Android Studio
(ako začať)
Jazyk Kotlin
(ako neskončiť)

Peter Borovanský
KAI, I-18

MS-Teams: 2sf3ph4, List, github

borovan 'at' ii.fmph.uniba.sk

Vývojové platformy

(pre natívne appky)

- A.I.D.E. – Android IDE on Android

  Java+Android SDK/ C/C++ Android NDK

  https://play.google.com/store/apps/details?id=com.aide.ui&hl=sk

- Android Studio (IntelliJ iDEA)

  http://developer.android.com/sdk/installing/studio.html

  http://developer.android.com/tools/studio/index.html

- IntelliJ iDEA pluging for Android (**Professional Android IDE)**

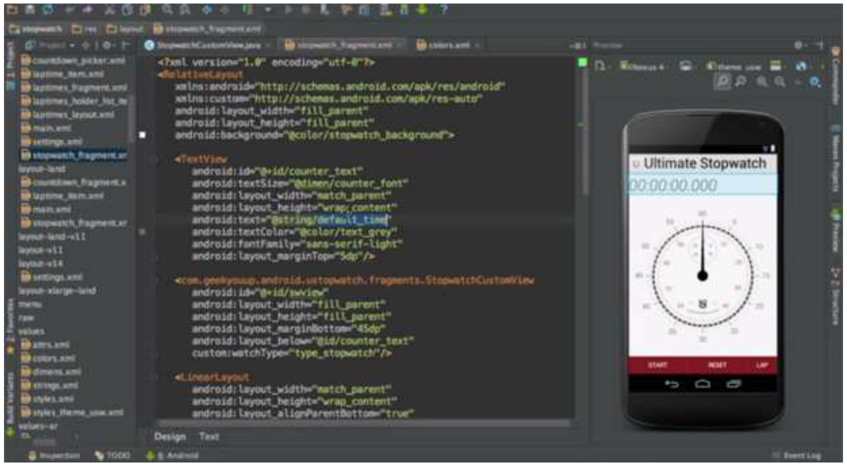
  http://www.jetbrains.com/idea/

- ADT plugin pre Eclipse

  Android Development Tool

  Eclipse ADT plugin is no longer supported

  Google ended official support in June 2015



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:custom="http://schemas.android.com/apk/res-auto"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="@color/stopwatch_background">

    <TextView
        android:id="@+id/counter_text"
        android:textSize="@dimen/counter_font"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/default_time"
        android:textColor="@color/text_grey"
        android:fontFamily="sans-serif-light"
        android:layout_marginTop="5dp"/>

