aplikacje mobilne Android Studio XML

tutorial version 1.2

tworzenie aplikacji mobilnej

różne technologie tworzenia aplikacji w Android Studio

layout w plikach XML (starsza technologia)

Jetpack Compose

tworzenie aplikacji w Android Studio: layout w plikach XML



Android Studio XML dokumentacja



Android Studio XML nowy projekt



Android Studio XML nowy projekt

Kew Project		\times
Empty Views Activity		
Creates a new empty activity		
<u>N</u> ame	button_xml	
Package name	com.example.button_xml	
Save location	C:\Users\Kasia\AndroidStudioProjects\button_xml	
<u>L</u> anguage	Kotlin ~	
Minimum SDK	API 26 ("Oreo"; Android 8.0)	
	Your app will run on approximately 95,4% of devices. Help me choose	
Build configuration language 🕗	Kotlin DSL (build.gradle.kts) [Recommended]	
The application name for most of the application name for most	st apps begins with an uppercase letter	
	Previous Next Cancel	<u>F</u> inish

AndroidManifest.xml

AndroidManifest.xml zawiera informacje, które Android używa do zarządzania aplikacją, takie jak:

- Deklaracja komponentów aplikacji (aktywności, usługi, odbiorniki rozgłoszeniowe, itp.).
- Zarządzanie uprawnieniami (jak dostęp do Internetu, kamery, lokalizacji).
- Określenie ustawień aplikacji (np. ikona aplikacji, motyw, nazwę aplikacji).
- Definiowanie zasad uruchamiania aplikacji (np. która aktywność jest główną).
- Konfiguracja wersji aplikacji (numer wersji, kod wersji).

🛎 📄 🔣 button_xml 🗸 Version control 🗸	e ۴
	✓> activity_main.xml @ MainActivity.kt M AndroidManifest.xml ×
	<pre>1 <?xml version="1.0" encoding="utf-8"?></pre>
□ anifests	<pre>2 <manifest <="" pre="" xmlns:android="http://schemas.android.com/apk/res/android"></manifest></pre>
AndroidManifest xml	<pre>3 xmlns:tools="http://schemas.android.com/tools"></pre>
	4
	5 <application< th=""></application<>
Contexample.button_xmi	6 android:allowBackup="true"
C MainActivity	7 android:dataExtractionRules="@xml/data_extraction_rules"
i com.example.button_xml (androidTest)	<pre>8 android:fullBackupContent="@xml/backup_rules"</pre>
> i com.example.button_xml (test)	9 📥 android:icon="@mipmap/ic_launcher"
✓ C⊇ res	10 android:label="button_xml"
v index drawable	11 android:roundIcon="@mipmap/ic_launcher_round"
ic_launcher_background.xml	12 android:supportsRtl="true"
ic_launcher_foreground.xml	13 android:theme="@style/Theme.Button_xml"
> 💿 layout	14 tools:targetApi="31">
> 💿 mipmap	15 <activity< th=""></activity<>
> i values	16 android:name=".MainActivity"
> ি xml	17 android:exported="true">
	18 <intent-filter></intent-filter>
) @ Gradle Scripts	19 <action android:name="android.intent.action.MAIN"></action>
	<pre>21 <category android:name="android.intent.category.LAUNCHER"></category></pre>
	22
	23
	24
	26

MainActivity.kt

W pliku MainActivity.kt definiujemy główną aktywność aplikacji Android, czyli klasę, która reprezentuje ekran, z którym użytkownik będzie miał do czynienia na początku (jeśli jest to aktywność startowa). Zwykle jest to miejsce, w którym zaczyna się logika interakcji użytkownika w aplikacji.

