

aplikacje mobilne
Android Studio

problemy przy instalacji Android Studio

Android SDK missing



I had to manually create

```
C:\Users\[USERNAME]\AppData\Local\Android\Sdk
```

0

and then I downloaded the tools from **Command Line Tools Only** section on:

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



then I extracted the files to:

```
C:\Users\[USERNAME]\AppData\Local\Android\Sdk
```

Hope this helps someone.

Command line tools only

If you do not need Android Studio, you can download the basic Android command line tools below. You can use the included [sdkmanager](#) to download other SDK packages.

These tools are included in Android Studio.

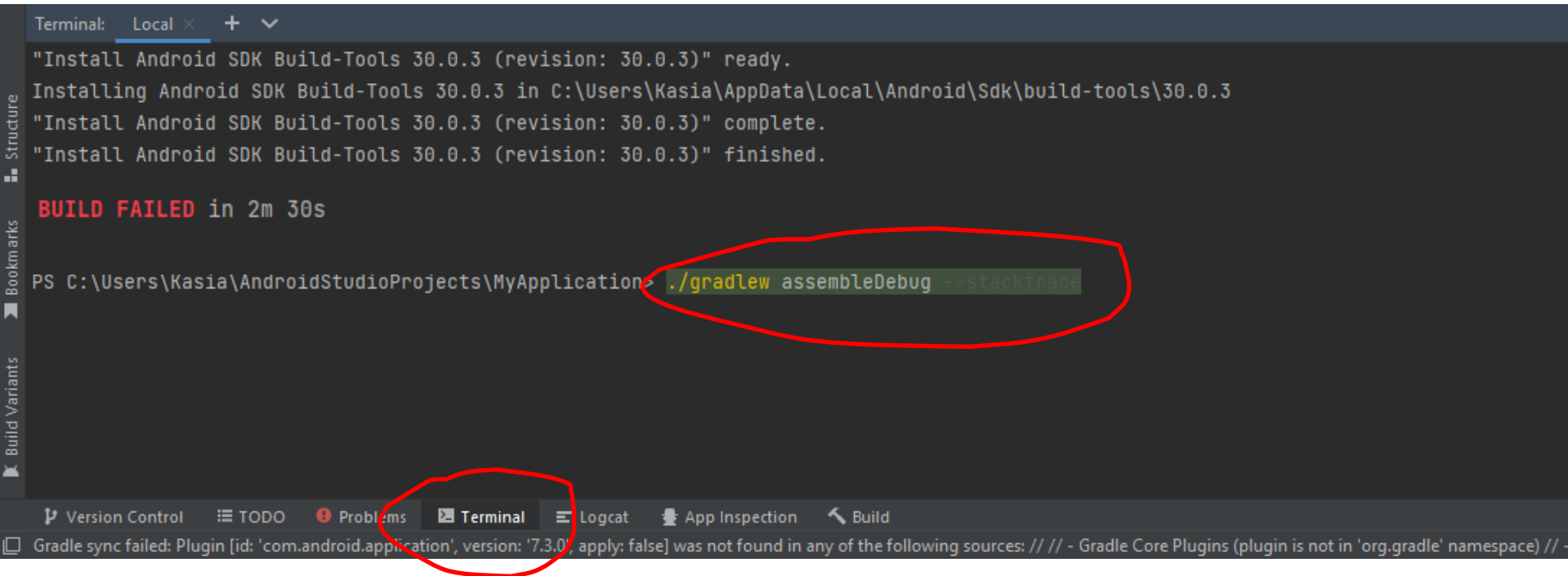
Platform	SDK tools package	Size	SHA-256 checksum
Windows	commandlinetools-win-8512546_latest.zip	108 MB	9cea28ab5d8c2f4b6077911f1475c4f532f42702c42f53ee1e331d9e36ee572
Mac	commandlinetools-mac-8512546_latest.zip	108 MB	f8101079e8907edc8389eb2560a62e9c3c87f2d1ae4a3d517f80234ff3f11
Linux	commandlinetools-linux-8512546_latest.zip	108 MB	2ccbda4302db862a28ads25aa7425d599dor9462046003c1714b059b65c4797068

problemy przy instalacji Android Studio

Gradle Core Plugins (plugin is not in 'org.gradle' namespace)

uruchomić polecenie w terminalu:

```
./gradlew assembleDebug --stacktrace
```



The screenshot shows the terminal window of an IDE. The terminal output includes the following text:

```
Terminal: Local x + v  
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" ready.  
Installing Android SDK Build-Tools 30.0.3 in C:\Users\Kasia\AppData\Local\Android\Sdk\build-tools\30.0.3  
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" complete.  
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" finished.  
  
BUILD FAILED in 2m 30s  
  
PS C:\Users\Kasia\AndroidStudioProjects\MyApplication> ./gradlew assembleDebug --stacktrace
```

The command `./gradlew assembleDebug --stacktrace` is highlighted with a red oval. Below the terminal window, the IDE's status bar shows a red error icon and the text: "Gradle sync failed: Plugin [id: 'com.android.application', version: '7.3.0', apply: false] was not found in any of the following sources: // - Gradle Core Plugins (plugin is not in 'org.gradle' namespace) // -". The "Terminal" tab in the status bar is also highlighted with a red oval.

problemy przy instalacji Android Studio



```
Terminal: Local x + v
at com.android.build.gradle.internal.SdkHandler.initTarget(SdkHandler.java:131)
... 167 more
* Get more help at https://help.gradle.org
BUILD FAILED in 1m 45s
```