    <com.geekyouup.android.ustopwatch.fragments.StopwatchCustomView
        android:id="@+id/swview"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:layout_marginBottom="45dp"
        android:layout_below="@id/counter_text"
        custom:watchType="type_stopwatch"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
```

Zdroje a Android Studio

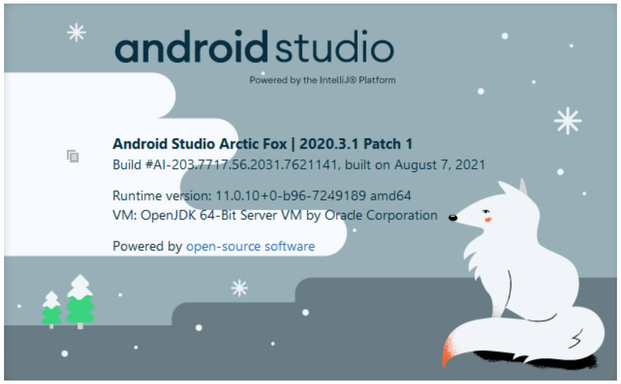
Android Studio a jeho eco-systém:

- Developer Android Forum (http://developer.android.com/)
- Stackoverflow (http://stackoverflow.com/)
- kotlin.org (https://kotlinlang.org/)
- iná literatúra (http://dai.fmph.uniba.sk/courses/VMA/android/pdfs/)

- (!) väčšinu odporúčaných kníh nájdete v našej knižnici ☹

univerzitná knižnica
v bratislave

Používame Android Studio 4.0 (Official IDE for Android)
https://developer.android.com/studio/index.html



android studio
Powered by the IntelliJ® Platform

Android Studio Arctic Fox | 2020.3.1 Patch 1
Build #AI-203.7717.56.2031.7621141, built on August 7, 2021

Runtime version: 11.0.10+0-b96-7249189 amd64
VM: OpenJDK 64-Bit Server VM by Oracle Corporation

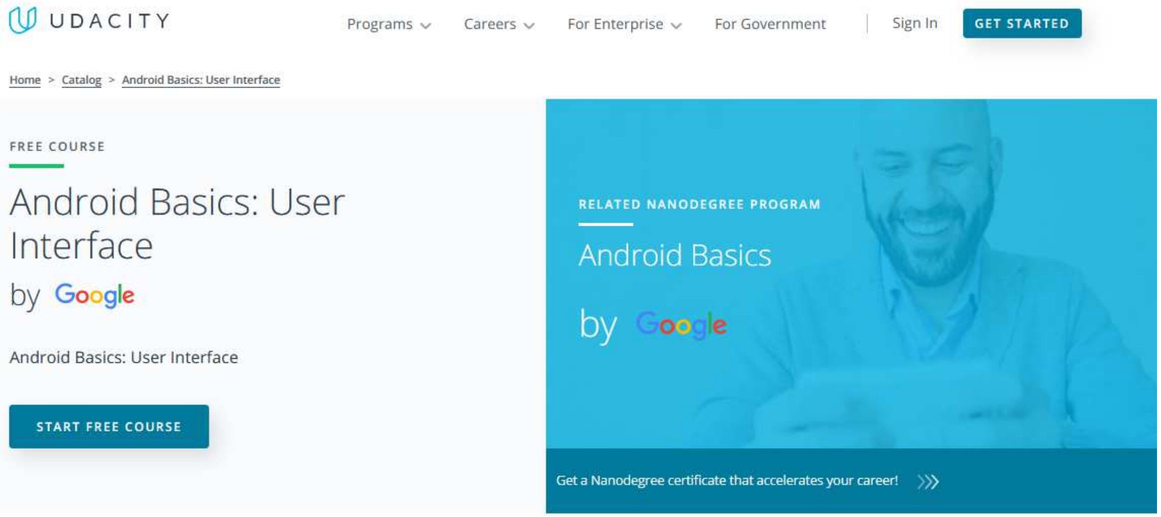
Powered by open-source software

Už obsahuje aj Kotlin
(1.5) support

UDACITY

Free Udacity courses
(alternatívne free online)

- user interface – začiatočník, user interface
- user input
- multiscreen apps – activities, fragments, master-detail view, …
- networking – http networking, json parsing, …
- data storage - sqlLite
- material design
- Google Firebase
- Kotlin for Android



Android Basics: User Interface

by Google

Android Basics: User Interface

Android Basics

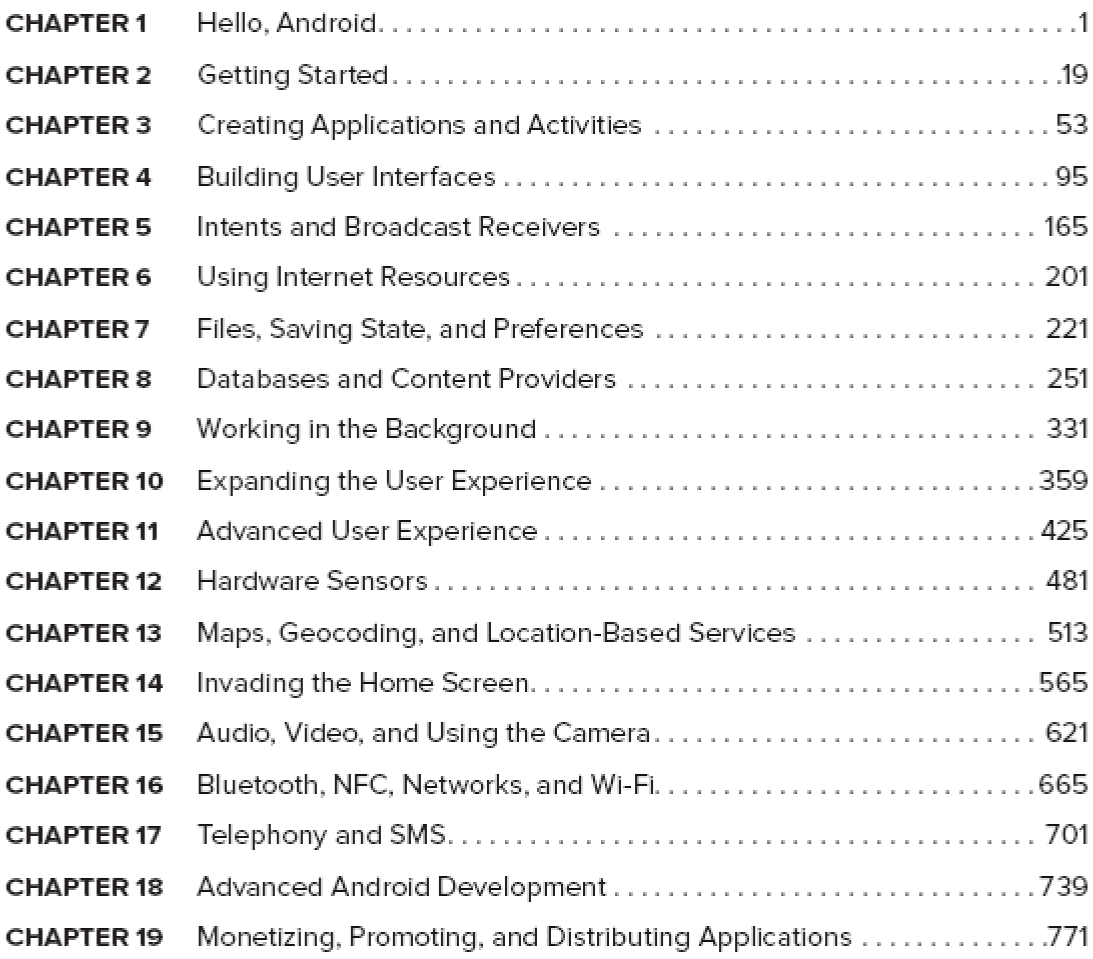
by Google

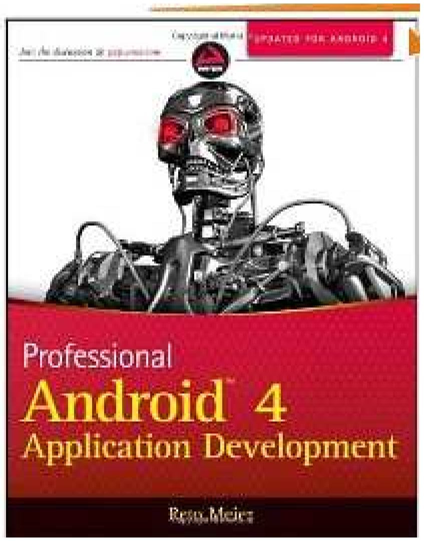
Get a Nanodegree certificate that accelerates your career!

Professional Android 4

(stará ale dobrá...java)

- 2012, Reto Meier, Amazon: 4/5

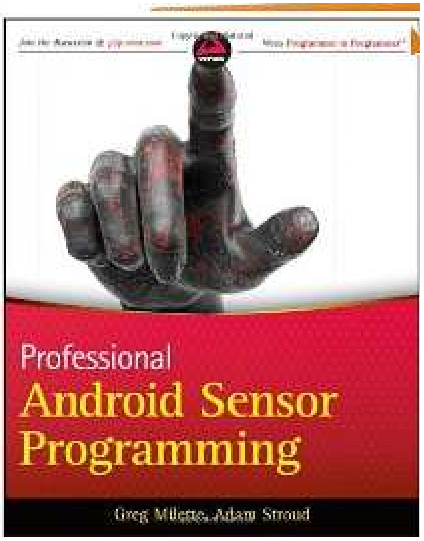




univerzitná knižnica
v bratislave

Android Sensor Programming

(stará ale dobrá...java)



2012, Greg Milette, Adam Stroud, Amazon: 5/5

PART I LOCATION SERVICES

CHAPTER 1 Introducing the Android Location Service
CHAPTER 2 Determining a Device's Current Location
CHAPTER 3 Tracking Device Movement
CHAPTER 4 Proximity Alerts

PART II INFERRING INFORMATION FROM PHYSICAL SENSORS

CHAPTER 5 Overview of Physical Sensors
CHAPTER 6 Errors and Sensor Signal Processing
CHAPTER 7 Determining Device Orientation
CHAPTER 8 Detecting Movement
CHAPTER 9 Sensing the Environment
CHAPTER 10 Android Open Accessor

PART III SENSING THE AUGMENTED, PATTERN-RICH EXTERNAL WORLD

CHAPTER 11 Near Field Communication (NFC)
CHAPTER 12 Using the Camera
CHAPTER 13 Image-Processing Techniques
CHAPTER 14 Using the Microphone

PART IV SPEAKING TO ANDROID

CHAPTER 15 Designing a Speech-Enabled App