2005	BX button_xml ~ Version control ~	[≟ Pixel 9 API 35 ∽
	Android \sim \bigcirc \Diamond \times \vdots $-$	activity_main.xml Image: MainActivity.kt Image: MainActivity.kt Image: MainActivity.kt Image: MainActivity.kt
<u>°</u> ⊿ 	 imanifests imanifests imanifests imanifest.xml iman	<pre>1 package com.example.button_xml 2 3 import android.os.Bundle 4 import androidx.activity.enableEdgeToEdge 5 import androidx.appcompat.app.AppCompatActivity 6 import androidx.core.view.ViewCompat 7 import androidx.core.view.WindowInsetsCompat </pre>
	 > in com.example.button_xml (test) 	° 9 ▷ class MainActivity : AppCompatActivity() {
	 ic_launcher_background.xml ic_launcher_foreground.xml ic_launcher_foreground.xml io layout io mipmap io values io xml io xml io res (generated) io Gradle Scripts 	10 Image: String of the system Bars.left, system Bars.top, system Bars.right, system Bars.bottom) 11 Image: System Bars of the system Bars.top, system Bars.right, system Bars.bottom) 12 Image: System Bars of the system Bars.top, system Bars.right, system Bars.bottom) 13 Image: System Bars of the system Bars.top, system Bars.right, system Bars.bottom) 14 Image: System Bars.left, system Bars.top, system Bars.right, system Bars.bottom) 16 Image: System Bars.left, system Bars.top, system Bars.right, system Bars.bottom) 17 Image: System Bars of the system Bars.top, system Bars.right, system Bars.bottom) 18 Image: System Bars of the system Bars.top, system Bars.right, system Bars.bottom) 19 Image: System Bars.top, system Bars.bottom) 19 Image: System Bars.top, system Bars.
	> & Gradle Scripts	20 }

res



□ button_xml > □ app > src > □ main > res > layout > � activity_main.xml

2:1 LF UTF-8 🤹 🗉 4 spaces 🗗

W Androidzie, Gradle to system budowania, który automatycznie zarządza procesem kompilacji aplikacji. Gradle Scripts to pliki konfiguracyjne, które określają, jak budować, testować i pakować aplikację Androidową.

3X button_xml < Version control < 🔓 Pix ⊕ ≎ × : -G MainActivity.kt M AndroidManifest.xml Android ~ activity_main.xml $\mathcal{Z}_{\mathcal{R}}^2$ build.gradle.kts (:app) \times You can use the Project Structure dialog to view and edit your project configuration Chapp 2 manifests plugins { 1 ... M AndroidManifest.xml alias(libs.plugins.android.application) kotlin+java alias(libs.plugins.kotlin.android) 3 ~ in com.example.button_xml 6 G MainActivity 5 6 android { > com.example.button_xml (androidTest) namespace = "com.example.button_xml" 7 > > com.example.button_xml (test) 8 compileSdk = 35 ✓ □ res 9 > 🗟 drawable defaultConfig { 10 Iayout applicationId = "com.example.button_xml" </> activity_main.xml minSdk = 26> 💿 mipmap targetSdk = 34 > lon values 14 versionCode = 1 > 🖻 xml versionName = "1.0" res (generated) 17 testInstrumentationRunner = "androidx.test.runner.AndroidJUnitRunner" C Gradle Scripts } 18 E build.gradle.kts (Project: button_xml) 19 C build.gradle.kts (Module :app) buildTypes { 20 ≡ proguard-rules.pro (ProGuard Rules for ":app") release { gradle.properties (Project Properties) isMinifyEnabled = false gradle.properties (Global Properties) proguardFiles(gradle-wrapper.properties (Gradle Version) 24 getDefaultProguardFile("proguard-android-optimize.txt"), Ibs.versions.toml (Version Catalog) 25 "proguard-rules.pro" Iocal.properties (SDK Location) 26) € settings.gradle.kts (Project Settings) 27 T 28 compileOptions { \triangleright 29 30 sourceCompatibility = JavaVersion.VERSION_11 Q 31 targetCompatibility = JavaVersion.VERSION_11

Gradle Scripts

Układ okien



kopiowanie kodu z pałacowej strony



opracowania

aplikacjeMobilneAndroidStudioXML_1.0.pdf

przydatne linki

Android developer XML //dokumentacja dotycząca pisania programów w Android Studio (XML)

napisz to sam //spróbuj nie zagladać do kodu



☆

MainActivity.kt

kopiowanie kodu z pałacowej strony

tutaj musi się znaleźć nazwa aplikacji

package com.example.kliker xml

import android.graphics.Bitmap import android.graphics.Canvas import android.graphics.Color import android.graphics.Paint import android.os.Bundle import android.widget.ImageView import androidx.activity.enableEdgeToEdge import androidx.appcompat.app.AppCompatActivity import androidx.core.view.ViewCompat import androidx.core.view.WindowInsetsCompat

class MainActivity : AppCompatActivity() {

private lateinit var bitmap: Bitmap private lateinit var canvas: Canvas private lateinit var paint: Paint

override fun onCreate(savedInstanceState: Bundle?) { super.onCreate(savedInstanceState) enableEdgeToEdge() setContentView(R.layout.activity_main) ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets -> val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars()) v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom) insets 3