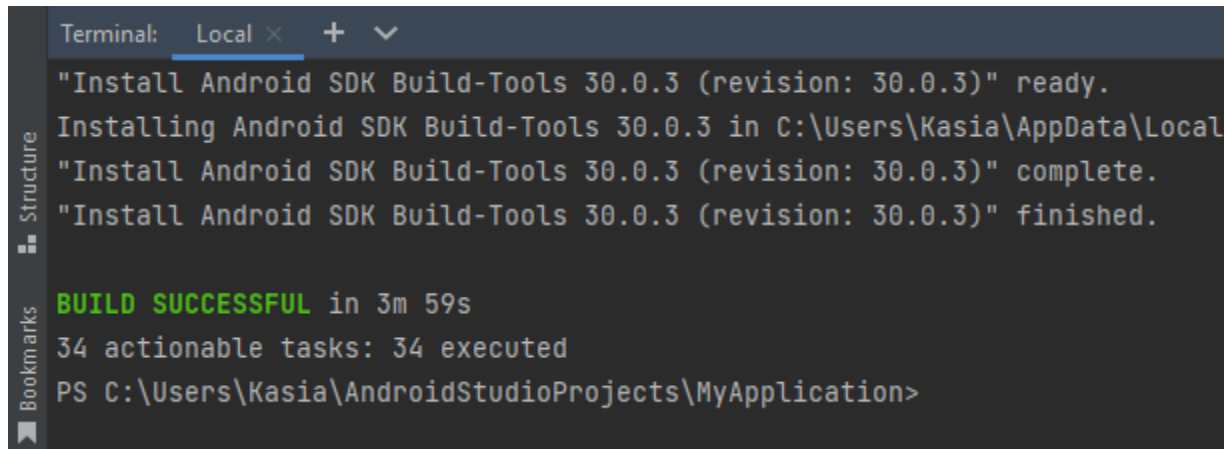
[Error:Unknown host 'dl.google.com' You may need to adjust the proxy settings in Gradle](#)

I was also having the same problem. I tried the following and it's working for me:

Windows

Go to : *File > Settings > Appearance & Behavior > System Settings > HTTP Proxy [Under IDE Settings]*
Enable following option Auto-detect proxy settings

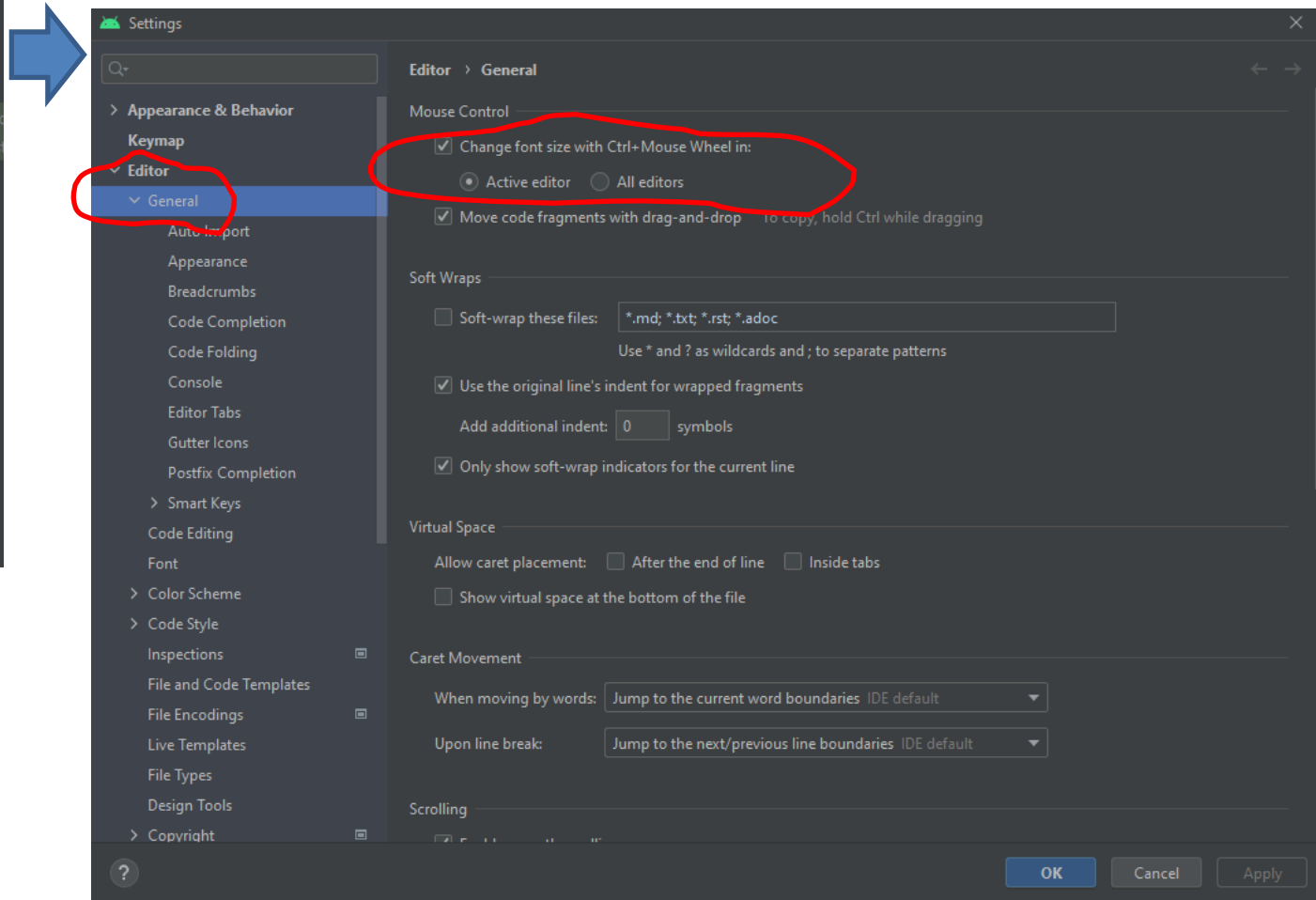
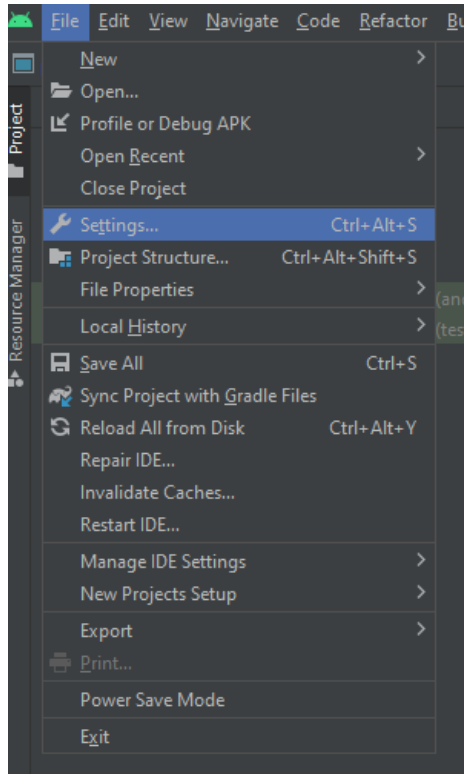
problemy przy instalacji Android Studio



```
Terminal: Local x + v
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" ready.
Installing Android SDK Build-Tools 30.0.3 in C:\Users\Kasia\AppData\Local
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" complete.
"Install Android SDK Build-Tools 30.0.3 (revision: 30.0.3)" finished.

BUILD SUCCESSFUL in 3m 59s
34 actionable tasks: 34 executed
PS C:\Users\Kasia\AndroidStudioProjects\MyApplication>
```