CHAPTER 16 Using Speech Recognition and Text-To-Speech APIs
CHAPTER 17 Matching What Was Said
CHAPTER 18 Executing Voice Actions
CHAPTER 19 Implementing Speech Activation

Prefessional Android Application Development

(stará ale dobrá...java)

Wrox Programmer to Programmer™
Professional
Android™
Application
Development
Reto Meier
wrox
Updates, source code, and Wrox technical support at www.wrox.com

- 2014, Reto Meier, Amazon: 4/5

1. Hello, Android
2. Getting Started
3. Creating Applications and Activities
4. Creating User Interfaces
5. Intents, Broadcast Receivers, Adapters, and the Internet
6. Data Storage, Retrieval, and Sharing
7. Maps, Geocoding, and Location-Based Services
8. Working in the Background
9. Peer-to-Peer Communication
10. Accessing Android Hardware
11. Advanced Android Development

V knižnici FMFI

Bohužial len na prezenčnú výpožicku (t.j. len tam):
•Meier: Professional Android 4 Application
•Milette: Professional Android Sensor Programming
•Wii-Meng Lee: Beginning Android 4 Application Development

univerzitná knižnica
v bratislave

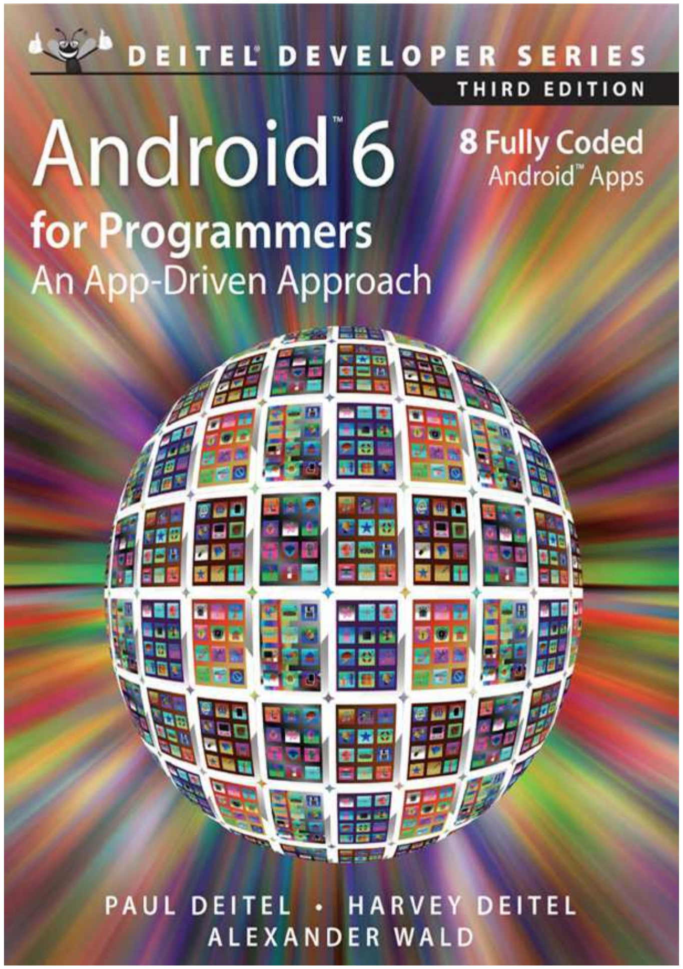
ANDROID™ 6 FOR PROGRAMMERS

AN APP-DRIVEN APPROACH

- Paul Deitel • Harvey Deitel • Alexander Wald, 2016, 3rd Edition, Amazon 4.4/5

8 konkrétnych appiek detailne vysvetlených

1. Welcome App
2. Cannon Game
3. Tip Calculator
4. Weather Viewer
5. Flag Quiz
6. Twitter® Searches
7. Doodlz
8. Address Book



Java vs. Kotlin

tradičný VMA kurz postavený na Java už druhý rok beží v jazyku Kotlin 1.3

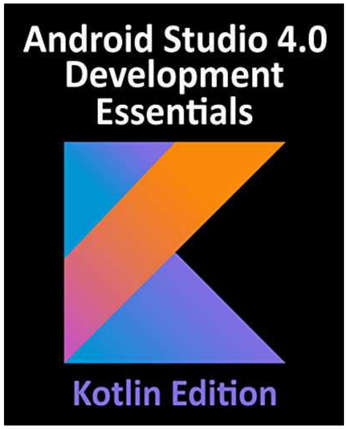
Dôvody:

- ako iOS má svoj moderný jazyk Swift (3/4/5), aj Android má svoj Kotlin
- Java je trochu *skamenelina* medzi modernými jazykmi (Swift, Kotlin, Scala)
- Kotlin je Googlom oficiálne podporovaným vývojový nástroj pre Android
- projekt Kotlin má už >10 rokov
- kompiluje do JVM
- funguje s Android Studiom
- na JetBrains produkty ste si asi zvykli, a sú top
- oboznámite sa s niektorými princípmi moderných jazykov

- Reference: https://kotlinlang.org/docs/reference/
- Online: https://play.kotlinlang.org/byExample/

Android Studio 4.* Development Essentials – Kotlin Edition

- https://www.amazon.com/Android-Studio-4-0-Development-Essentials-ebook/dp/B089T8Z66P
- sources: https://www.ebookfrenzy.com/retail/as40kotlin/page.php



Inštalácia Android Studio (4.0):

https://developer.android.com/studio

Predmet má cvičenie, ale aj tak:

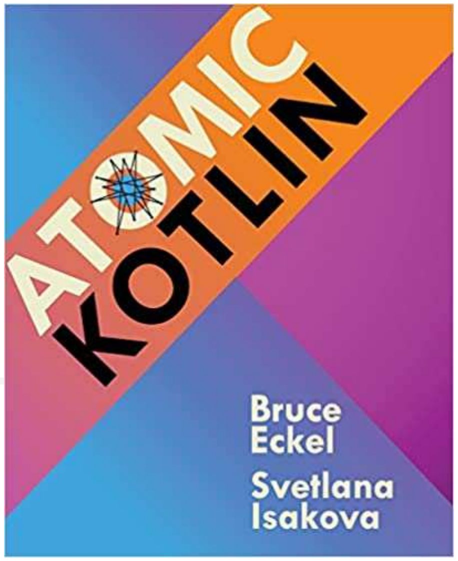
- ozvite sa v prípade problémov inštalácie na platformy, napr. Linux, Mac.
- Dominika, Jožo, Lukáš, ja sa vám posnažíme problém vyriešiť

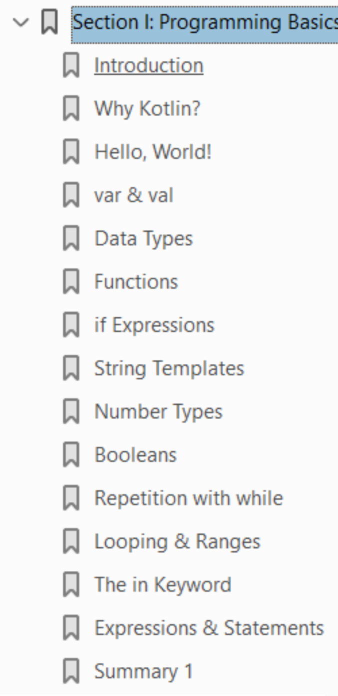
Inštalácia Android Studia:

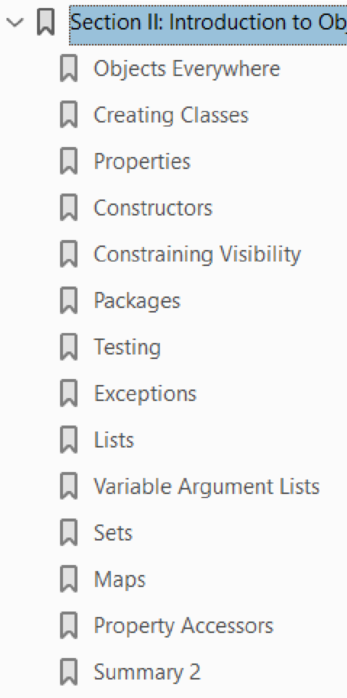
- SDK Packages: Tools/SDK Manager tab SDK Platforms
