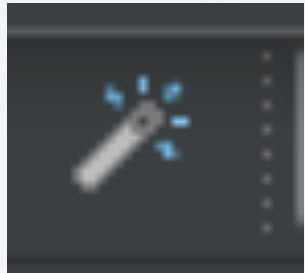


WSTĘP DO ANDROIDA

Laboratorium 3

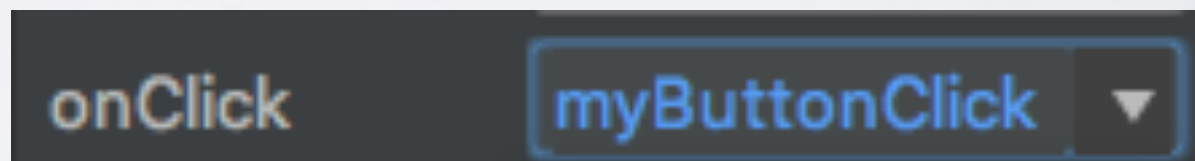
Systemy i aplikacje bez granic

|

- Tworzymy nowy projekt typu Empty Activity
- Poniżej Text View wstawiamy przycisk Button
- Tekst zmieniamy mu na Uruchom a id na myButton i wciskamy symbol 
- Text View zmieniamy id na statusText

|

- W atrybucie onClick przycisku wpisujemy myButtonClick



- W kodzie klasy aktywności dodajemy metodę myButtonClick

```
fun myButtonClick (v: View) {  
    statusText.text = "Uruchamiam"  
    showActivity()  
}
```

|

- Do klasy dodajemy pole
- Do projektu dodajemy nową aktywność Second Activity typu Empty Activity
- Wstawiamy do niej Text View o id statusText

```
val REQUEST_CODE=10000
```

|

- W MainActivity dodajemy metodę:

```
fun showActivity() {  
    val i=Intent( packageContext: this,SecondActivity::class.java)  
    i.putExtra( name: "Parametr", value: "Twoje dane")  
    startActivityForResult(i,REQUEST_CODE)  
}
```

- A w SecondActivity dodajemy do onCreate

```
val extras = intent.extras ?: return  
val message = extras.getString( key: "Parametr")  
statusText.text = message
```

- W SecondActivity dodajemy metodę:

```
override fun finish() {  
    val data = Intent()  
  
    data.putExtra( name: "returnString1", value: "Wiadomość dla ciebie")  
    setResult(Activity.RESULT_OK,data)  
  
    super.finish()  
}
```

- A w MainActivity:

```
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {  
    if ((requestCode==REQUEST_CODE)  
        && (resultCode== Activity.RESULT_OK)) {  
        if (data!=null) {  
            if (data.hasExtra( name: "returnString1")) {  
                statusText.text = data.extras.getString( key: "returnString1")  
            }  
        }  
    }  
}
```

|

- Uruchamiamy
- Do MainActivity dodajemy metodę:

```
fun showImplicitActivity() {  
    val i = Intent(Intent.ACTION_VIEW, Uri.parse(uriString: "http://www.predki.com"))  
    startActivity(i)  
}
```

- I podmieniamy jej wywołanie w obsłudze przycisku

II

- Utworzyć nowy projekt typu Empty Activity o nazwie BroadcastSample i aktywności `SendBroadcastActivity`
- Usunąć Text View i wstawić w to miejsce Button z tekstem „Wyślij”
- W atrybucie `onClick` wpisać „`broadcastIntent`”

