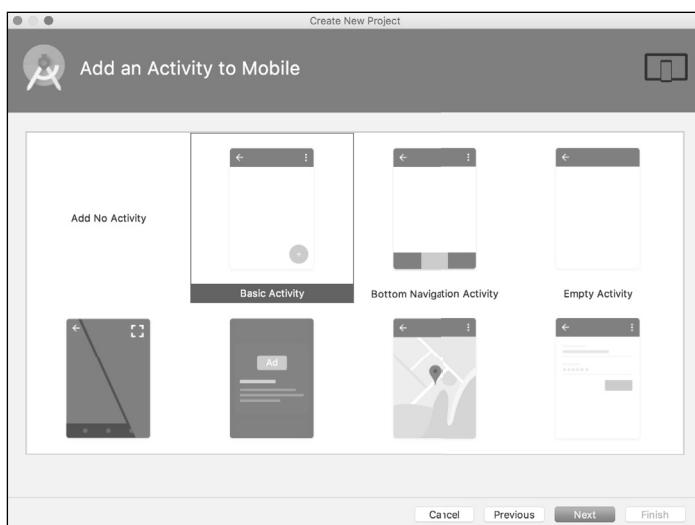
Podešavanje Project location će biti standardno podešavanje za lokaciju u direktorijumu `AndroidStudio-Projects`, koji se nalazi u home direktorijumu i može da bude promenjen ako kliknete na dugme desno od tekstualnog polja koje sadrži aktuelnu putanju.

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`.

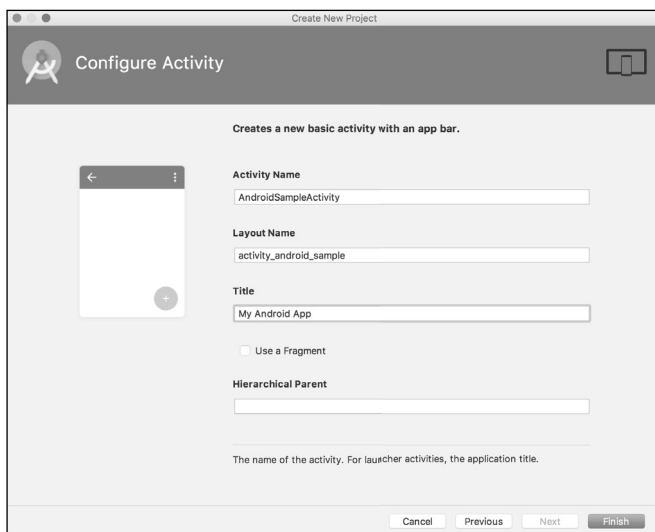
3.4 Kreiranje aktivnosti

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 poslednjem 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.

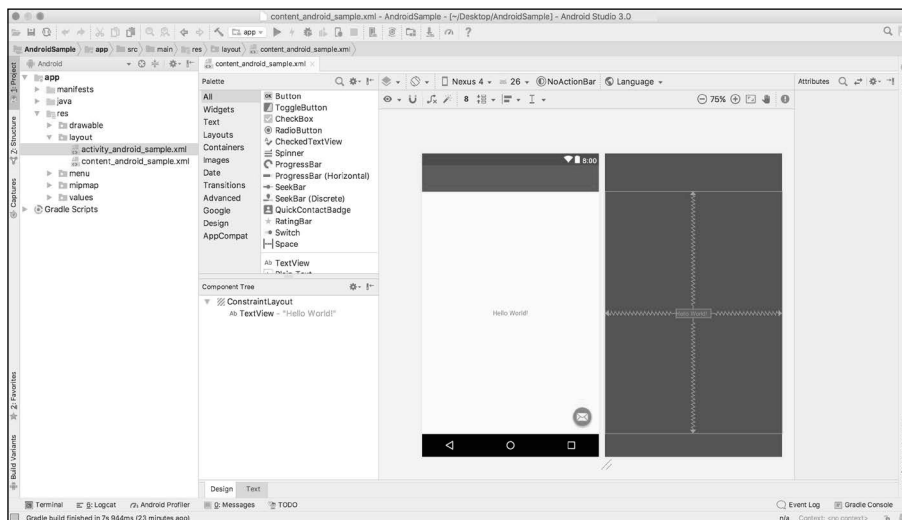


Slika 3-4

Kliknite na Finish da biste pokrenuli proces kreiranja projekta.

