

Android Studio

(ako začať')

Jazyk Kotlin

(ako neskončiť)



Peter Borovanský
KAI, I-18

MS-Teams: [2sf3ph4](#), [List](#), [github](#)

borovan 'at' ii.fmph.uniba.sk

Základné info o kurze

Stránka predmetu

- <https://dai.fmph.uniba.sk/courses/VMA/>

prihláste sa do [L.I.S.T.](#)

- ak ste v ňom nikdy neboli, ozvite sa mi mailom
- sledujte LIST, všetky zadania budú v ňom
- sledujte Teams [2sf3ph4](#), komunikácia/prednášky/oznamy



veľká časť kurzu bude dobre sledovateľná z knihy

- **Android Studio Koala Essentials - Kotlin Edition: Developing Android Apps Using Android Studio 2025.1.2 and Kotlin , Neil Smyth** <https://www.payloadbooks.com/product/android-studio-koala-essentials-kotlin-edition-ebook/>
- .pdf pre minuloročnú verziu Giraffe je k dispozícii... <https://www.amazon.com/Android-Studio-Giraffe-Essentials-Developing/dp/1951442776>
- **Android Studio Narwhal Essentials - Compose Edition: Developing Android Apps with Android Studio, Jetpack Compose, and Kotlin,** <https://www.amazon.com/Android-Studio-Narwhal-Essentials-Developing>

Vývojové jazyky/nástroje

Symbian

- C++, Java ME, Python, ...

■ Windows Mobile 6

- C# (MS Visual Studio)

■ iOS

- Objective-C -> Swift 3/4/5 (Xcode)

iOS - Apple Center kurz

■ Android

- scratch (MIT Inventor)

- java (Android SDK + plugin pre Eclipse) -> (Android Studio)

- java (A.I.D.E.)

- Kotlin (Android Studio 4+)

- C++ (Android NDK)

■ Multi-platform

- C# (Xamarin iOS, Android, Windows) – fy. MS, Visual Studio 2015

- Pascal (Delphi XE5 iOS, Android, Windows 10) – fy. Embarcadero

- JavaScript/TypeScript (React Native)

- Flutter od Google

■ game engine

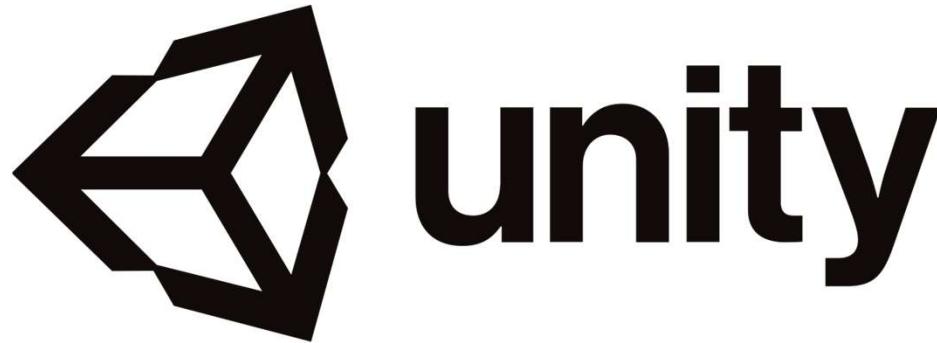
- C# (Unity 2D/3D), C++ (Unreal Engine)

1-AIN-303/24

...



Unity
(game engine)



#1 Unreal Engine

#2 Unity

Oblúbený nástroj pre tvorbu multi-platform aplikácií pre bakalárské práce

- 2-INF-263/15 magisterský predmet: Tvorba a dizajn počítačových hier
 - <http://sccg.sk/~mferko/tdh/>
 - <https://candle.fmph.uniba.sk/ucitelia/Michal-Ferko>

- 1-AIN-303/24 bakalársky predmet: Game Engines
 - Šimko (Gajdošech)

Vývoj a nástroje

(detailnejšie)

natívne aplikácie

- Android 
 - Java
 - Kotlin
- iOS 
 - Objective-C
 - Swift



Priamy prístup k všetkým fičúrkam a komponentom OS, aj tým najnovším ...

- hybridné aplikácie
 - Cordova
 - ionic



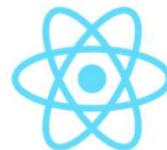
Web-app na báze .html, .css, .js, ktoré púšťame v prostredí WebView=browser/wrapper bez browserových ovládačov

Vývoj a nástroje

(detailnejšie)

■ kompilované aplikácie

- React Native
 - JavaScript
 - Facebook
 - nekompiluje do natívneho kódu
 - obmedzená množina widgets
- Flutter (Dart framework od Googlu)
 - Dart
 - Google
 - kompilované do ARM C++
 - bohatšia množina widgets
 - Material design (Quantum Paper)– Google 2014
- NativeScript
 - JavaScript



QUANTUM
and the building blocks of a unified interface



Aspekty programátora

Code sharing (write once, use everywhere)

- Cordova, ionic 
- Flutter (Material Design) 
- ReactNative 
- Java, Swift 

Knowledge sharing (learn once, use everywhere)

- Cordova, ionic, ReactNative (.js), Flutter (Dart) 
- Java, Kotlin, Swift 

Widget library

- Java, Swift, Cordova, ionic 
- Flutter (Dart) 
- ReactNative (.js)  <https://www.youtube.com/watch?v=bnYJRYFsrSw&feature=youtu.be>

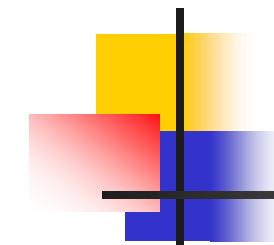
Aspekty programátora

Eco-system (schopnosť nájst' riešenie/radu/blog na stackoverflow,...)