||

- Aktualizujemy importy

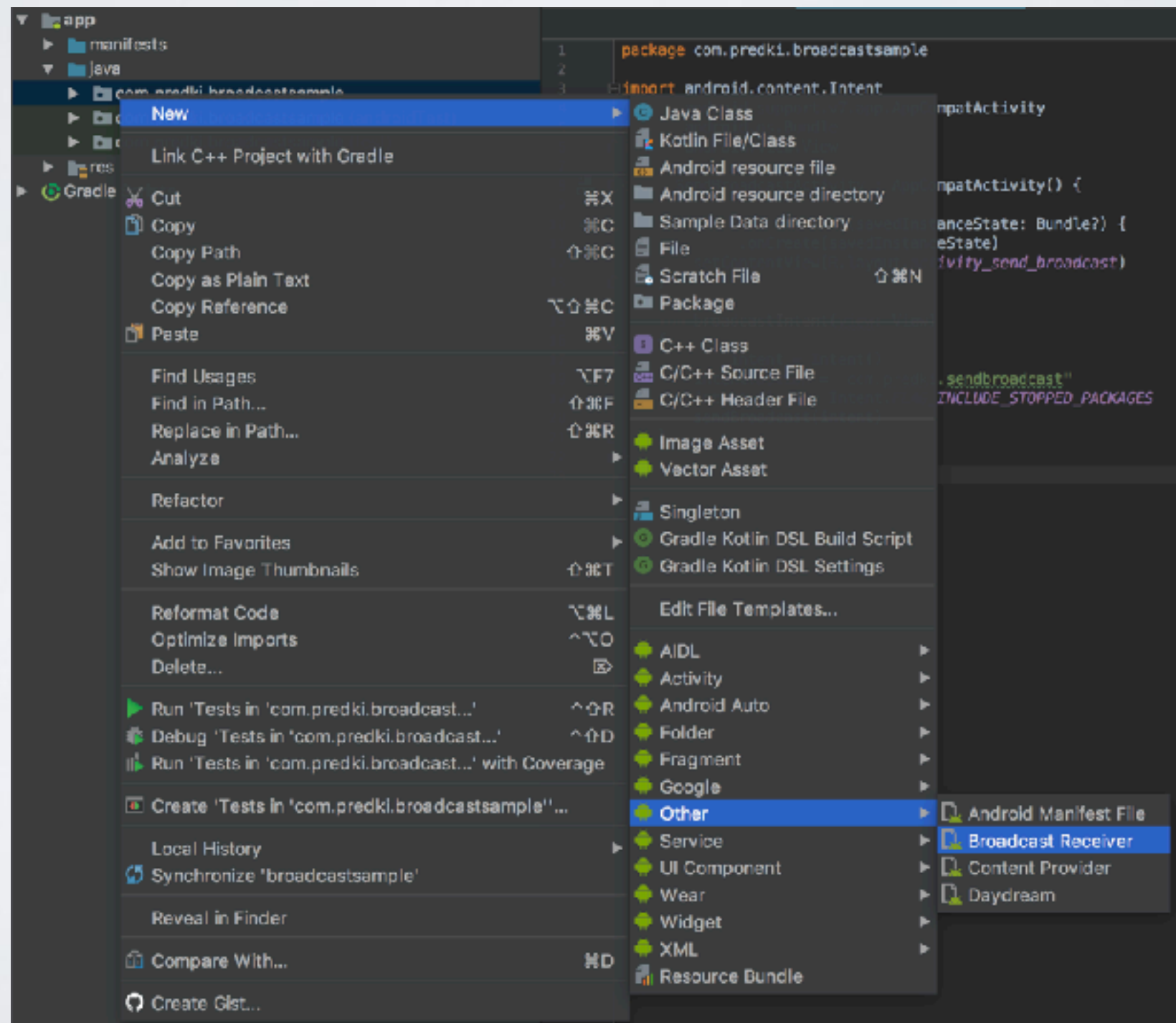
```
import android.content.BroadcastReceiver
import android.content.Intent
import android.content.IntentFilter
import android.support.v7.app.AppCompatActivity
import android.os.Bundle
import android.view.View
```

- Dodajemy metodę

```
fun broadcastIntent(view: View)
{
    val intent = Intent()
    intent.action = "com.predki.sendbroadcast"
    intent.flags = Intent.FLAG_INCLUDE_STOPPED_PACKAGES
    sendBroadcast(intent)
}
```

The figure consists of two vertical bars. The left bar is significantly taller than the right bar, indicating a much larger number of countries in the 'No' category compared to the 'Yes' category.