// metoda statyczna, która tworzy nową instancję obiektu Bitmap. // parametry: // szerokość i wysokość bitmapy // ARGB 8888 oznacza, że każdy piksel jest reprezentowany przez 4 składowe kolorów: // Alfa (A) - przezroczystość, Czerwony (R), Zielony (G), i Niebieski (B), // z każdym składnikiem zajmującym 8 bitów (czyli wartości od 0 do 255). bitmap = Bitmap.createBitmap(200, 300, Bitmap.Config.ARGB_8888) // tworzenie kanwy canvas = Canvas(bitmap) // referencja do imageView val imageView = findViewById<ImageView>(R.id.imageView) // Inicjalizacja farby paint = Paint().apply {

isAntiAlias = true color = Color.RED // Domyślny kolor owalu

rozmieszczenie elementów UI w aplikacji

LinearLayout

LinearLayout to kontener, który ustawia swoje elementy (widoki) w jednym rzędzie (horyzontalnie) lub w jednej kolumnie (wertykalnie), w zależności od ustawienia atrybutu android:orientation

RelativeLayout

RelativeLayout pozwala na rozmieszczanie widoków w stosunku do innych widoków lub krawędzi kontenera. Każdy element może być umieszczony w odniesieniu do innych elementów, takich jak: góra, dół, lewa strona, prawa strona, środek.

GridLayout

GridLayout pozwala na rozmieszczanie widoków w tabeli (siatce) z określoną liczbą wierszy i kolumn. Umożliwia precyzyjne kontrolowanie pozycji elementów w siatce.

FrameLayout

FrameLayout to bardzo prosty układ, który pozwala na umieszczanie widoków w jednej "ramce". Zwykle jest używany, gdy chcesz wyświetlić jeden element na tle innych.

ConstraintLayout

ConstraintLayout to zaawansowany kontener układu, który pozwala na dokładne określenie pozycji elementów na ekranie, przy pomocy tzw. "ograniczeń" (constraints), czyli zależności między elementami.

TableLayout

TableLayout to układ, który pozwala na rozmieszczanie widoków w formie tabeli. Każdy wiersz tabeli to jeden TableRow, który może zawierać różne widoki (np. przyciski, teksty, obrazy).

ScrollView

ScrollView jest układem, który pozwala na przewijanie zawartości, jeśli całość nie mieści się na ekranie. Można go używać do przewijania widoków pionowych lub poziomych.

NestedScrollView

NestedScrollView to rozszerzenie ScrollView, które pozwala na lepszą obsługę widoków, które zawierają inne widoki przewijane. Jest użyteczny, gdy masz w jednym układzie inne widoki, które również mogą być przewijane.

ConstraintLayout



dodanie komponentu Button do aplikacji



wielkość przycisku dostosowuje się do jego treści

Constraint Widget

wielkość przycisku jest stała



Hardcoded string " buton "