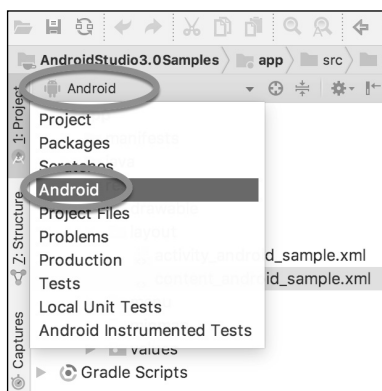
3.5 Modifikovanje primera aplikacije

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.

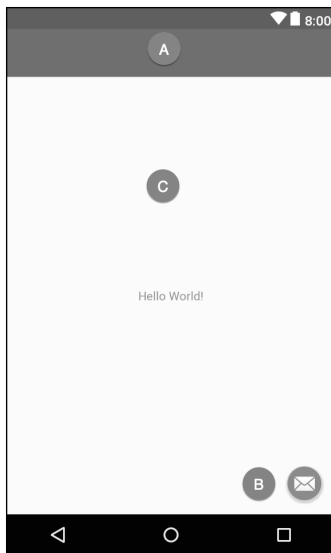


Slika 3-6

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

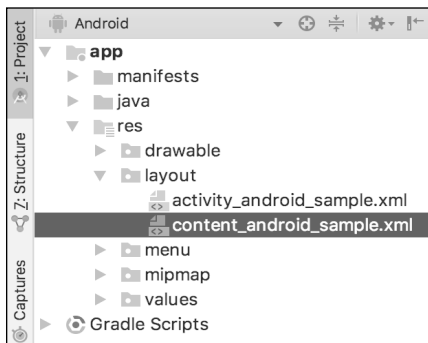
Sledeći korak u ovom uputstvu je da modifikujete korisnički interfejs aplikacije da prikazuje veći objekat prikaza teksta sa porukom drugačijom od one koju prikazuje Android Studio.

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).



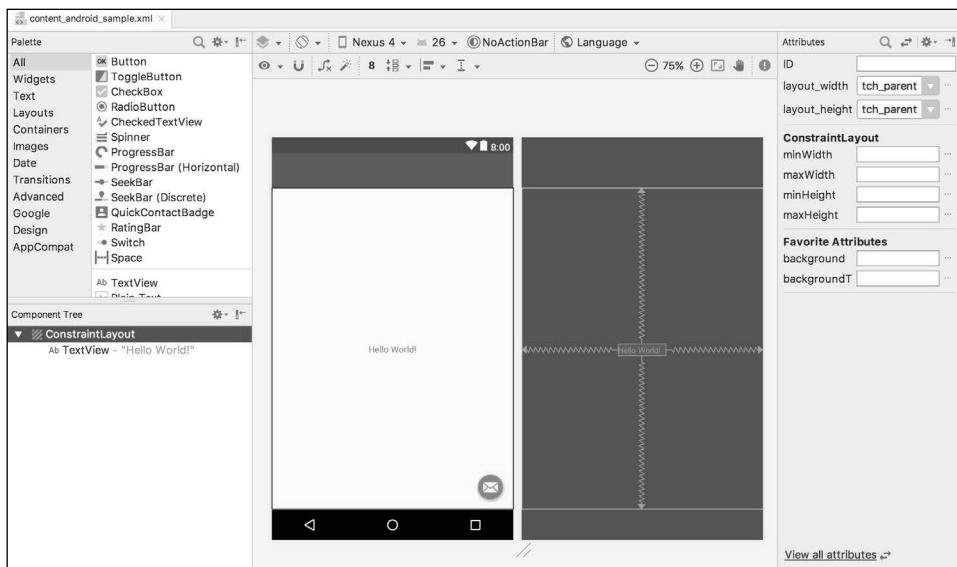
Slika 3-7

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.