Android Studio zmiana wielkości czcionek CTRL Mouse Wheel

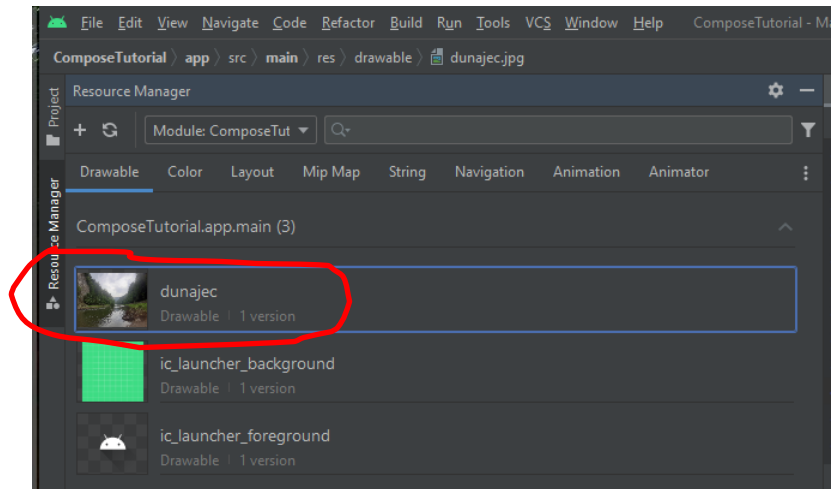
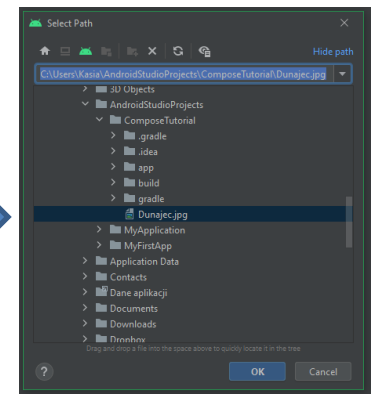
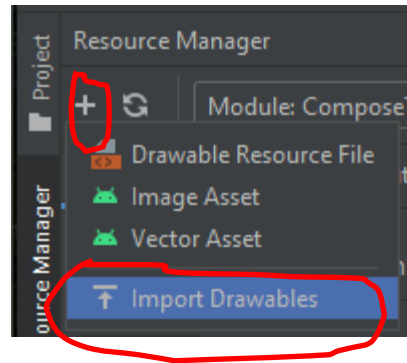
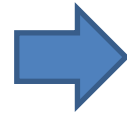
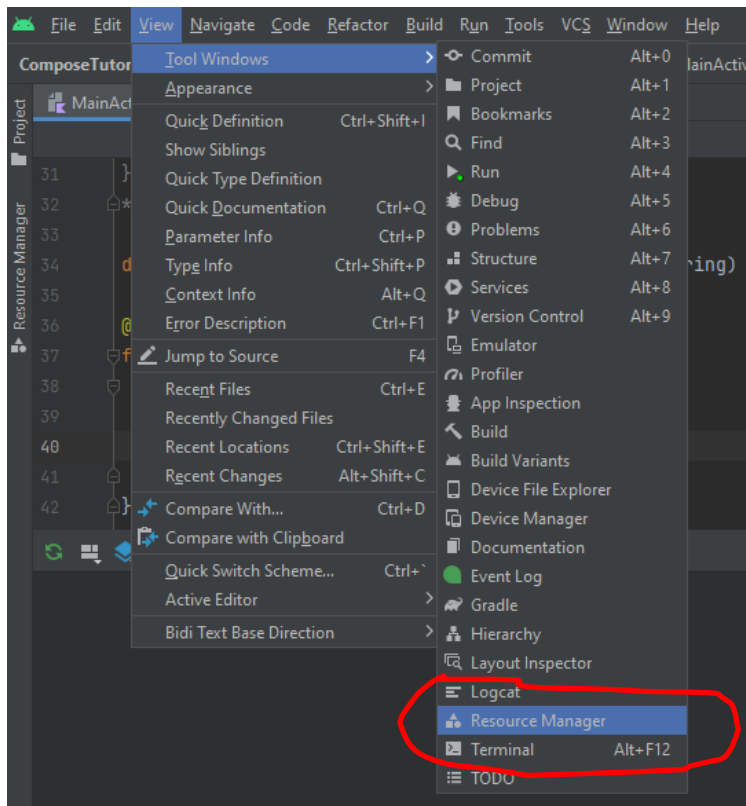


SDK Manager emulator akcelerator

The screenshot shows the Android Studio IDE with the 'Settings' dialog open to the 'Android SDK' section. The 'Tools' tab is active, displaying a list of available SDK developer tools. The 'Intel x86 Emulator Accelerator (HAXM installer)' is highlighted, indicating it is not installed. Two blue arrows point to the 'Tools' tab and the HAXM entry. A red circle highlights the 'Tools' icon in the IDE's top toolbar.