- AVD: Android Virtual Device

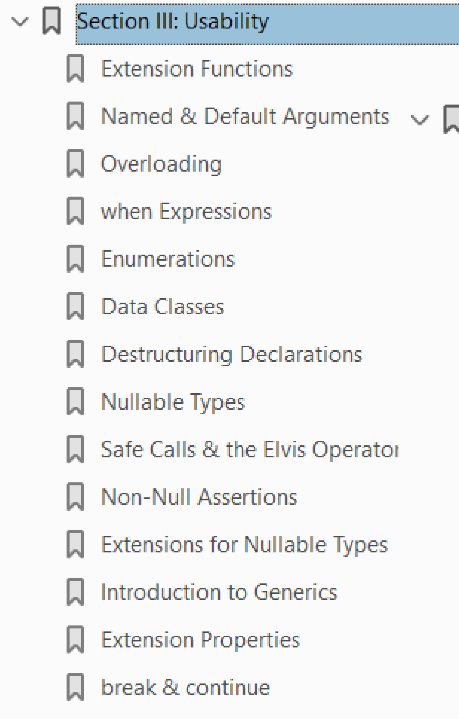
Atomic Kotlin

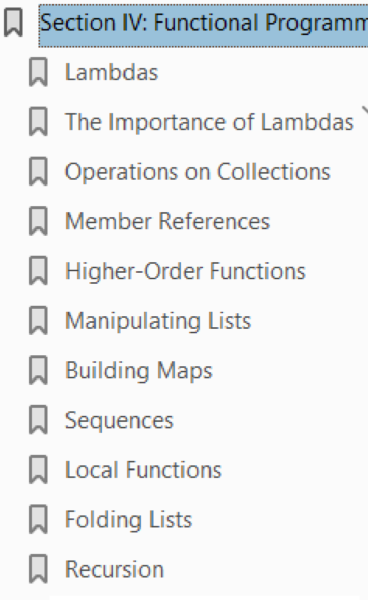
https://www.amazon.com/Atomic-Kotlin-Bruce-Eckel/dp/0981872557

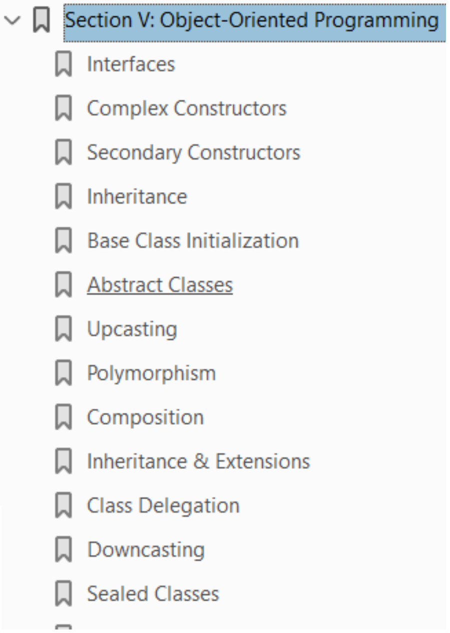








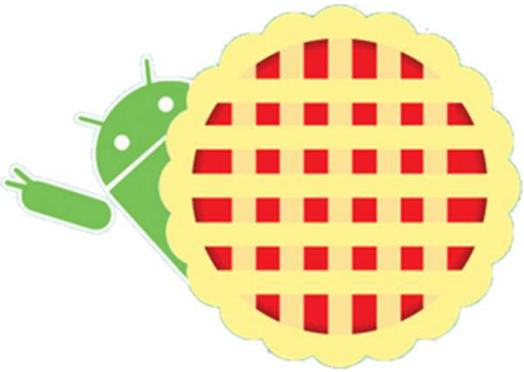




Why Teach Kotlin

- Kotlin is basically becoming the language of Android.
- Decrease in boilerplate helps us to quickly identify which fundamental Android concepts students are missing.
- On numerous courses, where we proceed through Java to Kotlin, we are considering a Kotlin-first approach.
- Students are happy to have the chance to program in something they may have heard about.
- I think our students benefit in general from being exposed to a wide range of programming languages, and I think it is valuable for them to gain experience in using more modern languages alongside the more traditional ones like Java and C++.
- My Kotlin students in fact understand OO concepts better than my Java students do.
- One of Kotlin's advantages is a good combination of strong typing and nullability.

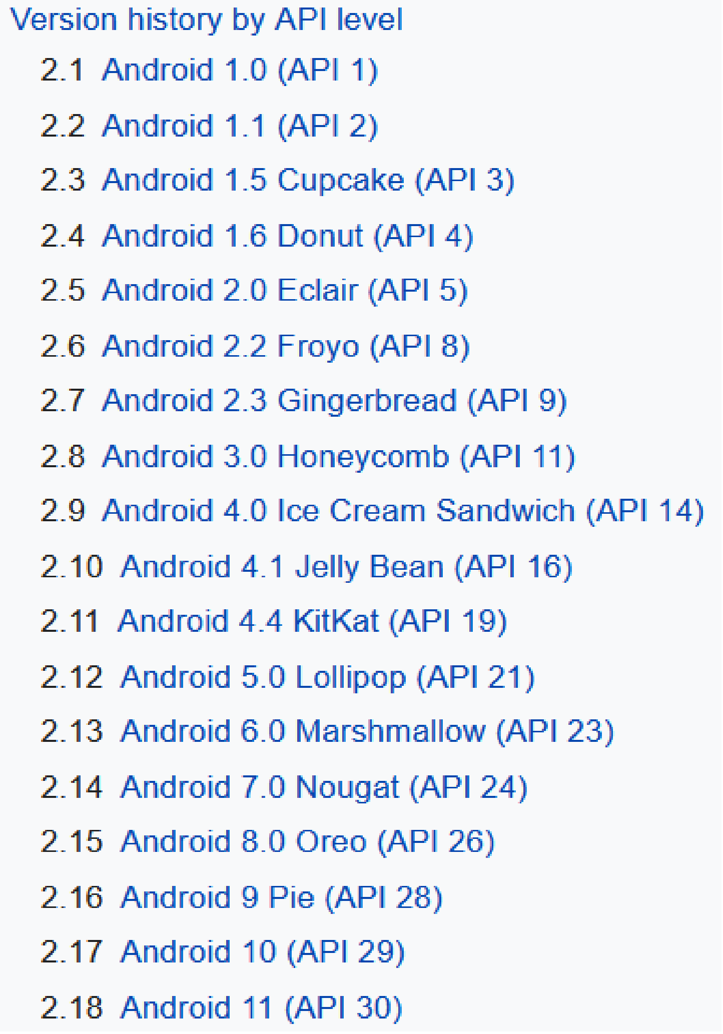
https://kotlinlang.org/education/why-teach-kotlin.html



Android a Google

- 2005 Google acquired Android Inc. with Rubin, Miner et at.
- 2007 Open Handset Alliance, a consortium
  - device manufacturers: HTC, Sony and Samsung,
  - wireless carriers: T-Mobile, …
  - chipset makers: Qualcomm, Texas Instruments,

  includes Google with a goal to develop open standards for mobile devices
- major release named in alphabetical order after a dessert or sugary treat
  - 2.3 Gingerbread
  - 4.3 *Jelly Bean, July, 2012,*
  - 4.4 *KitKat*, announced, October, 2013,
  - 5.1 Lollipop, November, 2014,
  - 6.0 Marsmallow, October, 2015,
  - 7.0 Nougat, August, 2016.
  - 8.0 Oreo, August, 2017,
  - 9.0 Pie, August, 2018,
  - 10.0 Android 10, September 2019
  - 11.0 Android 11, tba



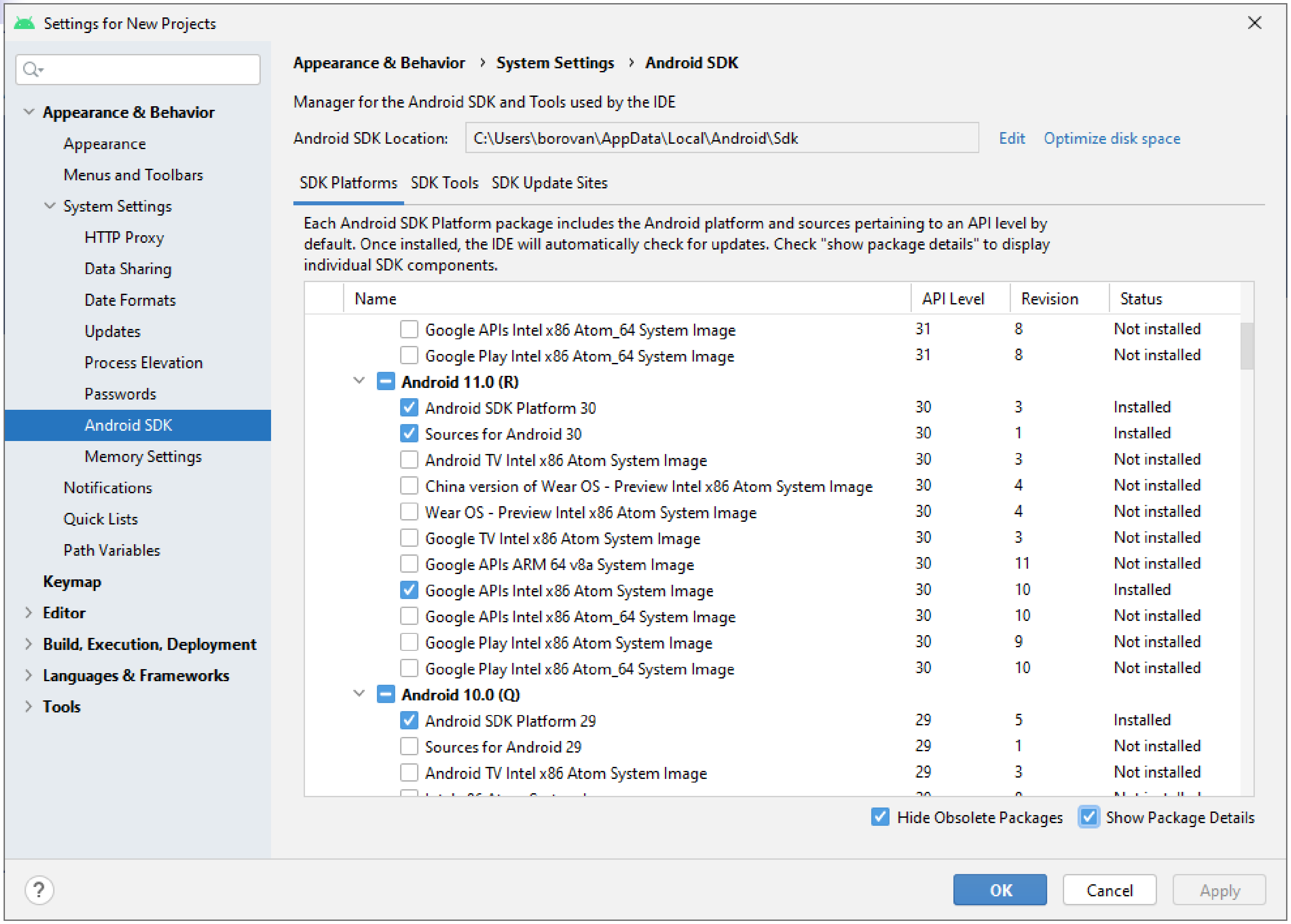
Version history by API level

2.1 Android 1.0 (API 1)
2.2 Android 1.1 (API 2)
2.3 Android 1.5 Cupcake (API 3)
2.4 Android 1.6 Donut (API 4)
2.5 Android 2.0 Eclair (API 5)
2.6 Android 2.2 Froyo (API 8)
2.7 Android 2.3 Gingerbread (API 9)
2.8 Android 3.0 Honeycomb (API 11)
2.9 Android 4.0 Ice Cream Sandwich (API 14)
2.10 Android 4.1 Jelly Bean (API 16)
2.11 Android 4.4 KitKat (API 19)
2.12 Android 5.0 Lollipop (API 21)
2.13 Android 6.0 Marshmallow (API 23)
2.14 Android 7.0 Nougat (API 24)
2.15 Android 8.0 Oreo (API 26)
2.16 Android 9 Pie (API 28)
2.17 Android 10 (API 29)
2.18 Android 11 (API 30)

http://en.wikipedia.org/wiki/Android_version_history

Android SDK Packages

Tools/SDK Manager tab SDK Platforms - API 30



Appearance & Behavior › System Settings › Android SDK

Manager for the Android SDK and Tools used by the IDE