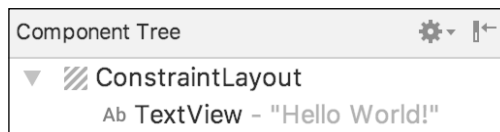


Slika 3-9

U paleti sa alatkama 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 pada-jući 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 komponenata 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.

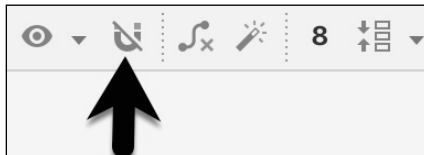


Slika 3-10

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 Editor uključen. To znači da će, dok se komponente dodaju u raspored, Layout Editor automatski dodati ograničenja da bi bio siguran

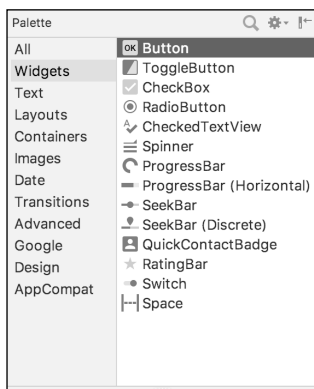
da su komponente tačno pozicionirane za različite veličine ekrana i orijentaciju uređaja (ova tema će biti detaljnije opisana u narednim poglavljima). Dugme Autoconnect se prikazuje kao ikonica magneta u paleti sa alatkama Layout Editora. Kada je opcija isključena, magnet će biti prevučen dijagonalnom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



Slika 3-11

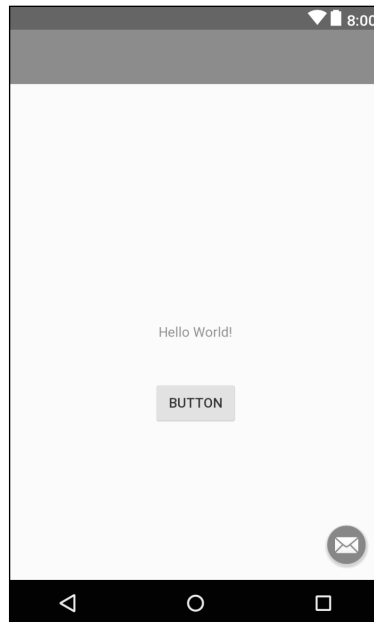
Sledeći korak u modifikovanju aplikacije je da dodate komponente u raspored; prva će biti dugme koje korisnik treba da pritisne da bi pokrenuo konverziju valute.

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.



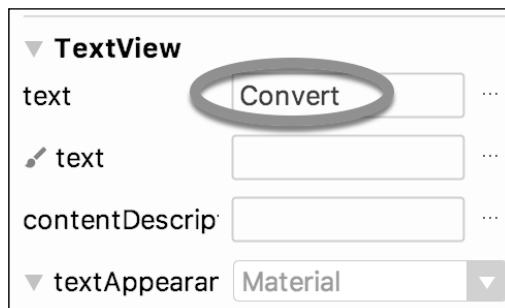
Slika 3-12

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.

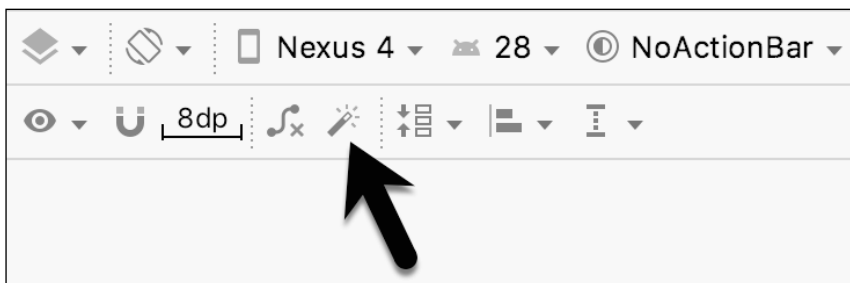


Slika 3-14

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.