Name	Version	Status
<input checked="" type="checkbox"/> Android SDK Build-Tools 33		Installed
<input type="checkbox"/> NDK (Side by side)		Not Installed
<input type="checkbox"/> Android SDK Command-line Tools (latest)		Not Installed
<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	1.1	Not installed
<input type="checkbox"/> Android Emulator	31.3.10	Update Available: 31.3.11
<input type="checkbox"/> Android Emulator Hypervisor Driver for AMD Processors (installer)	1.8.0	Not installed
<input checked="" type="checkbox"/> Android SDK Platform-Tools	33.0.3	Installed
<input type="checkbox"/> Google Play APK Expansion library	1	Not installed
<input type="checkbox"/> Google Play Instant Development SDK	1.9.0	Not installed
<input type="checkbox"/> Google Play Licensing Library	1	Not installed
<input type="checkbox"/> Google Play services	49	Not installed
<input type="checkbox"/> Google USB Driver	13	Not installed
<input type="checkbox"/> Google Web Driver	2	Not installed
<input checked="" type="checkbox"/> Intel x86 Emulator Accelerator (HAXM installer)	7.6.5	Not installed
<input type="checkbox"/> Layout Inspector image server for API 29-30	6	Not installed
<input type="checkbox"/> Layout Inspector image server for API 31 and T	1	Not installed
<input type="checkbox"/> Layout Inspector image server for API S	3	Not installed

Resource Manager



<https://developer.android.com/studio/write/resource-manager#import>

Podstawy aplikacji

- Aplikacje na Androida można pisać w językach Kotlin, Java i C++.
- Android SDK kompiluje kod aplikacji wraz z wszelkimi plikami danych i zasobów do pliku APK lub pakietu Android App Bundle.
- plik APK (Android Package Kit, archiwum z sufiksem .apk), jest używany do dystrybucji oraz instalacji przez urządzenia z systemem Android.
- Android App Bundle (archiwum z sufiksem .aab) jest formatem programów wymaganych od programistów przez sklep Google Play. Na jego podstawie serwery Google generują finalny APK oraz podpisują go kluczem Google lub kluczem używanym w imieniu wydawcy.
- Każda aplikacja na Androida jest uruchamiana we własnej bezpiecznej piaskownicy (system przypisuje każdej aplikacji unikalny identyfikator Linux user ID, domyślnie każda aplikacja działa w oddzielnym procesie, każdy proces ma własną maszynę wirtualną (VM))
- System Android realizuje zasadę najmniejszych uprawnień. Oznacza to, że domyślnie każda aplikacja ma dostęp tylko do tych komponentów, których potrzebuje do swojej pracy i żadnych innych.

<https://developer.android.com/guide/components/fundamentals>

<https://pl.wikipedia.org/wiki/APK>

Składniki aplikacji

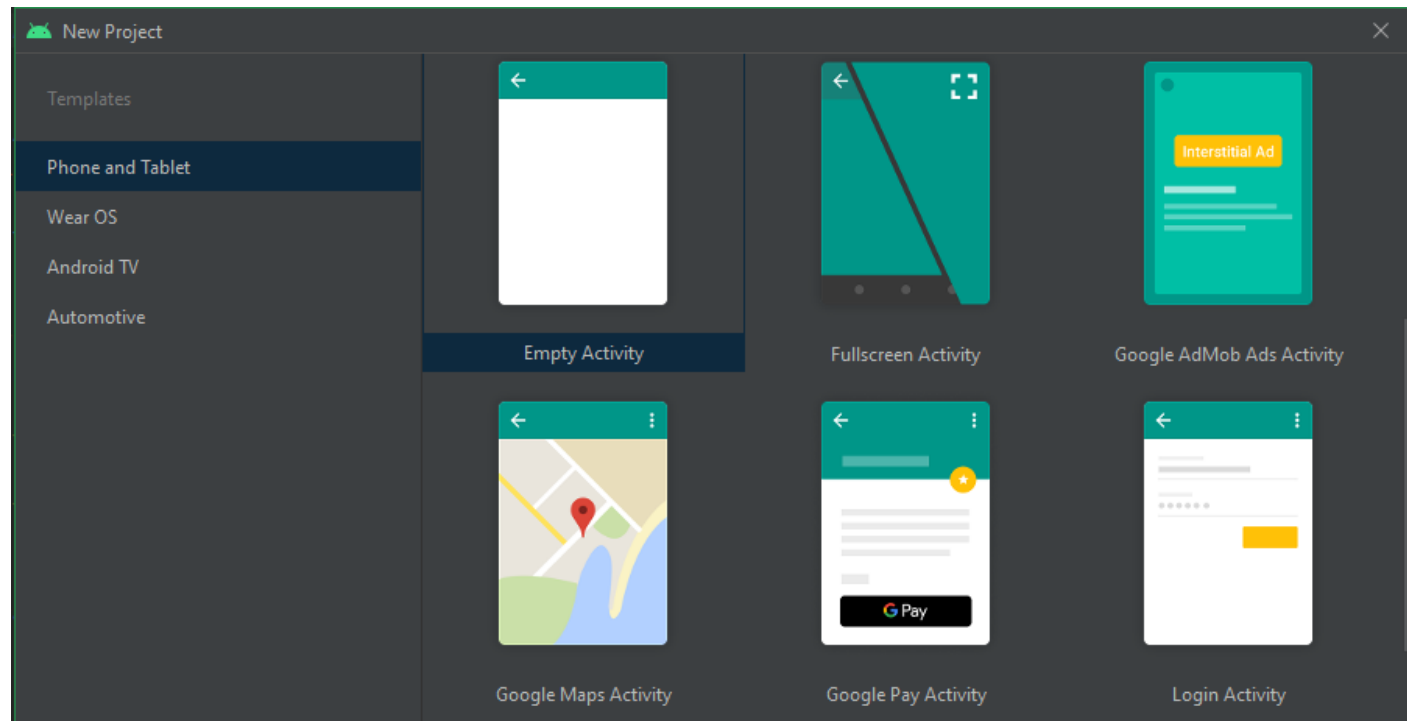
Składniki aplikacji to podstawowe elementy składowe aplikacji.