Android SDK Location: C:\Users\borovan\AppData\Local\Android\Sdk

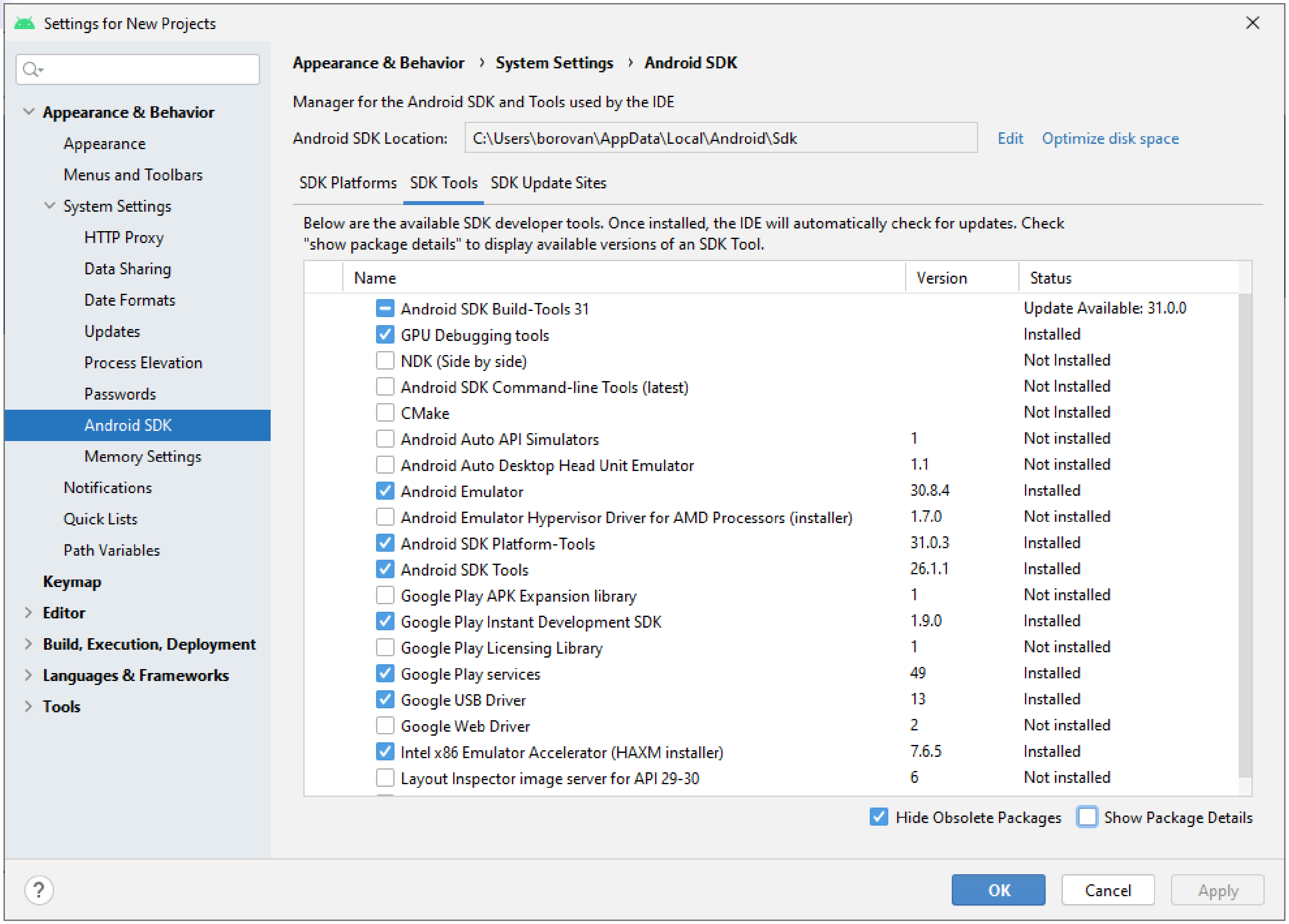
Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, the IDE will automatically check for updates. Check "show package details" to display individual SDK components.

Name	API Level	Revision	Status
□ Google APIs Intel x86 Atom_64 System Image	31	8	Not installed
□ Google Play Intel x86 Atom_64 System Image	31	8	Not installed
Android 11.0 (R)			
☑ Android SDK Platform 30	30	3	Installed
☑ Sources for Android 30	30	1	Installed
□ Android TV Intel x86 Atom System Image	30	3	Not installed
□ China version of Wear OS - Preview Intel x86 Atom System Image	30	4	Not installed
□ Wear OS - Preview Intel x86 Atom System Image	30	4	Not installed
□ Google TV Intel x86 Atom System Image	30	3	Not installed
□ Google APIs ARM 64 v8a System Image	30	11	Not installed
☑ Google APIs Intel x86 Atom System Image	30	10	Installed
□ Google APIs Intel x86 Atom_64 System Image	30	10	Not installed
□ Google Play Intel x86 Atom System Image	30	9	Not installed
□ Google Play Intel x86 Atom_64 System Image	30	10	Not installed
Android 10.0 (Q)			
☑ Android SDK Platform 29	29	5	Installed
□ Sources for Android 29	29	1	Not installed
□ Android TV Intel x86 Atom System Image	29	3	Not installed

☑ Hide Obsolete Packages ☑ Show Package Details

Android SDK Packages

Tools/SDK Manager tab SDK Tools



Appearance & Behavior › System Settings › Android SDK

Manager for the Android SDK and Tools used by the IDE

Android SDK Location: C:\Users\borovan\AppData\Local\Android\Sdk

Below are the available SDK developer tools. Once installed, the IDE will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.

Name	Version	Status
⊟ Android SDK Build-Tools 31		Update Available: 31.0.0
☑ GPU Debugging tools		Installed
☐ NDK (Side by side)		Not Installed
☐ Android SDK Command-line Tools (latest)		Not Installed
☐ CMake		Not Installed
☐ Android Auto API Simulators	1	Not installed
☐ Android Auto Desktop Head Unit Emulator	1.1	Not installed
☑ Android Emulator	30.8.4	Installed
☐ Android Emulator Hypervisor Driver for AMD Processors (installer)	1.7.0	Not installed
☑ Android SDK Platform-Tools	31.0.3	Installed
☑ Android SDK Tools	26.1.1	Installed
☐ Google Play APK Expansion library	1	Not installed
☑ Google Play Instant Development SDK	1.9.0	Installed
☐ Google Play Licensing Library	1	Not installed
☑ Google Play services	49	Installed
☑ Google USB Driver	13	Installed
☐ Google Web Driver	2	Not installed
☑ Intel x86 Emulator Accelerator (HAXM installer)	7.6.5	Installed
☐ Layout Inspector image server for API 29-30	6	Not installed

☑ Hide Obsolete Packages ☐ Show Package Details

Android Virtual Device

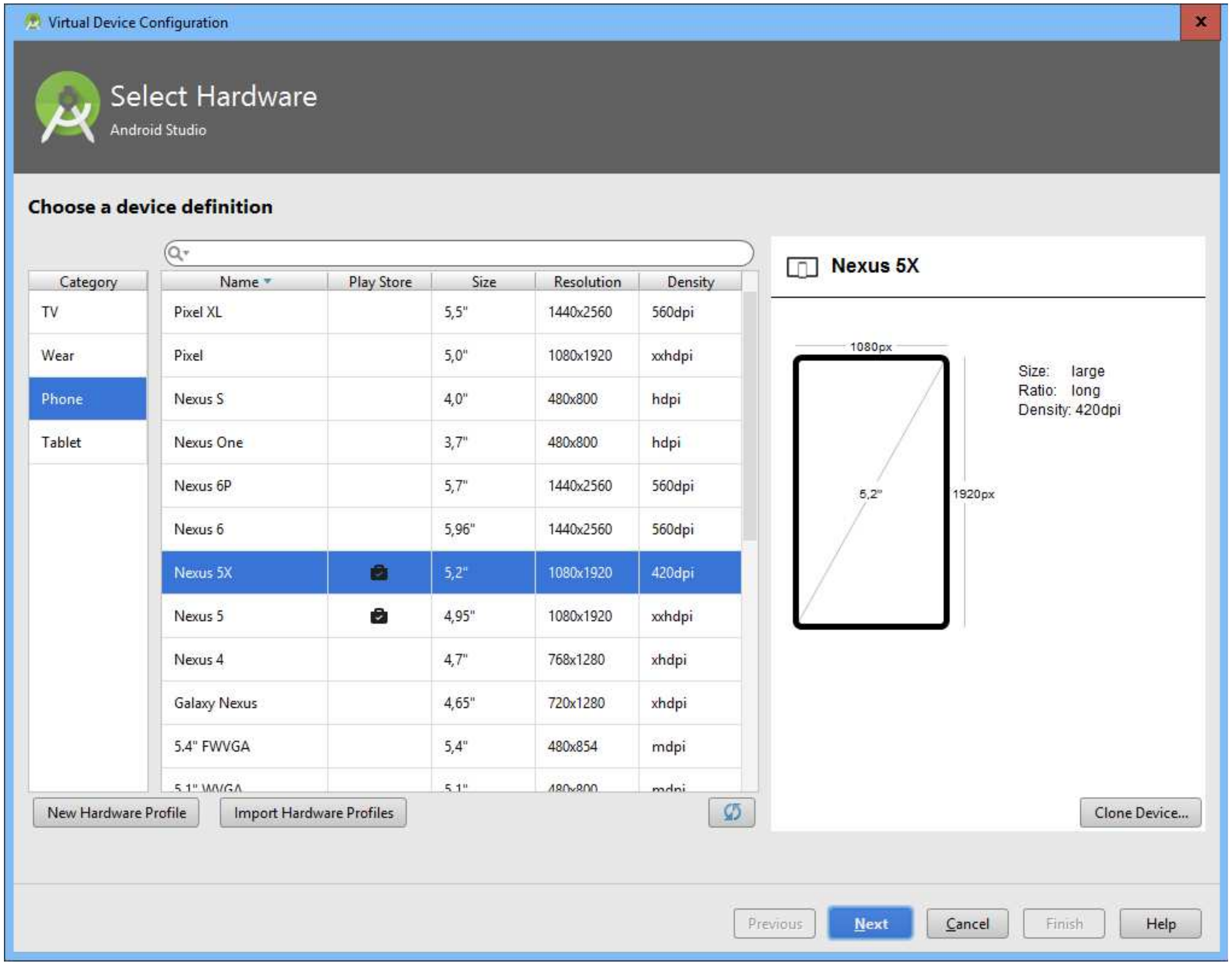
Tools/AVD manager