- Java, Swift, Kotlin 
- Cordova, ionic (.js) 
- ReactNative (.js, React) 
- Flutter (nové ale zlepšuje sa) 

Popularita

- Java, Swift, Kotlin 
- Cordova, ionic, ReactNative, Flutter (pushujú FB a Google) 



Kotlin Multiplatform

- *KMM* is an SDK designed to simplify creating cross-platform mobile applications (Android, iOS iPhone, watchOS, Windows, Linux)
- share common code between iOS and Android apps
- write platform-specific code
- platforms jvm, js, wasm
- xcode (mac) is necessary to build an iOS app

Android Studio



AA Shakil
Flutter sucks too. 1

Páčí sa mi to · Odpovedat · Zdieľať · 3 d.



Daniel Peraza
XCode is much worse 8

Páčí sa mi to · Odpovedat · Zdieľať · 4 d.



Eloy Hunter-Bruckhoff
Daniel Peraza for Android development, yes

Páčí sa mi to · Odpovedat · Zdieľať · 3 d.



imgfja.com

ADMIT IT
ANDROID STUDIO IS THE WORST
EXPERIENCE A DEVELOPER CAN EVER HAVE



Marko Bašelj
Daniel Peraza what do people use instead of xcode?

Páčí sa mi to · Odpovedat · Zdieľať · 3 d.

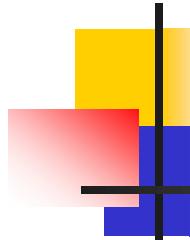


Daniel Peraza
Marko Bašelj there is a JetBrains IDE for iOS development but you still need XCode for compilation

Páčí sa mi to · Odpovedat · Zdieľať · 3 d.



How to use Android studio on low-end machines 4GB 8 GB of RAM



Java vs. Kotlin

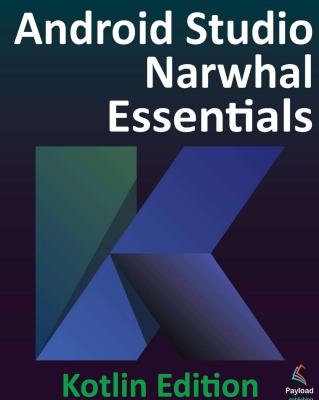


© ScienceSoft USA Corporation

tradičný VMA kurz postavený na Java už štvrtý rok beží v jazyku Kotlin 1.9

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 Googlem oficiálne podporovaným vývojový nástroj pre Android
 - projekt Kotlin má už >13 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 Koala Kotlin Edition

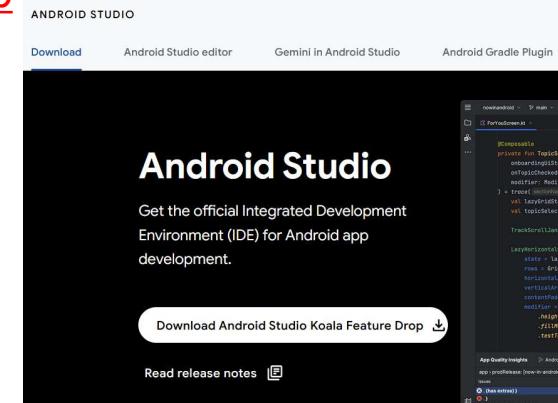
- <https://www.payloadbooks.com/product/android-studio-koala-essentials-kotlin-edition-ebook/>
- <https://www.amazon.com/Android-Studio-Giraffe-Essentials-Developing/dp/1951442776>
- <https://www.ebookfrenzy.com/errata/giraffekotlin.html>
- sources: <https://www.ebookfrenzy.com/retail/giraffekotlin/index.php>

Inštalácia Android Studio:

<https://developer.android.com/studio>

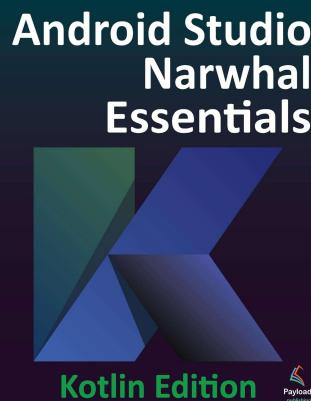
Predmet má cvičenie, ale aj tak:

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



Inštalácia Android Studia:

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



Android Studio Giraffe

Kotlin Edition

- <https://www.payloadbooks.com/product/android-studio-koala-essentials-kotlin-edition-ebook/>
- <https://www.amazon.com/Android-Studio-Giraffe-Essentials-Developing/dp/1951442776>
- <https://www.ebookfrenzy.com/errata/giraffekotlin.html>
- sources: <https://www.ebookfrenzy.com/retail/giraffekotlin/index.php>

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

3. Creating an Example Android App in AS

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

5. Using and Configuring the Android Studio AVD Emulator

6. A tour of the Android Studio User Interface

7. Testing Android Studio App on a Physical Android Device

8. The Basics of the Android Studio Code Editor.

9. An Overview of the Android Architecture

10. The Anatomy of an Android App

11. An Introduction to Kotlin

12. Kotlin Data Types, Variables, and Nullability

13. Kotlin Operators and Expressions

14. Kotlin Control Flow

15. An Overview of Kotlin Functions and Lambdas

16. The Basics of Object Oriented Programming in Kotlin

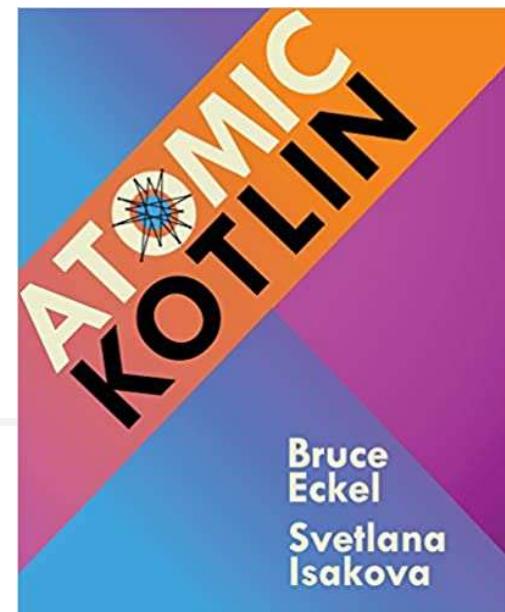
17. An Introduction to Kotlin Inheritance and Subclassing

...