Istnieją cztery różne typy składników aplikacji:

- Activities
- Services
- Broadcast receivers
- Content providers

Activities (widoki)

- Reprezentuje pojedynczy ekran z interfejsem użytkownika. Odpowiada za interakcję z użytkownikiem.
- Na przykład aplikacja poczty e-mail może mieć Activity, które pokazuje listę nowych wiadomości e-mail, inne Activity związane z tworzeniem wiadomości e-mail oraz inne Activity związane z czytaniem wiadomości e-mail.



Services (usługi)

- Jest to komponent, który działa w tle w celu wykonywania długotrwałych operacji lub wykonywania pracy dla zdalnych procesów.
- Usługa nie zapewnia interfejsu użytkownika. Na przykład może odtwarzać muzykę w tle, gdy użytkownik używa innej aplikacji, lub może pobierać dane przez sieć bez blokowania interakcji użytkownika z aplikacją.
- Istnieją dwa rodzaje usług, które informują system, jak zarządzać aplikacją:
 - ✓ started services (usługi uruchomione)
 - ✓ bound services (usługi powiązane)
- started services informują system, aby działały, dopóki ich praca nie zostanie zakończona. Może to być synchronizacja niektórych danych w tle lub odtwarzanie muzyki nawet po opuszczeniu aplikacji przez użytkownika
- bound services działają, ponieważ jakaś inna aplikacja (lub system) powiedziała, że chce z niej skorzystać. Jest to w zasadzie usługa udostępniająca interfejs API dla innego procesu. W ten sposób system wie, że istnieje zależność między tymi procesami, więc jeśli proces A jest powiązany z usługą w procesie B, wie, że musi utrzymać proces B (i jego usługę) w działaniu dla A.

<https://developer.android.com/guide/components/services>

Broadcast receivers

broadcast receiver (odbiornik powiadomień) to składnik, który umożliwia systemowi dostarczanie powiadomień do aplikacji, dzięki czemu aplikacja może odpowiadać na ogólnosystemowe zdarzenia. System może dostarczać powiadomienia nawet do aplikacji, które nie są uruchomione. Wiele powiadomień pochodzi z systemu — na przykład powiadomienie informujące o wyłączeniu ekranu, niskim poziomie naładowania baterii lub zrobieniu zdjęcia. Choć *broadcast receivers* nie wyświetlają interfejsu użytkownika, mogą utworzyć powiadomienie na pasku stanu, aby informować użytkownika o wystąpieniu zdarzenia.

<https://developer.android.com/guide/components/fundamentals>

Content providers

Content providers (dostawca zawartości) zarządza udostępnionym zestawem danych aplikacji, które można przechowywać w systemie plików, w bazie danych SQLite, w sieci Web lub w dowolnej innej trwałej lokalizacji, do której aplikacja ma dostęp. Za pośrednictwem dostawcy treści inne aplikacje mogą wysyłać zapytania do danych lub modyfikować je, jeśli pozwala na to dostawca treści. Na przykład system Android zapewnia dostawcę treści, który zarządza informacjami kontaktowymi użytkownika.

<https://developer.android.com/guide/topics/providers/content-providers>

Intent

Trzy z czterech komponentów — activities, services, and broadcast receivers— są aktywowane przez asynchroniczną wiadomość zwaną *intent*. *Intents* wiążą ze sobą poszczególne składniki w czasie działania aplikacji. Można myśleć o nich jako o komunikatorach, które żądają akcji od innych składników, niezależnie od tego, czy składnik należy do danej aplikacji, czy do innej.

Intent jest tworzony za pomocą obiektu Intent, który definiuje komunikat, w celu aktywowania określonego komponentu lub określonego typu komponentu.

Intent może na przykład zawierać prośbę o pokazanie obrazu lub otwarcie strony internetowej.

Inny *intent* może umożliwić użytkownikowi wybranie kontaktu osobistego i zwrócenie go do naszej aplikacji. *Intent* zwrotu zawiera identyfikator URI wskazujący na wybrany kontakt.

<https://developer.android.com/guide/components/intents-filters>

plik AndroidManifest.xml

Zanim system Android będzie mógł uruchomić komponent aplikacji, system musi wiedzieć, że komponent istnieje, odczytując plik manifestu aplikacji, AndroidManifest.xml. Aplikacja musi zadeklarować w nim wszystkie używane przez nią komponenty. Plik manifestu musi znajdować się w katalogu głównym projektu aplikacji.

Manifest oprócz deklarowania komponentów aplikacji wykonuje szereg innych czynności, takich jak:

- Identyfikuje wszelkie uprawnienia użytkownika wymagane przez aplikację, takie jak dostęp do Internetu lub dostęp do odczytu kontaktów użytkownika.
- Deklaruje minimalny poziom interfejsu API wymagany przez aplikację, na podstawie używanych przez aplikację interfejsów API.
- Deklaruje funkcje sprzętu i oprogramowania używane lub wymagane przez aplikację, takie jak aparat, usługi Bluetooth lub ekran wielodotkowy.
- Deklaruje biblioteki API, z którymi aplikacja musi być połączona (inne niż interfejsy API platformy Android), takie jak biblioteka Google Maps.

<https://developer.android.com/guide/topics/manifest/manifest-intro>

App resources (zasoby aplikacji)

Aplikacja na Androida składa się nie tylko z kodu — wymaga zasobów odrębnych od kodu źródłowego, takich jak obrazy, pliki audio i wszystko, co dotyczy wizualnej prezentacji aplikacji. Za pomocą plików XML można zdefiniować animacje, menu, style, kolory i układ interfejsów użytkownika.