Nakonfigurujte si AVD zodpovedajúci vášmu zariadeniu

alebo si vyberte zo zoznamu predvolených,

- Create Device
- modifikujte

nastavenia podľa vášho zariadenia



Virtual Device Configuration

Select Hardware

Android Studio

Choose a device definition

Category
TV
Wear
Phone
Tablet

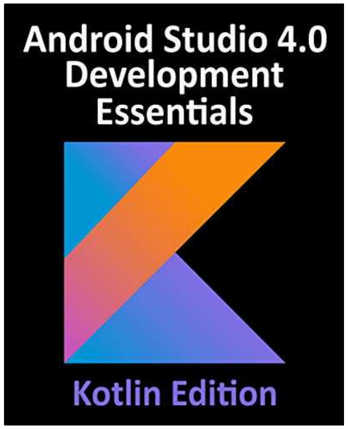
Name	Play Store	Size	Resolution	Density
Pixel XL		5,5"	1440x2560	560dpi
Pixel		5,0"	1080x1920	xxhdpi
Nexus S		4,0"	480x800	hdpi
Nexus One		3,7"	480x800	hdpi
Nexus 6P		5,7"	1440x2560	560dpi
Nexus 6		5,96"	1440x2560	560dpi
Nexus 5X	✓	5,2"	1080x1920	420dpi
Nexus 5	✓	4,95"	1080x1920	xxhdpi
Nexus 4		4,7"	768x1280	xhdpi
Galaxy Nexus		4,65"	720x1280	xhdpi
5.4" FWVGA		5,4"	480x854	mdpi
5.1" WVGA		5,1"	480x800	mdpi

Nexus 5X

Size: large
Ratio: long
Density: 420dpi

Android Studio 4.* Development Essentials – Kotlin Edition

- https://www.amazon.com/Android-Studio-4-0-Development-Essentials-ebook/dp/B089T8Z66P
- sources: https://www.ebookfrenzy.com/retail/as40kotlin/page.php



Inštalácia Android Studio (4.0):
https://developer.android.com/studio

Kapitola 2. Setting up an Android Studio Development Environment
(mac/Windows/Linux)

Kapitola 3. Creating an Example Android App in AS

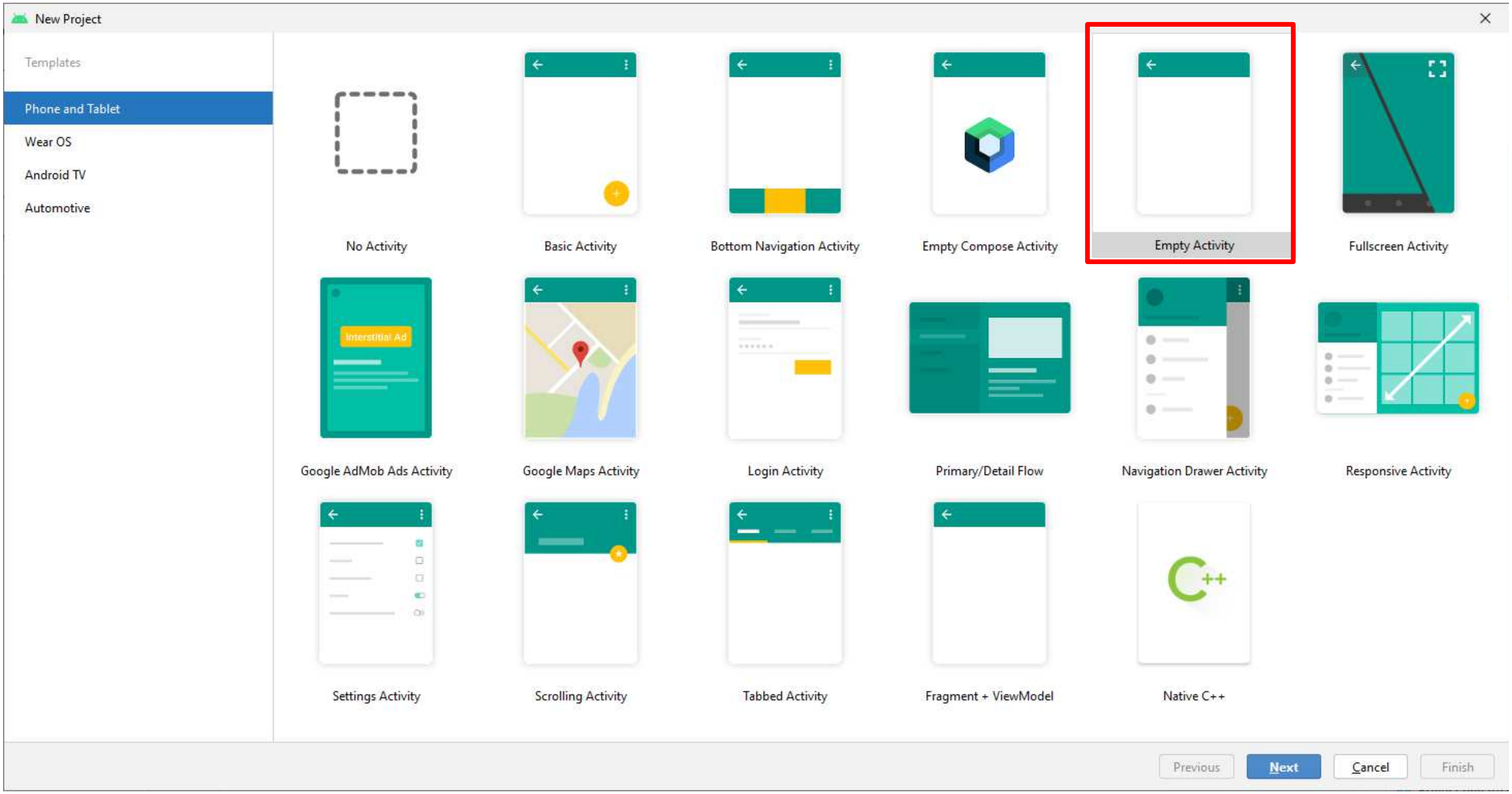
Kapitola 4. Creating an Android Virtual Device (AVD) in AS

Kapitola 5. Using and Configuring the Android Studio AVD Emulator

Kapitola 6. A tour of the Android Studio User Interface

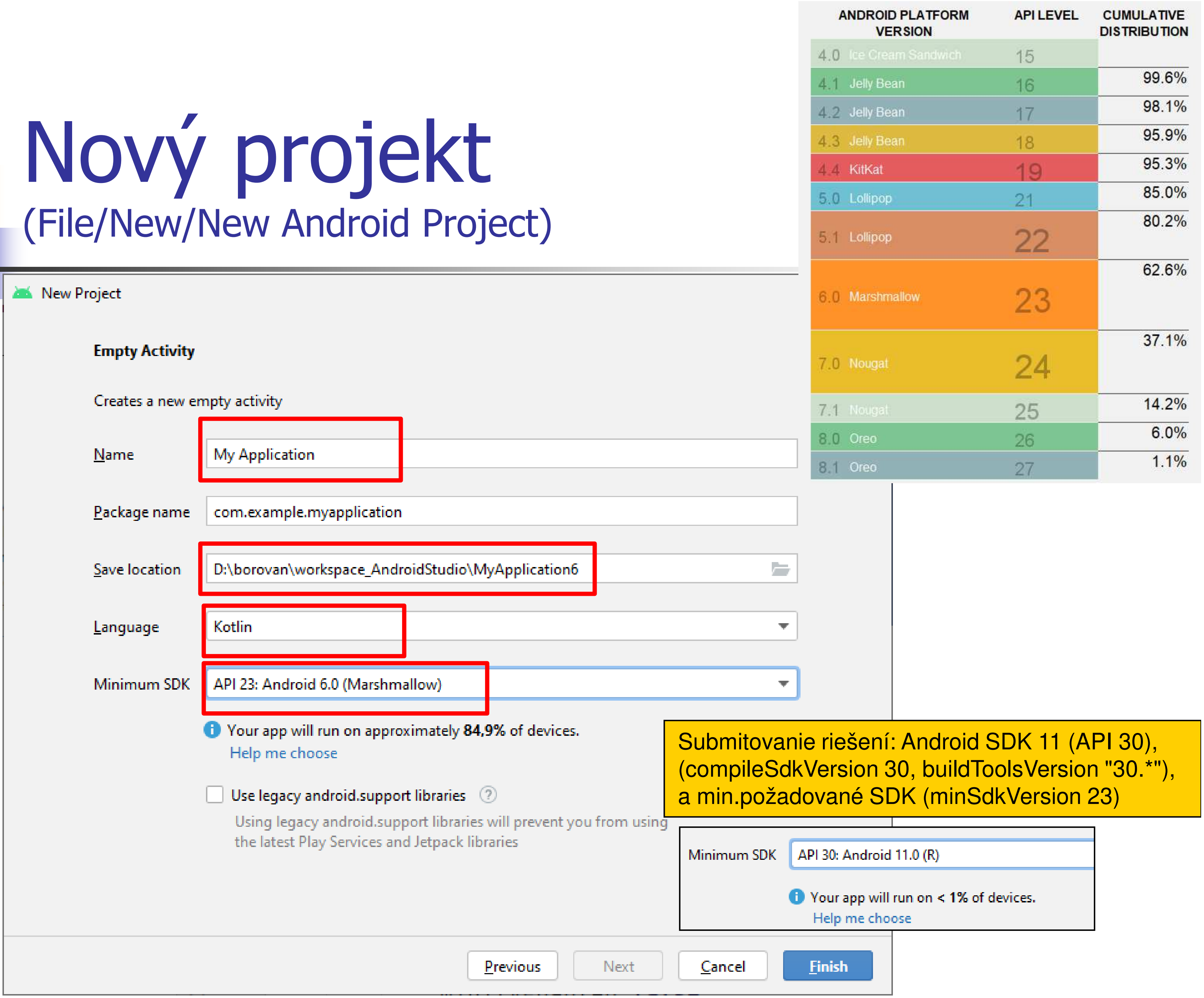
Kapitola 7. Testing Android Studio App on a Physical Android Device

Nový projekt
(File/New/New Android Project)



Nový projekt

(File/New/New Android Project)



ANDROID PLATFORM VERSION		API LEVEL	CUMULATIVE DISTRIBUTION
4.0	Ice Cream Sandwich	15	
4.1	Jelly Bean	16	99.6%
4.2	Jelly Bean	17	98.1%
4.3	Jelly Bean	18	95.9%
4.4	KitKat	19	95.3%
5.0	Lollipop	21	85.0%
5.1	Lollipop	22	80.2%
6.0	Marshmallow	23	62.6%
7.0	Nougat	24	37.1%
7.1	Nougat	25	14.2%
8.0	Oreo	26	6.0%
8.1	Oreo	27	1.1%

New Project

Empty Activity

Creates a new empty activity

Name: My Application

Package name: com.example.myapplication

Save location: D:\borovan\workspace_AndroidStudio\MyApplication6

Language: Kotlin

Minimum SDK: API 23: Android 6.0 (Marshmallow)

Your app will run on approximately **84,9%** of devices.
Help me choose

□ Use legacy android.support libraries

Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

Submitovanie riešení: Android SDK 11 (API 30), (compileSdkVersion 30, buildToolsVersion "30.*"), a min.požadované SDK (minSdkVersion 23)

Minimum SDK: API 30: Android 11.0 (R)

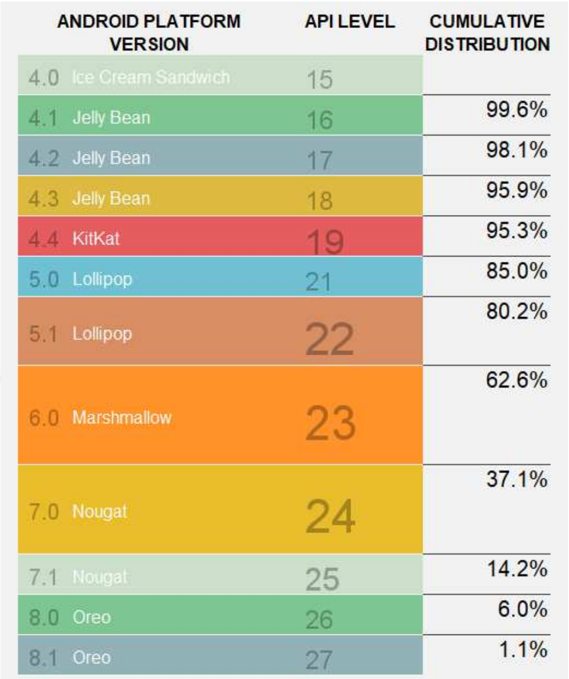
Your app will run on < **1%** of devices.
Help me choose

Nový projekt

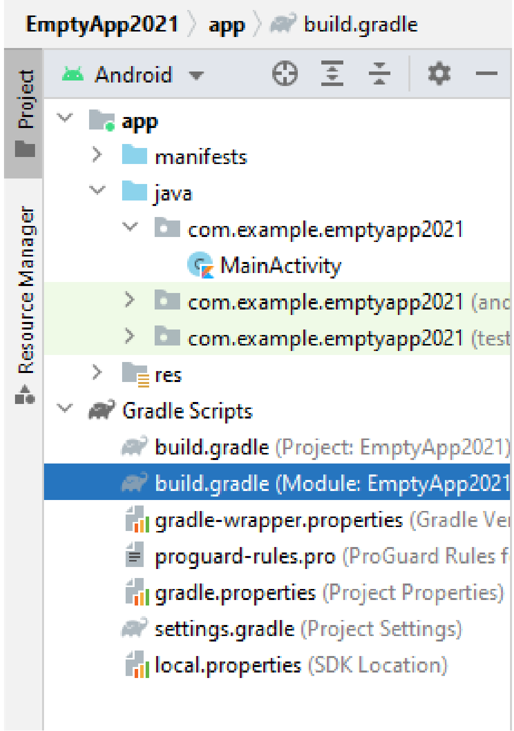
(File/New/New Android Project)