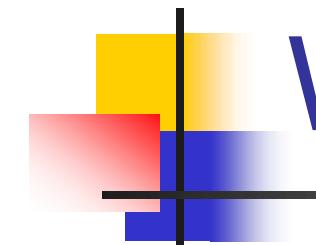
91. An Overview of Gradle in Android Studio



Atomic Kotlin

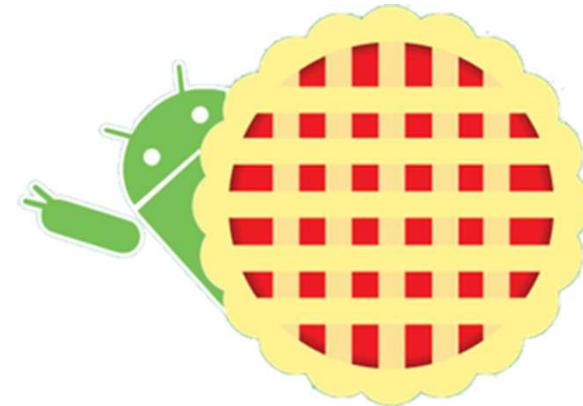


<ul style="list-style-type: none">Section I: Programming Basics<ul style="list-style-type: none">IntroductionWhy Kotlin?Hello, World!var & valData TypesFunctionsif ExpressionsString TemplatesNumber TypesBooleansRepetition with whileLooping & RangesThe in KeywordExpressions & StatementsSummary 1	<p>https://www.amazon.com/Atomic-Kotlin-Bruce-Eckel/dp/0981872557</p> <ul style="list-style-type: none">Section II: Introduction to Objects<ul style="list-style-type: none">Objects EverywhereCreating ClassesPropertiesConstructorsConstraining VisibilityPackagesTestingExceptionsListsVariable Argument ListsSetsMapsProperty AccessorsSummary 2Section III: Usability<ul style="list-style-type: none">Extension FunctionsNamed & Default ArgumentsOverloadingwhen ExpressionsEnumerationsData ClassesDestructuring DeclarationsNullable TypesSafe Calls & the Elvis OperatorNon-Null AssertionsExtensions for Nullable TypesIntroduction to GenericsExtension Propertiesbreak & continueSection IV: Functional Programming<ul style="list-style-type: none">LambdasThe Importance of LambdasOperations on CollectionsMember ReferencesHigher-Order FunctionsManipulating ListsBuilding MapsSequencesLocal FunctionsFolding ListsRecursionSection V: Object-Oriented Programming<ul style="list-style-type: none">InterfacesComplex ConstructorsSecondary ConstructorsInheritanceBase Class InitializationAbstract ClassesUpcastingPolymorphismCompositionInheritance & ExtensionsClass DelegationDowncastingSealed Classes
---	--



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.



Android a Google

2005 [Google](#) acquired Android Inc.
with Rubin, Miner et al.

- 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 [Marshmallow](#), 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 \(API 30\)](#)

API Levels

<https://apilevels.com/>

Version	SDK / API level	Version code	Codename	Cumulative usage ¹	Year ⁴
Android 16	Level 36	BAKLAVA	Baklava ²	0%	2025
Android 15	Level 35	VANILLA_ICE_CREAM	Vanilla Ice Cream ²	10.06%	2024
	▪ <code>targetSdk</code> must be 35+ for new apps and app updates as of August 31, 2025.				
Android 14	Level 34	UPSIDE_DOWN_CAKE	Upside Down Cake ²	43.5%	2023
Android 13	Level 33	TIRAMISU	Tiramisu ²	60.4%	2022
Android 12	Level 32 <small>Android 12L</small>	S_V2	Snow Cone ²	72.6%	
	Level 31 <small>Android 12</small>	S			2021
Android 11	Level 30	R	Red Velvet Cake ²	83.0%	2020
Android 10	Level 29	Q	Quince Tart ²	88.5%	2019
Android 9	Level 28	P	Pie	91.7%	2018
Android 8	Level 27 <small>Android 8.1</small>	O_MR1	Oreo	92.6%	2017
	Level 26 <small>Android 8.0</small>	O		94.8%	
Android 7	Level 25 <small>Android 7.1</small>	N_MR1	Nougat	95.0%	2016
	Level 24 <small>Android 7.0</small>	N		96.2%	
Android 6	Level 23	M	Marshmallow	97.6%	2015
Android 5	Level 22 <small>Android 5.1</small>	LOLLIPOP_MR1	Lollipop	98.0%	
	Level 21 <small>Android 5.0</small>	LOLLIPOP_L		99.8%	
	▪ Jetpack/AndroidX libraries require a <code>minSdk</code> of 21 or higher since April 2024. ▪ Jetpack Compose requires a <code>minSdk</code> of 21 or higher. ▪ Google Play services v23.30.99+ (August 2023) drops support for API levels below 21.				