Korzystanie z zasobów aplikacji ułatwia aktualizowanie różnych cech aplikacji bez modyfikowania kodu. Udostępnianie zestawów alternatywnych zasobów umożliwia optymalizację aplikacji pod kątem różnych konfiguracji urządzeń, takich jak różne języki i rozmiary ekranu.

Dla każdego zasobu uwzględnionego w projekcie SDK definiuje unikalny identyfikator, którego można użyć do odwoływania się do zasobu z kodu aplikacji lub z innych zasobów zdefiniowanych w języku XML.

Na przykład, jeśli aplikacja zawiera plik obrazu o nazwie obrazek.png (zapisany w katalogu res/drawable/), SDK generuje identyfikator zasobu o nazwie R.drawable.obrazek. Ten identyfikator jest mapowany na liczbę całkowitą specyficzną dla aplikacji, której można użyć do odwoływania się do obrazu i wstawienia go do interfejsu użytkownika.

<https://developer.android.com/guide/topics/resources/providing-resources>

developer.android.com

developers 

Platform

Android Studio

Google Play

Jetpack

Kotlin

Docs

Games

 Search

Join a Compose Camp



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

Android tutorial



LEARN ANDROID

application development



Android Video Tutorials

Android Basics

Android - Home

Android - Overview



LEARN ANDROID

simply easy learning

Android Tutorial

📄 PDF Version

🔗 Resources

🔍 Job Search

💬 Discussion

Android is an open source and Linux-based operating system for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies. This tutorial will teach you basic Android programming and will also take you through some advance concepts related to Android application development.

<https://www.tutorialspoint.com/android/index.htm>

Android tutorial

[Courses](#) ▾ [Tutorials](#) ▾ [Jobs](#) ▾ [Practice](#) ▾ [Contests](#)



[Data Structures](#) [Algorithms](#) [Interview Preparation](#) [Topic-wise Practice](#) [C++](#) [Java](#) [Python](#) [Competitive Programming](#) [Machine Learning](#) [JavaScript](#) [SDE Sheet](#) [Puzzles](#) [GFG School](#) [Projects](#)

Expected Number of Trials until Success

Linearity of Expectation

Expectation or expected value of an array

Random Walk (Implementation in Python)

Page Rank Algorithm and Implementation

Hyperlink Induced Topic Search (HITS) Algorithm using Networxx Module | Python

Introduction to Pafy Module in Python

YouTube Media/Audio Download using Python – pafy

Python | Download YouTube videos using youtube_dl module

Pytube | Python library to download youtube videos

Create GUI for Downloading Youtube Video using Python

Downloading files from web using Python

Implementing Web Scraping in Python with BeautifulSoup

Android Studio Tutorial

Difficulty Level : Easy • Last Updated : 13 May, 2021

[Read](#)

[Discuss](#)



It is stated that “If you give me six hours to chop down a tree then I will spend the first four hours in sharpening the axe”. So in the Android Development World if we consider Android Development as the tree then Android Studio should be the axe. Yes, if you are starting Android Development then you must be familiar with Android Studio i.e. the tool that is used for developing Android applications. Android Studio is the official IDE (Integrated Development Environment) for Android app development, and it is based on JetBrains’ IntelliJ IDEA software.



<https://www.geeksforgeeks.org/android-studio-tutorial/>

Android tutorial



How To Add Image To Drawable Folder In Android Studio

How To Change Icon Of Your Android App In Android Studio

How To Add Audio To App In Android Studio

Different Application Launcher Icon Sizes In Android Studio

How to Create Basic Activity In Android Studio

How To Implement Abstract Method In Android Studio

How To Change Package Name In Android Studio

How To Generate Signed Apk In Android Studio For Publishing & Updating App

How To Publish Android App On PlayStore

How To Create Anim Folder & Animation File In Android Studio

Android Studio Tutorials For Beginner – Step By Step:

Below are the tutorials links on Android Studio:

- [Start New Project](#) – Learn how to start or [create a new project](#) in Android Studio
- [Open Project](#) – Learn how to open projects and recent project
- [Reopen, Close & Save Project](#) – Learn more about how to open saved projects and close current project in Android Studio.
- [Create New Activity](#) – Learn to create New Activity in Android Studio further create XML file for designing UI and java file coding.
- [Create New Java Class](#) – Learn to create New Java Class in Android Studio.
- [Create Virtual Device](#) – Learn more to create a new AVD (virtual device) in Emulator.
- [Run App In AVD](#) – Learn how to run and test Android App in AVD of Emulator.
- [Run/Test App in Real Device](#) – Click to know more about how to run app in real device.
- [Create Drawable Resource XML File](#) – Learn how to create drawable resource XML file in Android Studio.
- [Add/Create Landscape Layout](#) – Learn to design app in Landscape orientation in Android Studio.
- [Create Local HTML File](#) – Learn how to create local [HTML](#) file in Android Studio.
- [Create Raw Folder](#) – Click & get more about how to create Raw Folder in Android Studio.
- [Add/Create Assets Folder](#) – Learn more to create Assets folder in Android Studio.
- [Install Genymotion Emulator](#) – Learn the steps to install Genymotion Emulator in Android Studio.
- [Import/Add External JAR File](#) – Learn how to import an External JAR file to Android Studio.
- [Change API SDK Level](#) – Learn how to change the SDK level of API in Android Studio.
- [Create/Add New Package Inside Src Folder](#) – Learn to know about creating new package inside Src Folder in Android Studio.
- [Creating Folders for Adding Different Resolution Images](#) – Learn how add folder for adding images with different resolution in Android Studio.
- [Create An Interface](#) – Learn how to create Interface in Android Studio.