ani.	\equiv BX button_xml \checkmark Version control \checkmark	
	Android ${\scriptstyle \curlyvee}$	Activity_main.xml × I strings.xml @ MainActivity.kt
<u>∩</u> ⊿ 	 Capp manifests kotlin+java Cares Car	<pre>1 <?xml version="1.0" encoding="utf-8"?> 2 @ <androidx.constraintlayout.widget.constraintlayout 3="" 4="" 5="" 6="" 7="" 8="" android:layout_height="match_parent" android:layout_width="match_parent" tools:context=".MainActivity" 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"> Hardcoded string "button".</androidx.constraintlayout.widget.constraintlayout></pre>
	> ≥ minimap > ≥ values <p< th=""><th>10 <textview< th=""> should use 11 android:id="@+id/textView" 12 android:layout_width="wrap_content" 13 android:layout_bight="wrap_content"</textview<></th></p<>	10 <textview< th=""> should use 11 android:id="@+id/textView" 12 android:layout_width="wrap_content" 13 android:layout_bight="wrap_content"</textview<>
7 데	 > Intermes (2) > Intermet xml Intermet xml I	 Hardcoding text attributes directly in layout files is bad for several reasons: Pilayout_constraintEnt_totoOf="parent" app:layout_constraintEnt_totoof="parent" app:layout_constraintVertical_bias="0.196" /> cButton android:layout_width="map_content" android:layout_width="map_content" android:layout_meight="map_content" android:layout_constraintEnt_totoof="parent" android:layout_neight="map_content" android:layout_meight="map_content" android:layout_meight="map_content" android:layout_meight="map_content" android:layout_meight="map_content" android:layout_constraintEnt_totoof="parent" android:layout_constraintEnt_totoof="parent" android:layout_constraintEnt_totoof="parent" android:layout_meight="map_content" android:layout_meight="map_content" android:layout_constraintEnt_totoof="parent" app:layout_constraintEnt_totoof="parent" app:
40 ⊆ ∰ ♥ ♥ ♥	Problems File 1 Project Errors Layout and C Image: Second	31 app:layout_constraintVertical_bias="0.13999999" /> 32 33 Walffiers 2 WudioProjects\button_xml\app\src\main\res\layout 2 problems Hardcoded text Hardcoded string "button", should use @string resource 1 1 2 2 3 (androidx.constraintlayout.widget.constraintlayout xulps:android="http://schemas.android.com/apk/res/android" The provide the provi

@string resource

 \times

Cancel

~		(b act	ivity main yml X () strings yml - C Main Astivity kt	 Extract Reso 	urce
		2	<pre>sandpoidx constraintlayout widget Constraintlayout xmlns:andpoid="http://schemas andpoid_com/a</pre>	Resource <u>n</u> ame:	button
Δ		8	<pre>tools:context=" MainActivity"> tools:context=" MainActivity"> </pre>		
	> 🗋 manifests	0			button
	> 🗋 kotlin+java	10	<textview< td=""><td>Resource value:</td><td></td></textview<>	Resource value:	
	✓ □ res	11	android·id="##id/textView"		
	> 🖻 drawable	12	android:lavout width="22do"		
	 Iayout 	13	android:layout height="14dp"	Source set:	main src/main/res
	activity_main.xml	14	android:text="Hello World!"		
	> 💿 mipmap	15	app:lavout_constraintBottom_toBottomOf="parent"	Eile name:	strings.xml
	values	16	app:layout_constraintEnd_toEnd0f="parent"	Create the resou	rce in directories:
		17 app:layout_constraintStart_toStartOf="parent"			
	strings.xml	18 app:layout_constraintTop_toTopOf="parent"		+ - 🗹 🖻	
	> Themes (2)	19	app:layout_constraintVertical_bias="0.196" />	🗸 values	
		20 21 <button< td=""><td>values-night</td><td></td></button<>		values-night	
				- Values hight	
	Lares (generated)	22	android:id="@+id/button2"		
	> &7 Gradie Scripts	23	android:layout_width="wrap_content"		
		24	android:layout_height="wrap_content"		
		25	android:layout_marginTop="32dp"		
		26	<pre>g android:text= button</pre>		
		27	app:layout_constr Hardcoded string " hutton" should use @string resource More (Ctrl+F1)		
		28	app:layout_constr		
		29	app:layout_constr Extract string resource Alt+Shift+Enter More actions Alt+Enter		
		30	app:layout_constraintrop_toportomOf="@+id/textView"		
		31	app:layout_constraintVertical_bias="0.13999999" />		

2

...