2. Setting up an Android Studio Development Environment

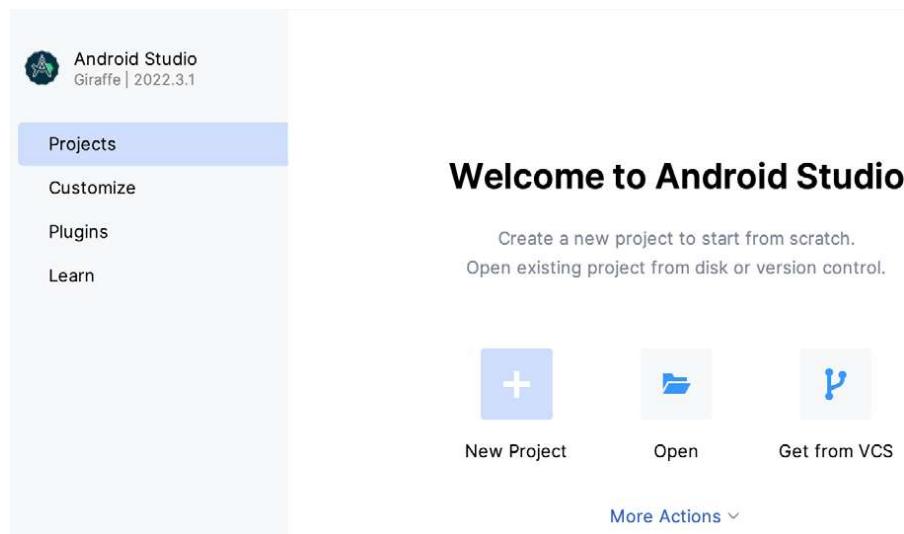


Inštalácia

System requirements

- Windows 8/10/11 64-bit
- macOS 10.14 or later running on Intel or Apple silicon
- Chrome OS device with Intel i5 or higher
- Linux systems with version 2.31 or later of the GNU C Library (glibc)
- **Minimum of 8GB of RAM**
- **Approximately 8GB of available disk space**
- 1280 x 800 minimum screen resolution

<https://developer.android.com/studio/index.html>



2. Setting up an Android Studio Development Environment

Android SDK Packages

Tools/SDK Manager tab SDK Platforms - API 35



Settings

Languages & Frameworks > Android SDK

Manager for the Android SDK and Tools used by the IDE

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

SDK Platforms **SDK Tools** **SDK Update Sites**

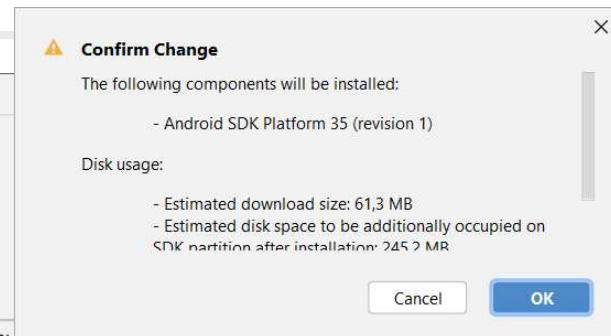
Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, the IDE will automatically check for updates. Check "show package details" to display individual SDK components.

Name	API Level	Rev...	Status
Android API 35			
<input checked="" type="checkbox"/> Android SDK Platform 35	35	1	Not installed
<input type="checkbox"/> Sources for Android 35	35	1	Not installed
<input type="checkbox"/> Google APIs ARM 64 v8a System Image	35	8	Not installed
<input type="checkbox"/> Google APIs Intel x86_64 Atom System Image	35	8	Not installed
<input type="checkbox"/> Google Play ARM 64 v8a System Image	35	8	Not installed
<input type="checkbox"/> Google Play Intel x86_64 Atom System Image	35	8	Not installed
<input type="checkbox"/> Google Play Experimental 16k Page Size ARM 64 v8a System Image	35	3	Not installed
<input type="checkbox"/> Google Play Experimental 16k Page Size Intel x86_64 Atom System Image	35	3	Not installed
<input type="checkbox"/> Pre-Release 16 KB Page Size Google APIs ARM 64 v8a System Image	35	3	Not installed
<input type="checkbox"/> Pre-Release 16 KB Page Size Google APIs Intel x86_64 Atom System Image	35	3	Not installed
Android Vanillacream Preview			
<input type="checkbox"/> Android SDK Platform Vanillacream	Vanillacream	4	Not installed
<input type="checkbox"/> Google APIs ARM 64 v8a System Image	Vanillacream	5	Not installed
<input type="checkbox"/> Google APIs Intel x86_64 Atom System Image	Vanillacream	5	Not installed
<input type="checkbox"/> Google Play ARM 64 v8a System Image	Vanillacream	5	Not installed
<input type="checkbox"/> Google Play Intel x86_64 Atom System Image	Vanillacream	5	Not installed
<input type="checkbox"/> Pre-Release 16 KB Page Size Google APIs ARM 64 v8a System Image	Vanillacream	1	Not installed
<input type="checkbox"/> Pre-Release 16 KB Page Size Google APIs Intel x86_64 Atom System Image	Vanillacream	1	Not installed

Hide Obsolete Packages Show Package Details

?

OK Cancel Apply



2. Setting up an Android Studio Development Environment

Android SDK Packages

Tools/SDK Manager tab SDK Platforms - API 33



Confirm Change

The following components will be installed:

- Google Play Intel x86_64 Atom System Image API 33 (revision 7)
- Sources for Android 33 (revision 1)

Disk usage:

- Estimated download size: 1,4 GB
- Estimated disk space to be additionally occupied on SDK partition after installation: 5,7 GB
- Currently available disk space in SDK root (C:\Users\borovan\AppData\Local\Android\Sdk): 24,0 GB

Name	API Level
Android UpsideDownCakePrivacySandbox Preview	UpsideDownCakePrivacySandbox
Android SDK Platform UpsideDownCakePrivacySandbox	UpsideDownCakePrivacySandbox
Google Play Intel x86_64 Atom System Image	UpsideDownCakePrivacySandbox
Android API 34	
Android TiramisuPrivacySandbox Preview	TiramisuPrivacySandbox
Android SDK Platform TiramisuPrivacySandbox	TiramisuPrivacySandbox
Google Play Intel x86_64 Atom System Image	TiramisuPrivacySandbox
Android 13.0 ("Tiramisu")	
Android SDK Platform 33	33
Sources for Android 33	33
Android TV ARM 64 v8a System Image	33
Android TV Intel x86 Atom System Image	33
Google TV ARM 64 v8a System Image	33
Google TV Intel x86 Atom System Image	33
Google APIs ARM 64 v8a System Image	33
Google APIs Intel x86 Atom_64 System Image	33
Google Play Intel x86_64 Atom System Image	33

Cancel **OK**

Hide Obsolete Packages Show Package Details

SDK Platforms **SDK Tools** **SDK Update Sites**

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.