<https://abhiandroid.com/androidstudio/>

Android tutorial

Android Studio Tutorial

AS APPS
Contains ads

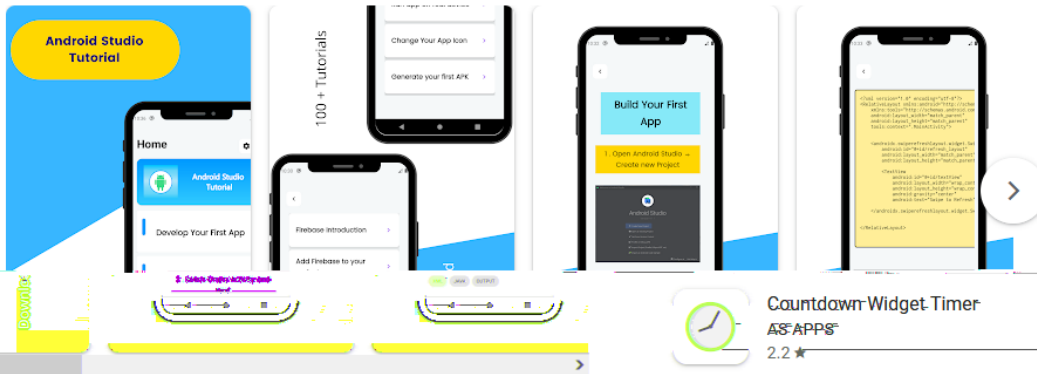
4.0★
454 reviews

10K+
Downloads

E
Everyone

Install

Add to wishlist



Developer contact

More by AS APPS



Rock Paper Scissor & TicTacToe
AS APPS



Countdown-Widget-Timer
AS APPS
2.2★



Birthday Countdown
AS APPS

this app →

About

<https://play.google.com/store/apps/details?id=com.as.androidstudiotutorials&hl=en&gl=US>



First app

DOCUMENTATION

Filter

- App Basics
- Introduction
- Build your first app
 - Overview**
 - Create an Android project
 - Run your app
 - Build a simple user interface

Android Basics > Docs > Guides

Czy te wskazówki były pomocne?  

Build your first app

This section describes how to build a simple Android app. First, you learn how to create a "Hello, World!" project with Android Studio and run it. Then, you create a new interface for the app that takes user input and switches to a new screen in the app to display it.

Before you start, there are two fundamental concepts that you need to understand about Android apps: how they provide multiple entry points, and how they adapt to different devices.

entry points

...ion of components that can be invoked individually. For example, an *activity* is a user interface (UI).

...ser taps your app's icon. You can also direct the user to an activity from elsewhere, ... from a different app.

...anager, allow your app to perform background tasks without a UI.

...n learn more about the other app components at [Application fundamentals](#).

t devices

- Start another activity
- App fundamentals
 - App resources
 - App manifest file
- Devices
 - Device compatibility
 - Large screens — tablets, Chromebooks, foldables
 - Wear
 - Android TV
 - Android for Cars
 - Chrome OS devices

multiple entry points, and how they a

Apps provide multiple

Android apps are built as a combination of different types of app component that provides

The "main" activity starts when the user taps the app icon, such as from a notification or even from another app.

Other components, such as *WorkManager*, allow your app to perform background tasks without a UI.

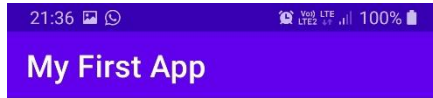
After you build your first app, you can learn more about the other app components at [Application fundamentals](#).

Apps adapt to different

<https://developer.android.com/training/basics/firstapp>

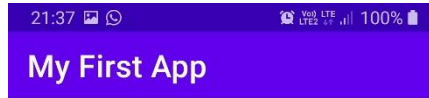
<https://javastart.pl/baza-wiedzy/android/pierwsza-aplikacja>

First app



Enter a messa

SEND

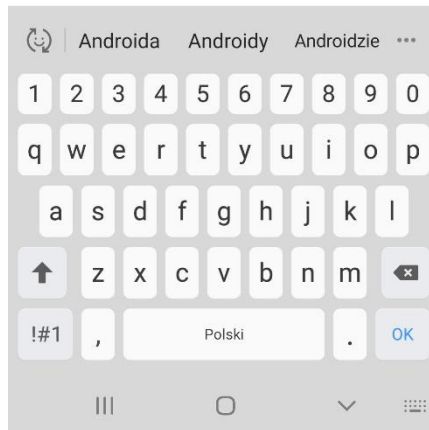


Android

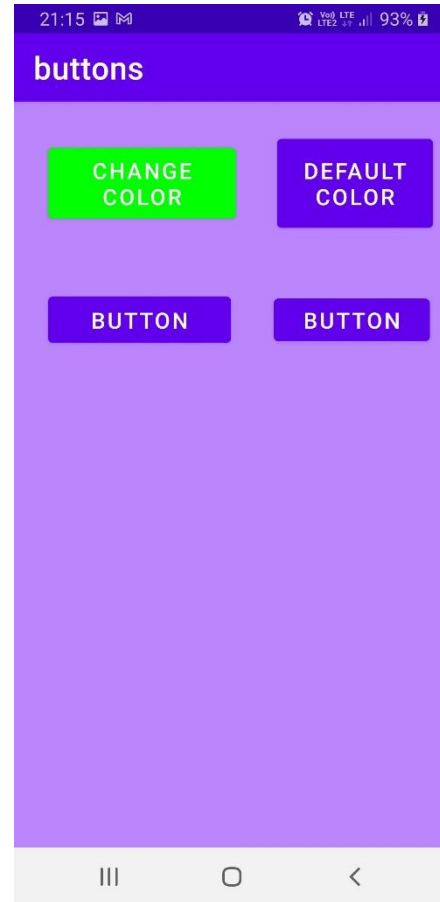
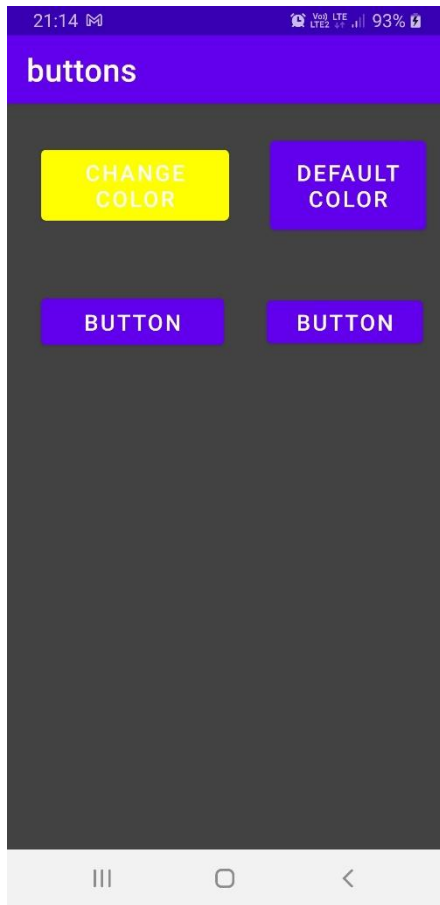
SEND