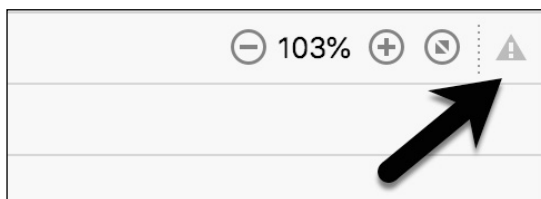
Drugo svojstvo text, pored kojeg se nalazi ikonica sa ključem, omogućava da svojstvo text bude podešeno, što će biti prikazano samo unutar alatke Layout Editor, ali ne u toku izvršenja. To je korisno za testiranje načina na koji će se vizuelna komponenta i raspored ponašati sa različitim podešavanjima,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

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.



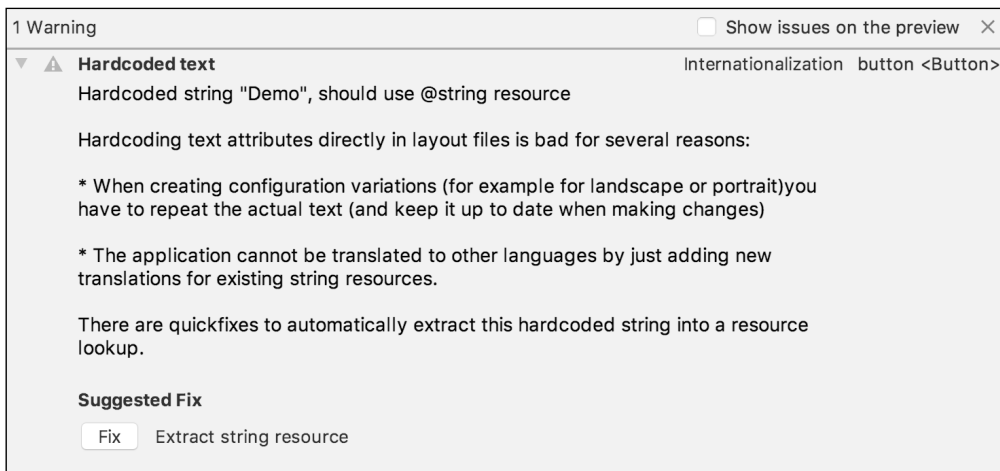
Slika 3-15

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.



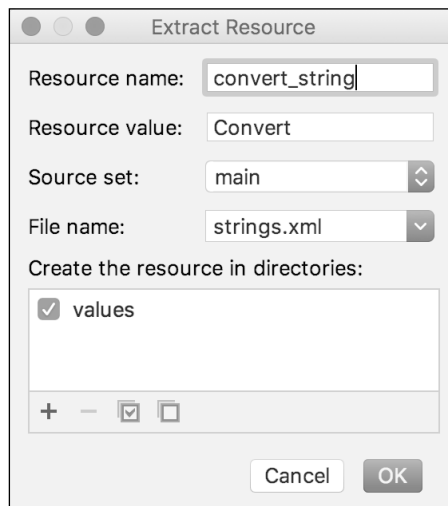
Slika 3-17

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 ekstrahovanje potrebnih resursa kada je raspored završen.

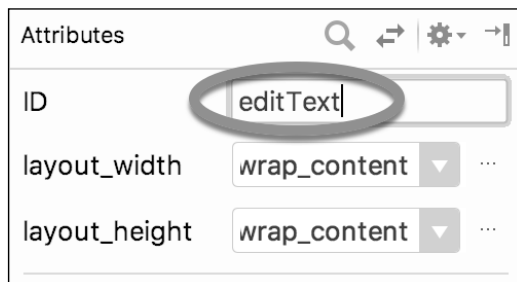
Sledeći vidžet koji ćemo dodati je EditText, u koji će korisnik uneti vrednost dolara koja će biti konvertovana. Iz palete vidžeta selektujte kategoriju Text i kliknite i prevucite komponentu Number (Decimal) na raspored da bude centrirana horizontalno i pozicionirana iznad postojećeg vidžeta TextView. 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.