Settings

Languages & Frameworks > Android SDK

Manager for the Android SDK and Tools used by the IDE

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

Appearance & Behavior
Keymap
Editor
Plugins
Version Control
Build, Execution, Deployment
Languages & Frameworks
C/C++
Schemas and DTDs
Android SDK
GitHub Copilot
Kotlin
Markdown
Template Data Languages
Tools
Advanced Settings
Kotlin Compiler
Experimental

2. Setting up an Android Studio Development Environment

Android SDK Packages

Tools/SDK Manager tab SDK Tools



Settings

Languages & Frameworks > Android SDK

Manager for the Android SDK and Tools used by the IDE

Android SDK Location: C:\Users\borovan\AppData\Local\Android\Sdk [Edit](#) [Optimize disk space](#)

SDK Platforms [SDK Tools](#) [SDK Update Sites](#)

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
<input checked="" type="checkbox"/> Android SDK Build-Tools 35		Update Available: 35.0.0
<input checked="" type="checkbox"/> GPU Debugging tools		Installed
<input type="checkbox"/> NDK (Side by side)		Not Installed
<input checked="" type="checkbox"/> Android SDK Command-line Tools (latest)		Update Available: 16.0
<input type="checkbox"/> CMake		Not Installed
<input type="checkbox"/> Android Auto API Simulators	1	Not installed
<input type="checkbox"/> Android Auto Desktop Head Unit Emulator	2.0	Not installed
<input checked="" type="checkbox"/> Android Emulator	32.1.15	Update Available: 35.1.21
<input type="checkbox"/> Android Emulator hypervisor driver (installer)	2.2.0	Not installed
<input checked="" type="checkbox"/> Android SDK Platform-Tools	35.0.2	Installed
<input checked="" type="checkbox"/> Android SDK Tools	26.1.1	Installed
<input checked="" type="checkbox"/> Android Support Repository	47.0.0	Installed
<input checked="" type="checkbox"/> 1.0.2	1	Installed
<input type="checkbox"/> Google Play APK Expansion library	1	Not installed
<input checked="" type="checkbox"/> Google Play Instant Development SDK	1.9.0	Installed
<input type="checkbox"/> Google Play Licensing Library	1	Not installed
<input checked="" type="checkbox"/> Google Play services	49	Installed
<input checked="" type="checkbox"/> Google Repository	58	Installed
<input checked="" type="checkbox"/> Google USB Driver	13	Installed

Hide Obsolete Packages Show Package Details

?