BX button_xml < Version control <	BX button_xml // Version co	BX button_xml - Version control -		
Andraid	Ch astivity main yml	○ X : - ♦ activity_main.xml × ♦ strings.xml @ MainActivity.kt		
Android ~	activity_main.xmi v strings.xmi × Cr MainActivity.kt	2 <androidx.constraintlayout.widget.constraintlayout xmlns:android="</p"></androidx.constraintlayout.widget.constraintlayout>		
	Edit translations for all locales in the translations aditor Committee aditor	10 <textview 14 angroup.text- netto worku:</textview 		
	Euclidaristations for an locales in the translations editor. ···· > External kollin+java	15 app:layout_constraintBottom_toBottomOf="parent"		
> manifests	1 <resources></resources>	16 app:layout_constraintEnd_toEndOf="parent"		
> 🗋 kotlin+java	2 ● <string name="app_name">button_xml</string>	17 app:layout_constraintStart_toStartOf="parent"		
✓ □ res	3 <string name="button">button</string> > > > > > > > > > > > > > > > > > >	<pre>18 app:layout_constraintTop_toTopOf="parent"</pre>		
> 🗟 drawable	✓ Zerrange and Sector Poet	19 app:layout_constraintVertical_bias="0.196" />		
	> imipmap	20		
	✓ invalues	21 <button< td=""></button<>		
>activity_main.xml	colors.xml	22 android:layout width="wpan content"		
> 🖻 mipmap	< <u>⟩</u> strings.xm	24 android:layout_width="wrap_content"		
✓ lin values	> in themes (2)	25 android:layout_marginTop="32dp"		
	> 🖻 xml	26 android:text="@string/button"		
COLORS.XIII	□ res (generated)	27 app:layout_constraintBottom_toBottomOf="parent"		
>strings.xml	> 😂 Gradle Scripts	<pre>28 app:layout_constraintEnd_toEndOf="parent"</pre>		
> themes (2)		<pre>29 app:layout_constraintStart_toStartOf="parent"</pre>		
> ি xml		30 app:layout_constraintTop_toBottomOf="@+id/textView"		
		<pre>31 app:layout_constraintVertical_bias="0.13999999" /></pre>		
L ieres (generated)		32		
> 🖾 Gradle Scripts		<pre>>>> </pre>		

Button – gotowy layout



Button – aplikacja

*	\equiv BX button_xml \vee Version control \vee	🗋 samsung SM-A750FN 🗸 📩 app 🗸 💽 🔅 🔲		
	Android ~	◇ activity_main.xml	12:51	voi) lte+ lte2 ++ ,iii 100% 🖿
<u>∩</u>	<pre> Comparison of the second se</pre>	<pre>visit class MainActivity : AppCompatActivity() { visit class MainActivity : AppCompatActivity : AppCompA</pre>	Hello World	!!
	Logcat Logcat × +			
	Samsung SM-A750FN (320083cc47881665) Android	y → y package:mine		
4 - - - - - - - - - - - - -	2025-03-31 12:54:28.934 21889-21889 Inpt 2025-03-31 12:54:29.927 21889-21889 View 2025-03-31 12:54:29.990 21889-21889 View 2025-03-31 12:54:29.990 21889-21889 View 2025-03-31 12:54:29.994 21889-21889 View 2025-03-31 12:54:37.018 21889-21889 View 2025-03-31 12:54:37.117 21889-21889 View 2025-03-31 12:54:37.120 21889-21889 View 2025-03-31 12:54:38.701 21889-21889 View 2025-03-31 12:54:38.706 21889-21889 View 2025-03-31 12:54:38.768 21889-21889 View 2025-03-31 12:54:38.768 21889-21889 View 2025-03-31 12:54:38.768 21889-21889 Syst	UtTransportcom.example.button_xmlDInput channel destroyed: 'ClientS', fd=70VRootImActivitycom.example.button_xmlDViewPostIme pointer 0VRootImActivitycom.example.button_xmlDViewPostIme pointer 1rem.outcom.example.button_xmlDViewPostIme pointer 0VRootImActivitycom.example.button_xmlDViewPostIme pointer 1vRootImActivitycom.example.button_xmlDViewPostIme pointer 0vRootImActivitycom.example.button_xmlDViewPostIme pointer 1rem.outcom.example.button_xmlDViewPostIme pointer 0vRootImActivitycom.example.button_xmlDViewPostIme pointer 1rem.outcom.example.button_xmlDViewPostIme pointer 0vRootImActivitycom.example.button_xmlDViewPostIme pointer 0vRootImnActivitycom.example.button_xmlDViewPostIme pointer 1rem.outcom.example.button_xmlDViewPostIme pointer 1rem.outcom.example.button_xmlDViewPostIme pointer 1rem.outcom.example.button_xmlIPrzycisk zostat kliknięty!	III O	<

arkusz "Domek w górach"