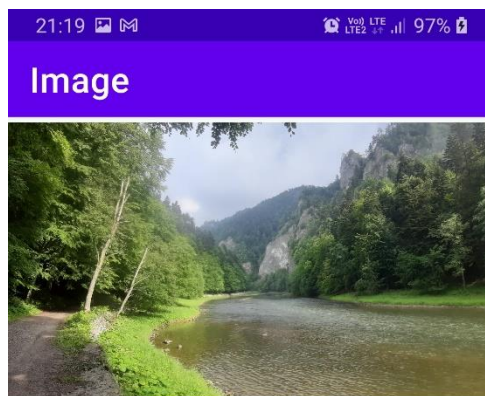
Android



buttons



image

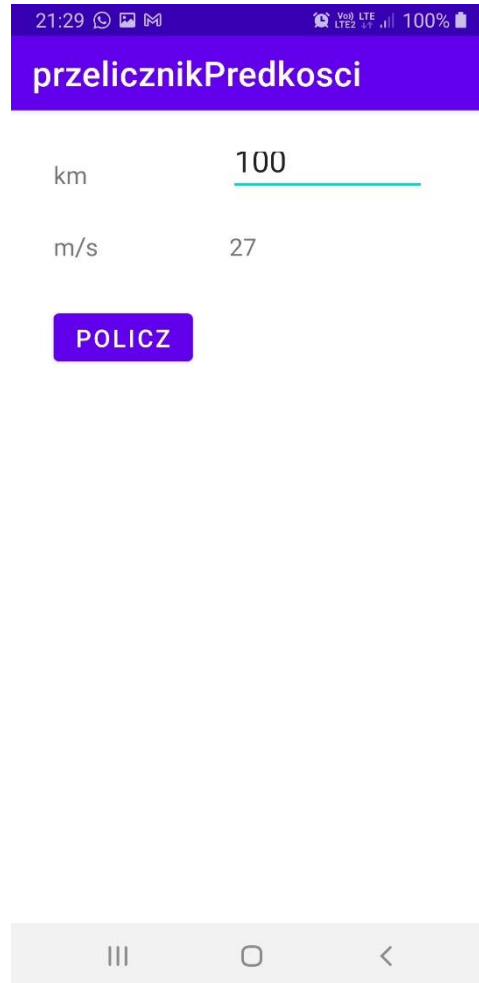


rzeka

Dunajec



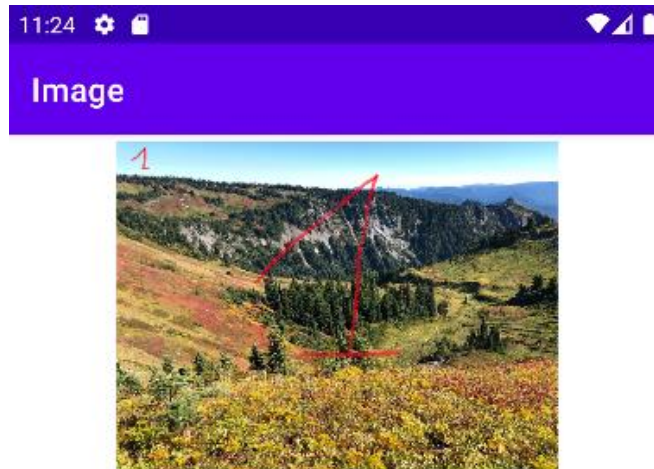
przelicznikPredkosci



password



gallery



GÓRY

FORWARD

BACKWARDS

Który obraz wyświetlić?

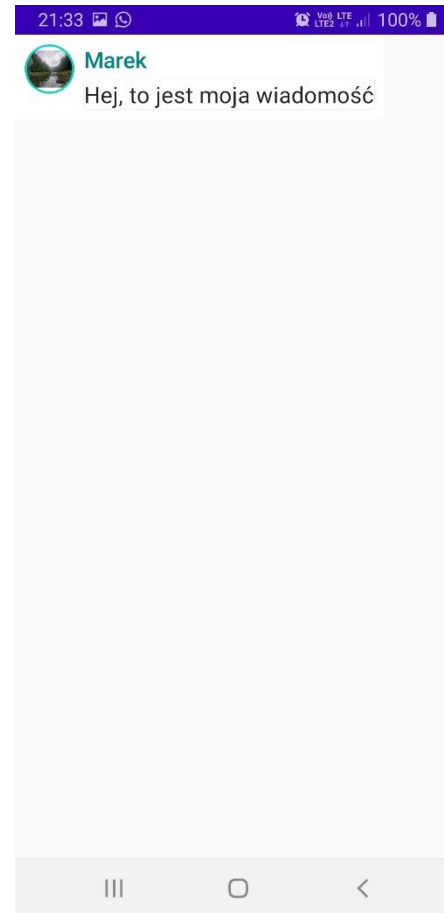
1

Tło niebieskie?



Jetpack Compose Tutorial

[Jetpack Compose Tutorial](#)



// first app

<https://developer.android.com/training/basics/firstapp>

// samples

<https://developer.android.com/samples?language=kotlin&skill=beginner>

<https://developer.android.com/guide/components/fundamentals>

<https://developer.android.com/guide/practices/compatibility>

<https://developer.android.com/courses/android-basics-kotlin/course>

// **Build a Responsive UI with ConstraintLayout**

<https://developer.android.com/develop/ui/views/layout/constraint-layout>