OK Cancel Apply

4. Creating an Android Virtual Device (AVD) in Android Studio

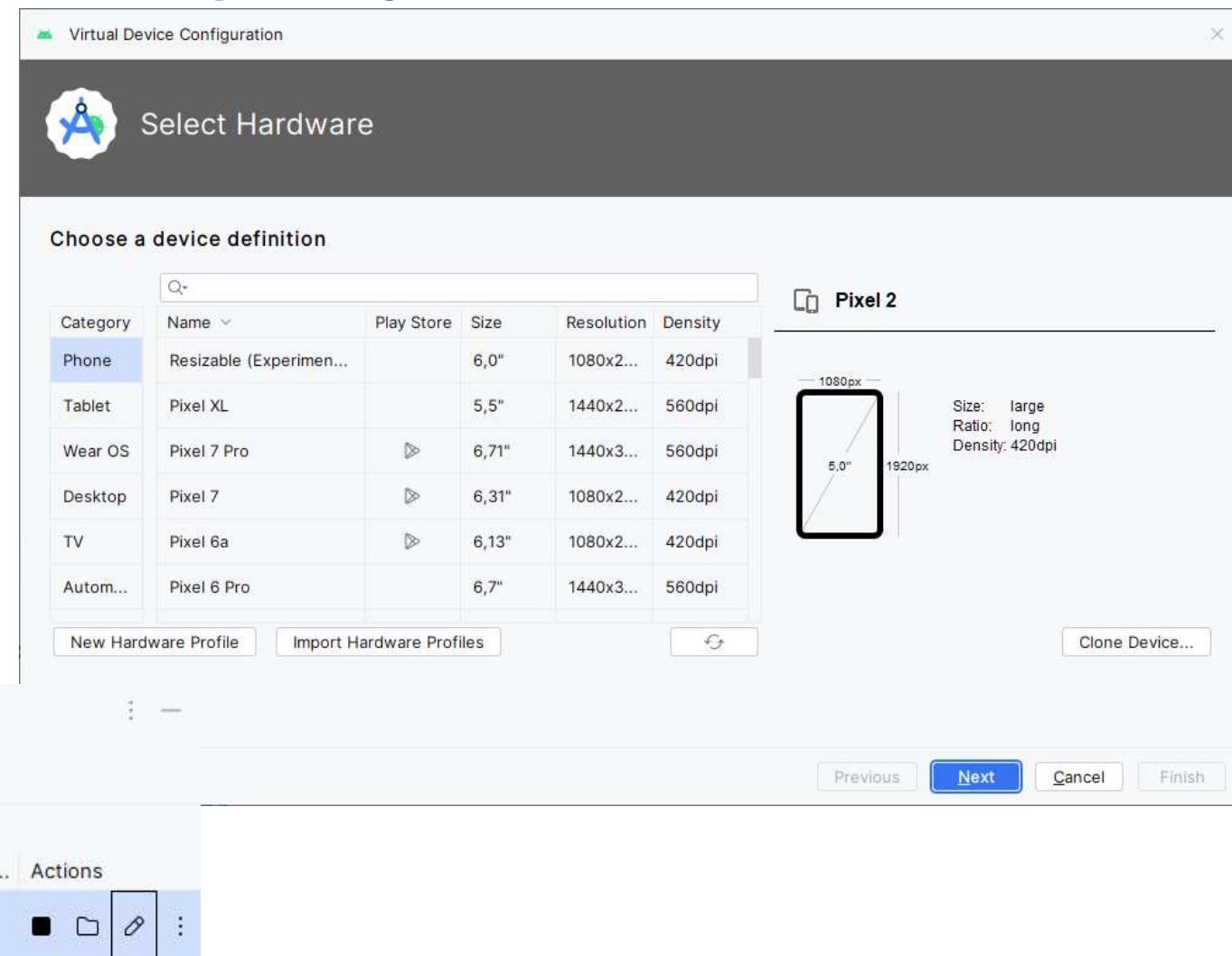
Android Virtual Device

Tools/AVD manager

Nakonfigurujte si AVD zodpovedajúci vášmu zariadeniu

alebo si vyberte zo
zoznamu
predvolených,

- Create Device
- modifikujte
nastavenia
podľa vášho
zariadenia

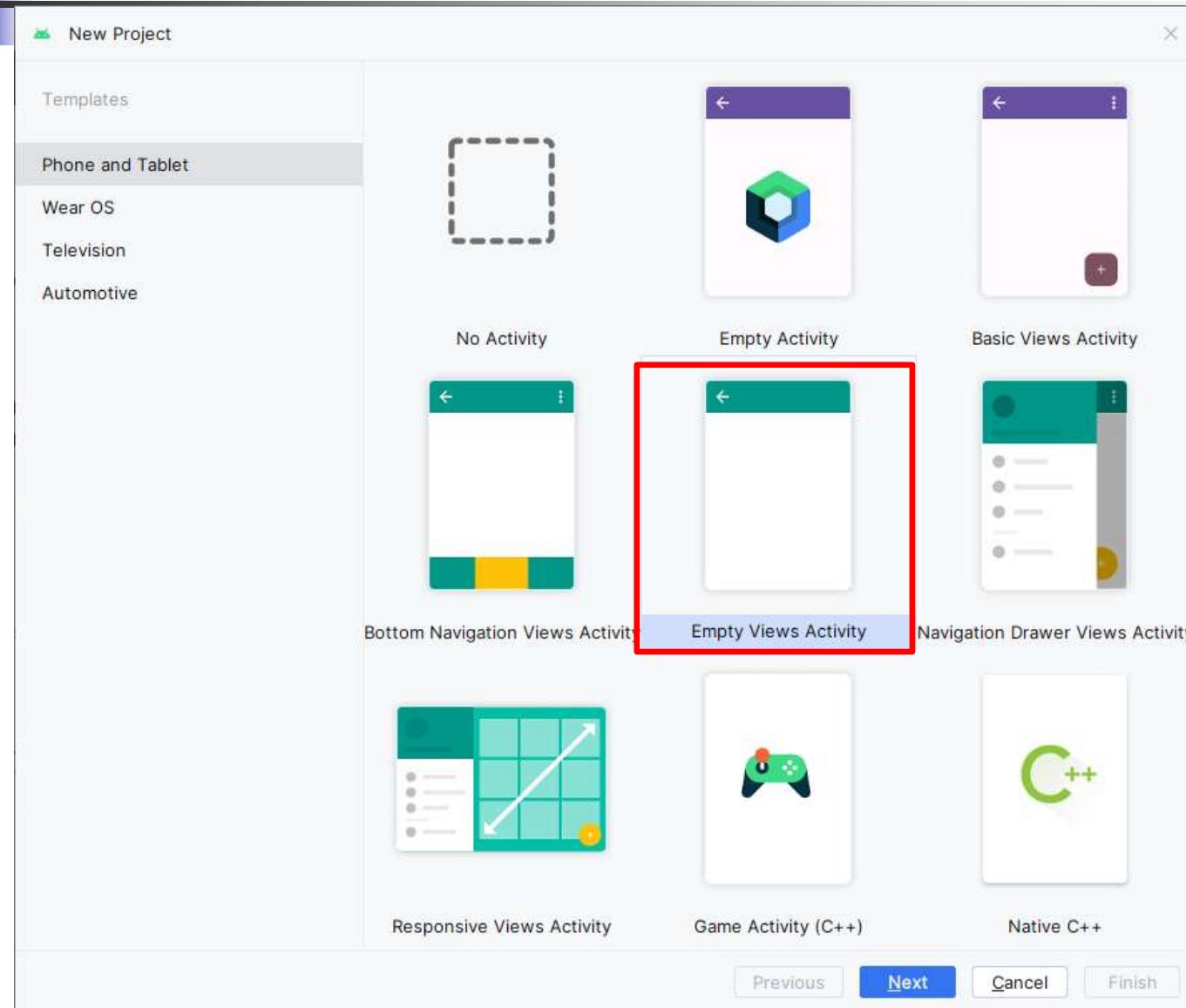


Chapter 3

3. Creating an Example Android App in Android Studio

Nový projekt

(File/New/New Android Project)



Nový projekt

(File/New/New Android Project)

New Project

Empty Views Activity
Creates a new empty activity

Name: EmptyApplication2024

Package name: com.example.emptyapplication2024

Save location: D:\borovan\workspace_ArAndroidStudio\EmptyApplication2024

Language: Kotlin

Minimum SDK: API 23 ("Marshmallow"; Android 6.0)

*Your app will run on approximately **98,8%** of devices.*
[Help me choose](#)