```
plugins {
    id 'com.android.application'
    id 'kotlin-android'
}

android {
    compileSdk 30

    defaultConfig {
        applicationId "com.example.emptyapp2021"
        minSdk 23
        targetSdk 30
        versionCode 1
        versionName "1.0"
    }
```

Submitovanie riešení: Android SDK 11 (API 30), (compileSdkVersion 30, buildToolsVersion "30.*"), a min.požadované SDK (minSdkVersion 23)



ANDROID PLATFORM VERSION		API LEVEL	CUMULATIVE DISTRIBUTION
4.0	Ice Cream Sandwich	15	
4.1	Jelly Bean	16	99.6%
4.2	Jelly Bean	17	98.1%
4.3	Jelly Bean	18	95.9%
4.4	KitKat	19	95.3%
5.0	Lollipop	21	85.0%
5.1	Lollipop	22	80.2%
6.0	Marshmallow	23	62.6%
7.0	Nougat	24	37.1%
7.1	Nougat	25	14.2%
8.0	Oreo	26	6.0%
8.1	Oreo	27	1.1%

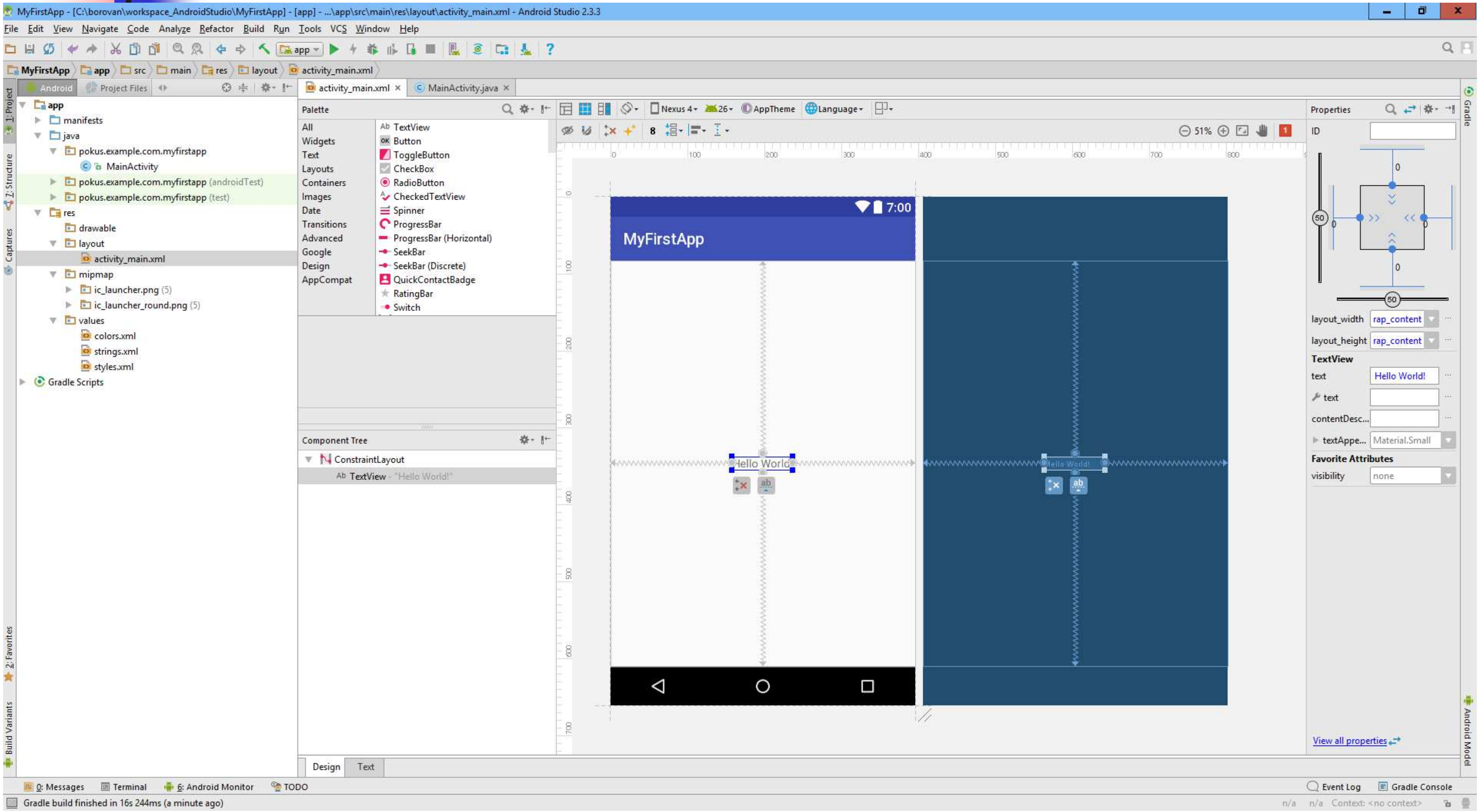


EmptyApp2021 〉 app 〉 build.gradle