zmiana ConstraintLayout na LinearLayout

2005	■ DW domekWGorach_xml ∨ Version control ∨	
	✓> activity_main.xml ×	
<u>₀</u>	<pre>1 <?xml version="1.0" encoding="utf-8"?> 2 @ <androidx.constraintlayout.widget.constraintlayout <br="" xmlns:android="http: 3</th><th>//schemas.android.com/apk/res/android">zaznaczyć jak na screenie i zacząć wpisywać Linear</androidx.constraintlayout.widget.constraintlayout></pre>	
× ••• •••	<pre> Ow domekWGorach_xml < Version control < Activity_main.xml × @ MainActivity.kt <pre></pre></pre>	pojawi się podręczne menu z wybranym LinearLayout, klikamy Enter
<u>∩</u> _	<pre>1 <?xml version="1.0" encoding="utf-8"?> 2 @ <linearlayout 3<="" th="" xmlns:android="http://schemas.android.com/apk/res/android"><th>Layout został zmieniony na LinearLayout</th></linearlayout></pre>	Layout został zmieniony na LinearLayout
	<pre>7 android:layout_height="match_parent" 8 tools:context=".MainActivity"></pre>	LinearLayout YouTube

dodawanie komponentów w activity_main.xml



wszystkie komponenty można wpisywać "z palca" w pliku activity_main.xml

activity_main.xml

> act	ivity_main.xml × 🕞 MainActivity.kt
1	xml version="1.8" encoding="utf-8"?
2 G	<linearlayout <="" th="" xmlns:android="http://schemas.android.com/apk/res/android"></linearlayout>
3	xmlns:app="http://schemas.android.com/apk/res-auto"
á.	xmlns:tools="http://schemas.android.com/tools"
5	android:id="@+id/main"
	android:layout_width="match_parent"
7	android:layout_height="match_parent"
8	android:paddingTop="20dp"
9	android:orientation="vertical"
3	tools:context=".MainActivity">
L	
2	wszystkie napisy należy dodać do res/values/strings.xml
	<textview< td=""></textview<>
á.	android:id="@+id/textViewDomekWGorach"
5	android:layout_width="match_parent"
	android:layout_height="wrap_content"
	android:textSize="30sp"
	android:text="Domek w górach"
	/>
1	
	zdjecie agrestu należy pobrać poprzez Resource Manager
2	<imageview< td=""></imageview<>
	android:id="@+id/imageViewAgrest"
	android:layout_width="match_parent"
	android:layout_height="306dp"
	app:srcCompat="@drawable/agrest" />
7	
8	android:background="#FF008080"</td
9	dla LinearLayout i button nie działa, z poziomu kodu ok>
9	<linearlayout< td=""></linearlayout<>
	android:id="@+id/btnLayout"
2	android:layout_width="match_parent"
5	android:layout_height="wrap_content"
	android:orientation="horizontal">
5	
5	<button< td=""></button<>
7	android:id="@+id/btnPolub"
8	android:layout_width="wrap_content"
9	android:layout_height="wrap_content"
8	android:textColor="@color/white"
1	android:layout_weight="1"
2	android:text="POLUB"/>
ŝ	
4	<button< td=""></button<>
5	android:id="@+id/btnUsun"
	android:layout_width="wrap_content"
	android:layout_height="wrap_content"

android:textColor="@color/white" android:layout_weight="1" android:text="USUN"/>

52	
53	< <mark>Button</mark>
54	android:id="@+id/btnZapisz"
55	android:layout_width="wrap_content"
50	android:layout_height="wrap_content"
57	
38	android:textColor="@color/white"
39	android:layout_weight="1"
68	android:text="ZAPISZ"/>
01	
62	
63	<textview< th=""></textview<>
64	android:id="@+id/textViewPolubienia"
65	android:layout_width="match_parent"
00	android:layout_height="wrap_content"
67	android:gravity="end"
68	android:text="0 polubień"
69	/>
78	szara linia
71	<view< th=""></view<>
72	android:layout_width="match_parent"
73	android:layout_height="1dp"
74	android:background="#FF808080" />
75	<textview< th=""></textview<>
70	android:layout_width="match_parent"
77	android:layout_height="wrap_content"
78	android:textStyle="bold"
79	android:text="Opis"/>
89	<textview< th=""></textview<>
81	android:layout_width="match_parent"
82	android:layout_height="wrap_content"
83	android:textColor="#808080"
84	android:text="Odwiedź komfortowy domek w Sudetach, blisko do szlaków"/>