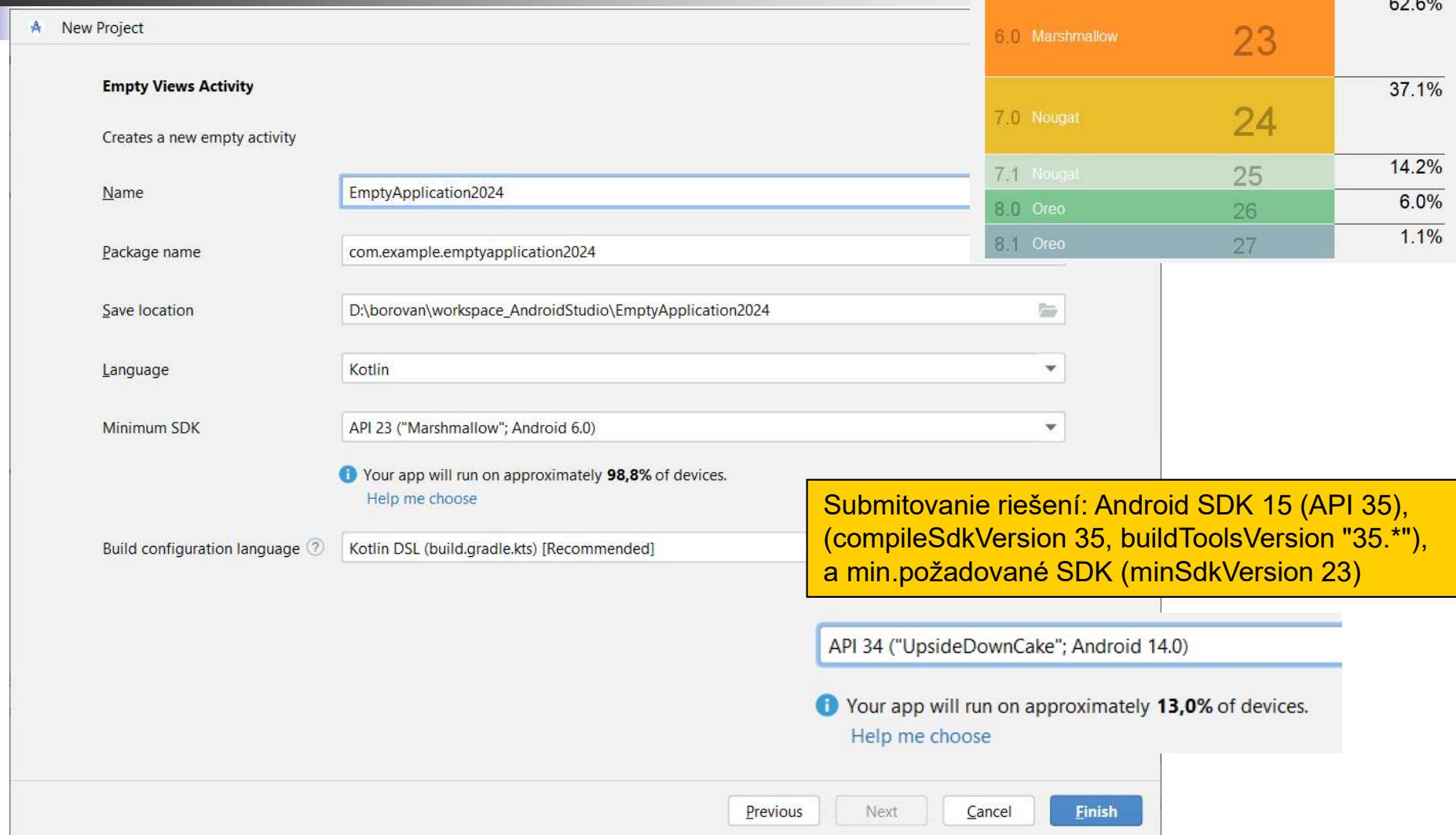
Build configuration language: [Kotlin DSL \(build.gradle.kts\) \[Recommended\]](#)

**Submitovanie riešení: Android SDK 15 (API 35),
(compileSdkVersion 35, buildToolsVersion "35.*"),
a min.požadované SDK (minSdkVersion 23)**

API 34 ("UpsideDownCake"; Android 14.0)

*Your app will run on approximately **13,0%** of devices.*
[Help me choose](#)

Previous Next Cancel Finish



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%

Nový projekt

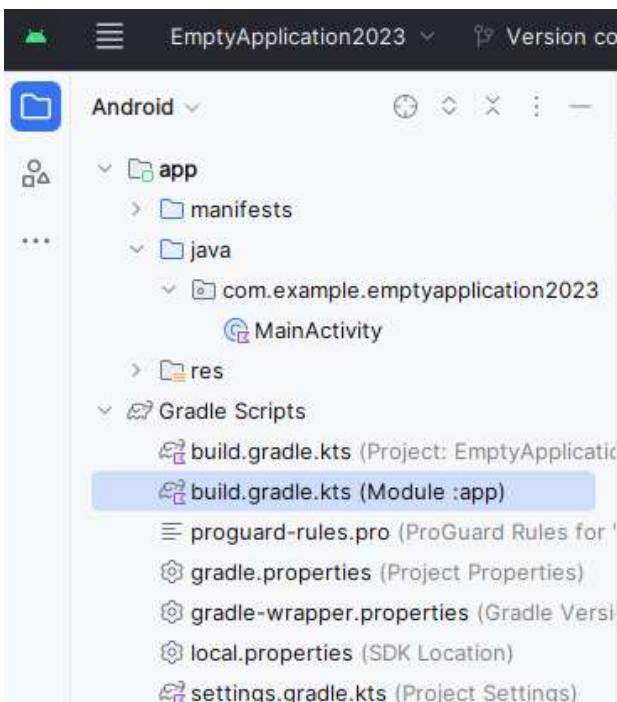
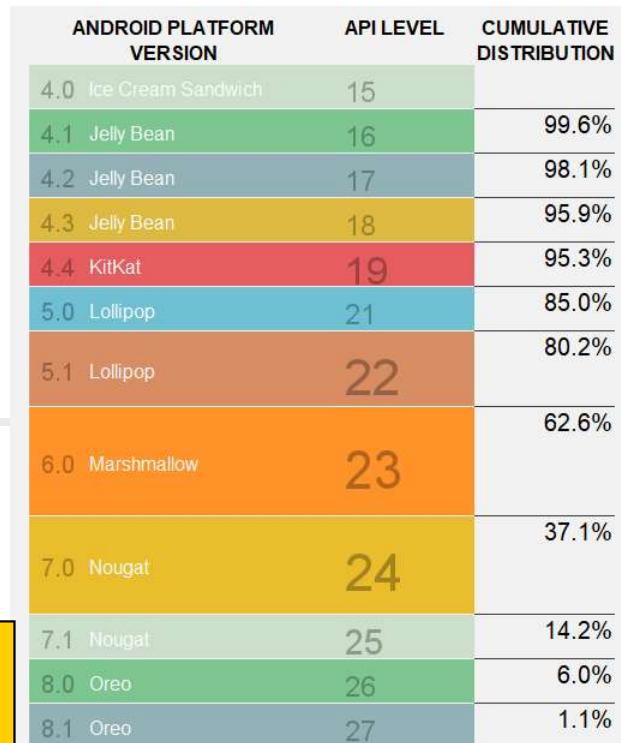
(File/New/New Android Project)

```
plugins {  
    id("com.android.application")  
    id("org.jetbrains.kotlin.android")  
}
```