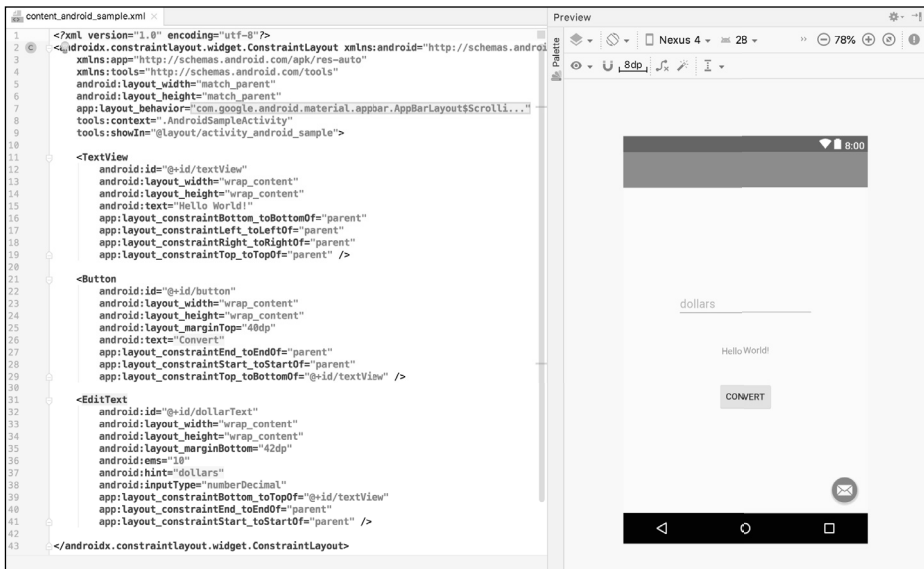


Slika 3-20

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 demonstrirajući 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 #a0ff28 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:

```

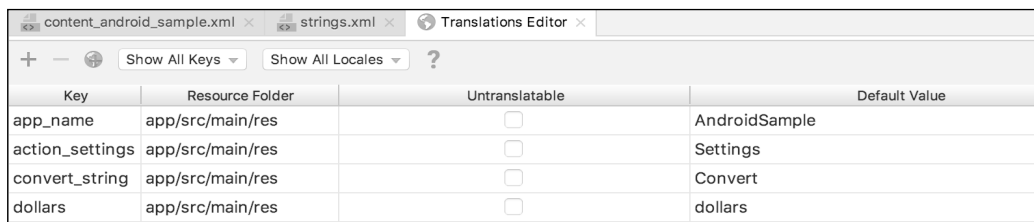
<resources>
    <string name="app_name">AndroidSample</string>
    <string name="action_settings">Settings</string>
    <string name="convert_string">Convert</string>
    <string name="dollars">dollars</string>
</resources>

```

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.



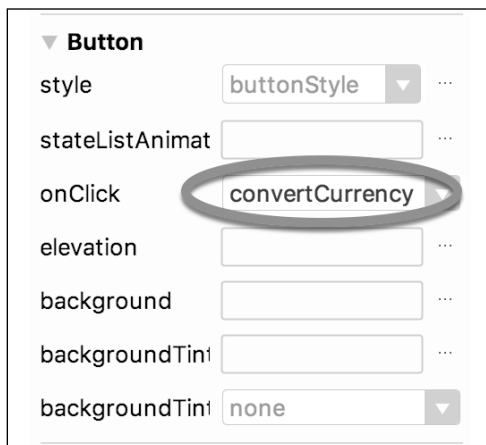
Key	Resource Folder	Untranslatable	Default Value
app_name	app/src/main/res	☐	AndroidSample
action_settings	app/src/main/res	☐	Settings
convert_string	app/src/main/res	☐	Convert
dollars	app/src/main/res	☐	dollars

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 zareзом 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.