```
app
    manifests
    java
        com.example.emptyapp2021
            MainActivity
        com.example.emptyapp2021 (anc
        com.example.emptyapp2021 (test
    res
Gradle Scripts
    build.gradle (Project: EmptyApp2021)
    build.gradle (Module: EmptyApp2021
    gradle-wrapper.properties (Gradle Ve
    proguard-rules.pro (ProGuard Rules f
    gradle.properties (Project Properties)
    settings.gradle (Project Settings)
    local.properties (SDK Location)
```

Nový projekt

(java)



MyFirstApp - [C:\borovan\workspace_AndroidStudio\MyFirstApp] - [app] - ...\app\src\main\res\layout\activity_main.xml - Android Studio 2.3.3

ID

layout_width wrap_content

layout_height wrap_content

TextView

text Hello World!

text

contentDesc...

textAppe... Material.Small

Favorite Attributes

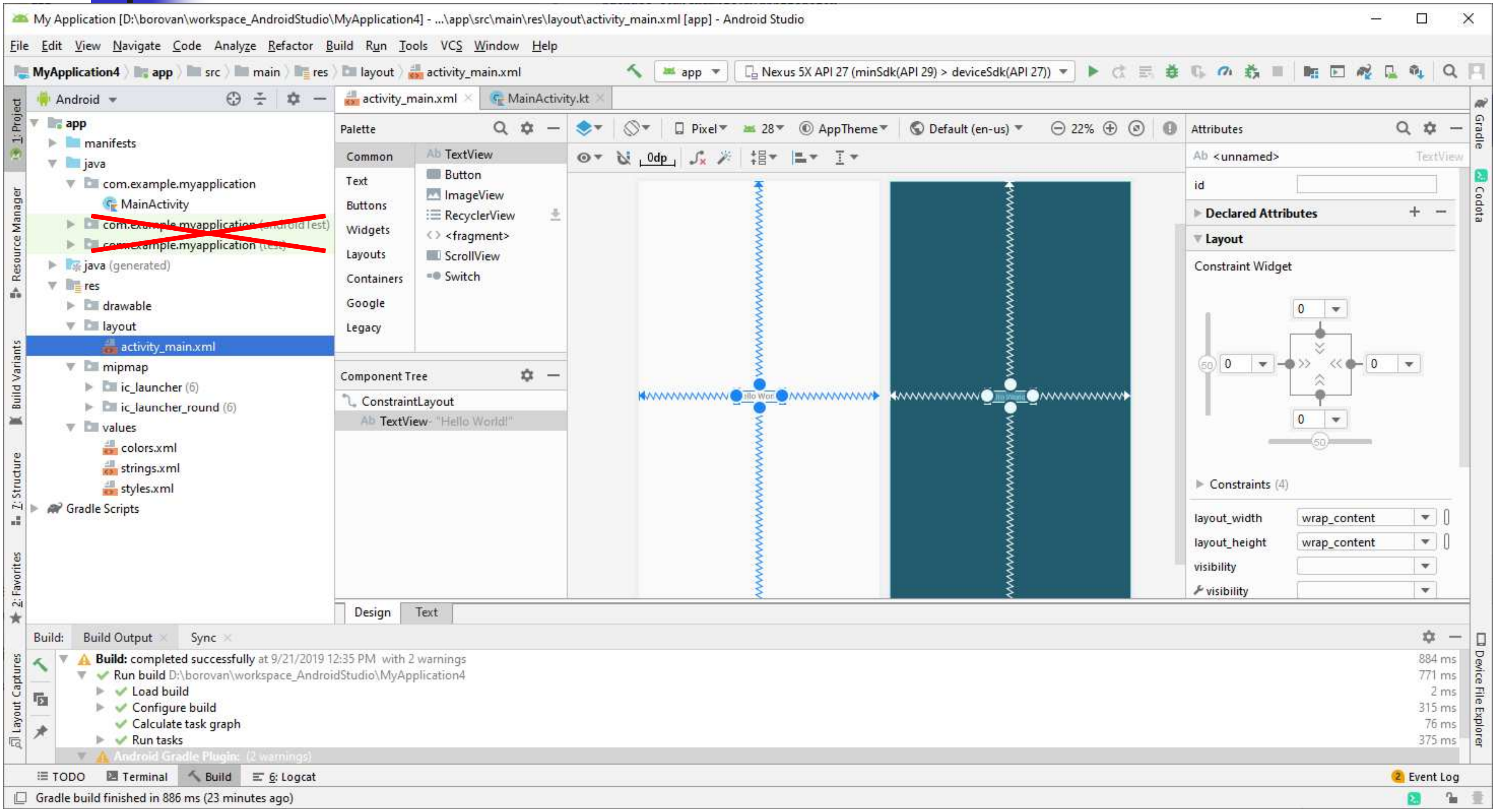
visibility none

Gradle build finished in 16s 244ms (a minute ago)

n/a n/a Context: <no context>

Nový projekt

(kotlin)



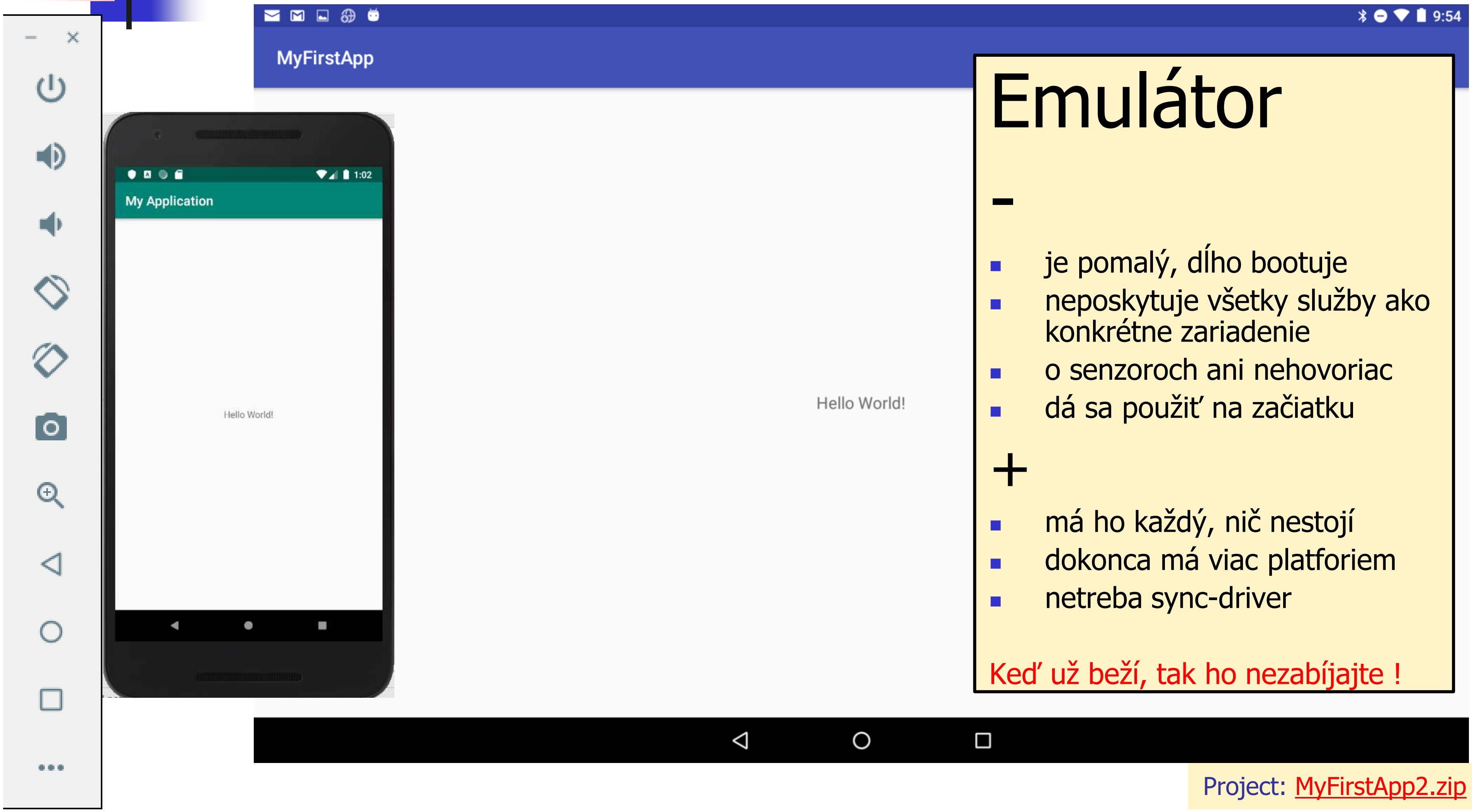
My Application [D:\borovan\workspace_AndroidStudio\MyApplication4] - ...\app\src\main\res\layout\activity_main.xml [app] - Android Studio

Build: Build Output × Sync ×

```
Build: completed successfully at 9/21/2019 12:35 PM with 2 warnings     884 ms
  Run build D:\borovan\workspace_AndroidStudio\MyApplication4            771 ms
    Load build                                                           2 ms
    Configure build                                                      315 ms
    Calculate task graph                                                 76 ms
    Run tasks                                                            375 ms
  Android Gradle Plugin: (2 warnings)
```

Gradle build finished in 886 ms (23 minutes ago)

Pýtajte sa
kým nedostanete



Emulátor

-

- je pomalý, dĺho bootuje
- neposkytuje všetky služby ako konkrétne zariadenie
- o senzoroch ani nehovoriac
- dá sa použiť na začiatku

+

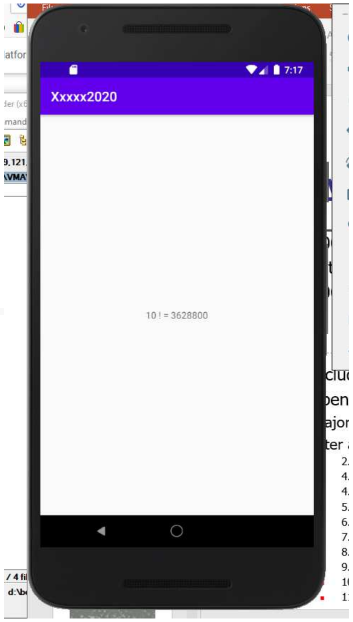
- má ho každý, nič nestojí
- dokonca má viac platforiem
- netreba sync-driver

Keď už beží, tak ho nezabíjajte !

Project: MyFirstApp2.zip

Ako si skúšať Kotlin v AS

(kým sa nedozvieme viac)

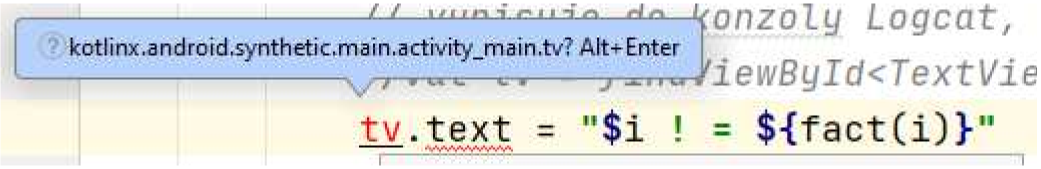


Xxxxx2020

10 ! = 3628800

```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        //println(fact(10))
        for (i in 0..10) {
            Log.d("TAG", "$i ! = ${fact(i)}")
            // vypisuje do konzoly Logcat, pouzite filter s "TAG"
            val tv = findViewById<TextView>(R.id.tv)
            tv.text = "$i ! = ${fact(i)}"
            // vypise do View komponentu, ktory je v Activite
            Toast.makeText(this, "$i ! = ${fact(i)}",
            Toast.LENGTH_SHORT).show()
            // Toast alias Notifier (MITI)
        }
    }
    fun fact(n : Int) : Int = if (n == 0) 1 else n * fact(n-1)
}
```

Integrovanie Android Extensions



```
// vypisuje do konzoly Logcat,
//val tv = findViewById<TextVie
tv.text = "$i ! = ${fact(i)}"
```

kotlinx.android.synthetic.main.activity_main.tv? Alt+Enter

```
plugins {
   id 'com.android.application'
   id 'kotlin-android'
   id 'kotlin-android-extensions'
}
```

```
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log
import android.widget.TextView
import android.widget.Toast
import kotlinx.android.synthetic.main.activity_main.*
```

```
for (i in 0..10) {
    Log.d("TAG", "$i ! = ${fact(i)}")
    // vypisuje do konzoly Logcat, pouzite filter s "TAG"
    ~~val tv = findViewById<TextView>(R.id.tv)~~
    tv.text = "$i ! = ${fact(i)}"
    // vypise do View komponentu, ktory je v Activite
    Toast.makeText(this, "$i ! = ${fact(i)}",
    Toast.LENGTH_SHORT).show()
    // Toast alias Notifier (MITI)
}
```

Break point

(štruktúrou projektu pokračujeme na budúce)

- Switch to kotlin intro