Submitovanie riešení: Android SDK 15 (API 35),
(compileSdkVersion 35, buildToolsVersion "35.*"),
a min.požadované SDK (minSdkVersion 23)

```
android {  
    namespace = "com.example.emptyapplication2024"  
    compileSdk = 35  
  
    defaultConfig {  
        applicationId = "com.example.emptyapplication2024"  
        minSdk = 23  
        targetSdk = 35  
        versionCode = 1  
        versionName = "1.0"  
    }  
}
```

ects be like...



Nový projekt (Empty views activity)



The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The left sidebar shows the project structure for "EmptyApplication2023". It includes the **app** module with **manifests**, **java** (containing `MainActivity.kt`), and **res** (containing `layout/activity_main.xml`, `drawable`, `values/colors.xml`, `values/strings.xml`, and `themes`). The `activity_main.xml` file is currently selected.
- Layout Editor:** The main area displays the XML layout file `activity_main.xml`. It contains a single `TextView` with the text "Hello World!". The `TextView` is constrained to the center of the screen using a `ConstraintLayout`.
- Toolbar:** The top toolbar shows the device as "Pixel 6 API 33", the orientation as "Portrait", and other standard icons.
- Right Panel:** The right panel contains the **Attributes** panel for the selected `Ab TextView` component. It shows the following settings:
 - Declared Attributes:** `id` (set to <unnamed>).
 - Layout:** `Constraint Widget` section shows the constraint graph with two horizontal constraints and one vertical constraint.
 - Constraints (4):** `layout_width` is set to `wrap_content`, `layout_height` is set to `wrap_content`, and `visibility` is set to `visible`.
 - Transforms:** `Rotation` is set to 0 for all axes (x, y, z). `rotation`, `rotationX`, `rotationY`, and `scaleX` are also set to 0.
- Bottom Status Bar:** The status bar at the bottom shows the project name "EmptyApplication2023", the file path "app/src/main/res/layout/activity_main.xml", and system information like "6.7 LF", "UTF-8", and "4 spaces".

Pýtajte sa kým nedostanete



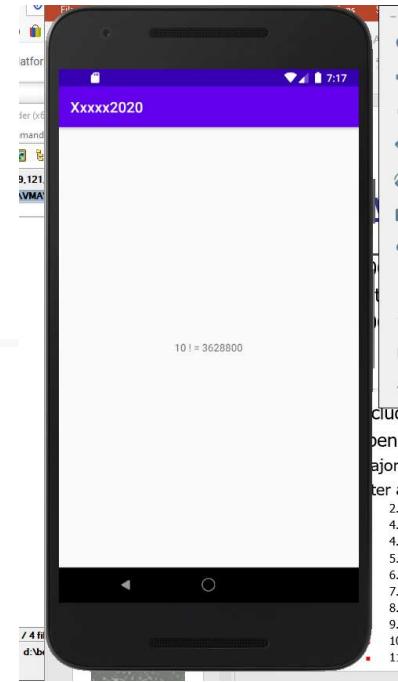
The screenshot displays the Android Studio interface for a project named "EmptyApplication2023".

- Project Structure:** Shows the project tree with modules like app, java, res, and layout.
- Layout Editor:** The main window shows the XML layout file "activity_main.xml" containing a single "Hello World" TextView. The layout uses a ConstraintLayout.
- Code Editor:** The "MainActivity.kt" file is open, showing the generated Java code for the activity.
- Logcat Panel:** Displays logs from the emulator, indicating the app was launched on "Pixel 6 API 33" and the intent was successfully sent to the MainActivity.
- Toolbar:** Standard Android Studio icons for file operations, navigation, and search.
- Bottom Status Bar:** Shows the current file path ("EmptyApplication2023 > app > src > main > res > layout > activity_main.xml") and system status (8:1 LF, UTF-8, 4 spaces).

Ako si skúšať Kotlin v AS

(kým sa nedozvieme viac)

```
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 != ${fact(i)}")  
            // vypisuje do konzoly Logcat, použite filter s "TAG"  
            val tv = findViewById<TextView>(R.id.tv)  
            tv.text = "$i != ${fact(i)}"  
            // vypise do View komponentu, ktorý je v Aktivite  
            Toast.makeText(this, "$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)  
}
```

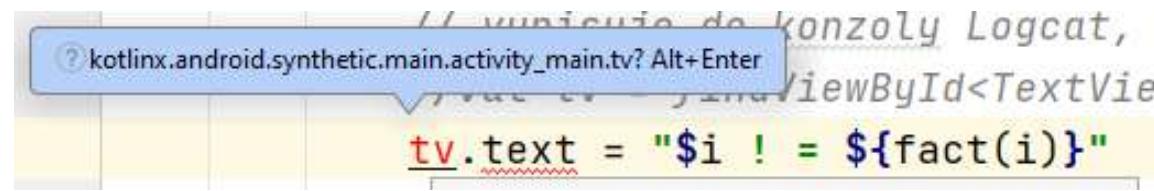


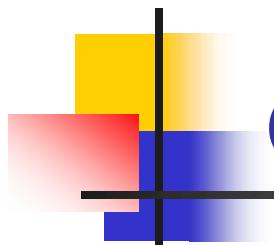
Integrovanie Android Extensions

```
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 ! = ${fact(i)}"
    // vypise do View komponentu, ktory je v Aktivite
    Toast.makeText(this, "$i ! = ${fact(i)}",
        Toast.LENGTH_SHORT).show()
    // Toast alias Notifier (MITI)
}
```





Break point

(štruktúrou projektu pokračujeme na budúce)

- Switch to kotlin intro