</LinearLayout>

MainActivity.kt

Activity_main.xml package com.example		_main.xml	🕞 MainActivity.kt 🛛 🛛	
		ckage com.example	.domekwgorach_xml	
2				
3	im	port android.gra	hics.Color	
4	im	port android.os.	lundle	
5	im	port android.wid	jet.Button	
٥	im	port android.wid	et.LinearLayout	
7	im	port android.wid	et.TextView	
8	im	port androidx.ac	ivity.enableEdgeToEdge	
9	im	port androidx.ap	compat.app.AppCompatActivity	
10				
11	im	port androidx.co	e.view.ViewCompat	
12	im	port androidx.co	e.view.WindowInsetsCompat	
13				
14 ▷	♦ cl	ass MainActivity	: AppCompatActivity() {	
15	6	override fun o	Create(savedInstanceState: Bundle?) {	
10		super.onCr	ate(savedInstanceState)	
17		enableEdge	"oEdge()	
18		setContent	/iew(R.layout. <u>activity_main</u>)	
19		ViewCompat	<pre>setOnApplyWindowInsetsListener(findViewById(R.id.<u>main</u>)) { v, insets -></pre>	
28		val sy	temBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())	
21		v.setP	dding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)	
22		insets		
23		}		
24				
25		val btnPol	<pre>b: Button = findViewById<button>(R.id.<u>btnPolub</u>)</button></pre>	
20		val btnZap	<pre>sz: Button = findViewById<button>(R.id.<u>btnZapisz</u>)</button></pre>	
27		val btnUsu	: Button = findViewById <button>(R.id.<u>btnUsun</u>)</button>	
28		val btnLay	ut: LinearLayout = findViewById <linearlayout>(R.id.<u>btnLayout</u>)</linearlayout>	
29				
38		val textVi	<pre>wPolubienia: TextView = findViewById<textview>(R.id.<u>textViewPolubienia</u>)</textview></pre>	
31				
32		// zmiana	oloru tła (tylko tak działa, w XML nie działa)	
		btnPolub.s	tBackgroundColor(Color.parseColor("#FF008080"))	
34		btnZapisz.	etBackgroundColor(Color.parseColor("#FF008080"))	
35		btnUsun.se	BackgroundColor(Color.parseColor("#FF008080"))	
30		btnLayout.	etBackgroundColor(Color.parseColor("#FF888888"))	
37			ain. Tak - 0	
38		var polubi	HIA: INT = 0	
39		var komuni	ar: string = ""	
48		ht-D-1-th	+0-01:-11:-+ <i>(</i>	
41		DTNPOLUD.S	stonolickListener į	
42		polubi	<u>nia = polubienia + 1</u>	
43		komun1	at = potopienia.tostring() + " potopien"	
44		textVi	wroludienia.setiex(<u>KomUnikat</u>)	
40		1		
40		htelleur, and	OnClickListenen J	
47		otnusuh.se	unglicklistener j	
48		polubic if ()	$\frac{1}{10000000000000000000000000000000000$	
-17		1+(pol)	utenta < 0)1	
38		1	<u>100160119</u> = 0	
51		1	at - selubionis toStains() + " selubici"	
52		KOMUNI	<pre>(at = polubienia.tostring() + " polubien" """""""""""""""""""""""""""""""""""</pre>	
53		textVi	wroluuienia.setlext(<u>komunikat</u>)	
34		1		
22	1	r		
50	3			
57				

Android Studio - update



?

<u>F</u>ile

2

...



Android Studio - update

Aby mieć najnowszą wersję Android Studio można również ściągnąć program ze strony https://developer.android.com/studio?hl=pl i wgrać na istniejącą wersję (przy instalacji jest opcja deinstalacji istniejącego środowiska). Po zainstalowaniu programu trzeba sprawdzić update (czy nie ma czegoś do zaktualizowania) oraz zainstalować nową wersje Gradle (po zainstalowaniu automatycznie wyskoczy okienko do instalacji najnowszej wersji)

Android Studio

Oficjalne środowisko IDE do tworzenia aplikacji na Androida przyspiesza teraz produktywność dzięki Gemini w Android Studio, czyli asystentowi kodowania opartemu na Al.

Pobierz pakiet nowych funkcji Meerkat w Android Studio 🕁

Przeczytaj informacje o wersji 🔳