- Dodajemy nową klasę do projektu



||

- Modyfikujemy kod tej klasy

```
package com.predki.broadcastsample

import android.content.BroadcastReceiver
import android.content.Context
import android.content.Intent
import android.widget.Toast

class MyReceiver : BroadcastReceiver() {

    override fun onReceive(context: Context, intent: Intent) {
        Toast.makeText(context, text: "Broadcast Intent Detected.",
            Toast.LENGTH_LONG).show()
    }
}
```

||

- Dodajemy do klasy aktywności:

- pole

```
var receiver: BroadcastReceiver? = null
```

- metody

```
private fun configureReceiver() {  
    val filter = IntentFilter()  
    filter.addAction( action: "com.predki.sendbroadcast")  
    receiver = MyReceiver()  
    registerReceiver(receiver, filter)  
}  
  
override fun onDestroy() {  
    super.onDestroy()  
    unregisterReceiver(receiver)  
}
```

||

- i wywołanie metody `configureReceiver()` na końcu `onCreate`

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    setContentView(R.layout.activity_send_broadcast)  
    configureReceiver()  
}
```

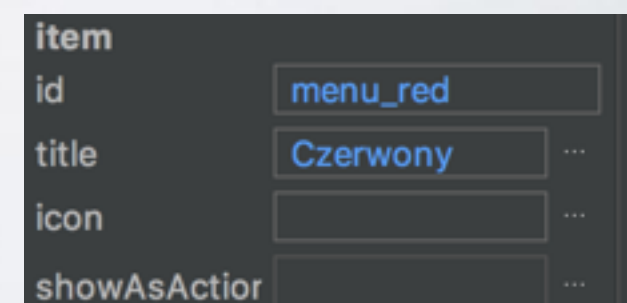
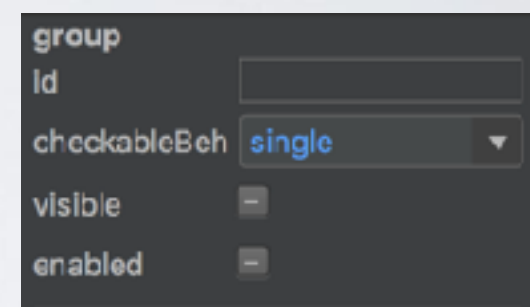
- Uruchamiamy

III

- Tworzymy nowy projekt typu Basic Activity o nazwie MenuExample
- Rozwijamy projekt app\res\menu i otwieramy menu_main.xml
- Usuwamy z Component Tree dotychczasowe menu Settings
- Z palety dodajemy nowe Menu

III

- Do menu dodajemy element Group
- Dla grupy ustawiamy atrybut checkableBehavior na single
- Do grupy dodajemy po kolei 4 elementy Menu Item
- Ustawiamy ich id i text, żeby odpowiadały 4 kolorom: czerwony, zielony, żółty, niebieski





```
<menu 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"
      tools:context="com.predki.menuapplication.MainActivity">
    <group android:checkableBehavior="single" >
        <item
            android:id="@+id/menu_red"
            android:title="Czerwony" />
        <item
            android:id="@+id/menu_green"
            android:title="Zielony" />
        <item
            android:id="@+id/menu_yellow"
            android:title="Żółty" />
        <item
            android:id="@+id/menu_blue"
            android:title="Niebieski" />
    </group>
</menu>
```



```

override fun onOptionsItemSelected(item: MenuItem): Boolean {
    when (item.itemId) {
        R.id.menu_red -> {
            if (item.isChecked)
                item.isChecked = false
            else
                item.isChecked = true
            layoutView.setBackgroundColor(android.graphics.Color.RED)
            return true
        }
        R.id.menu_green -> {
            if (item.isChecked)
                item.isChecked = false
            else
                item.isChecked = true
            layoutView.setBackgroundColor(android.graphics.Color.GREEN)
            return true
        }
        R.id.menu_yellow -> {
            if (item.isChecked)
                item.isChecked = false
            else
                item.isChecked = true
            layoutView.setBackgroundColor(android.graphics.Color.YELLOW)
            return true
        }
        R.id.menu_blue -> {
            if (item.isChecked)
                item.isChecked = false
            else
                item.isChecked = true
            layoutView.setBackgroundColor(android.graphics.Color.BLUE)
            return true
        }
        else -> return super.onOptionsItemSelected(item)
    }
}

```

- Zmień kolor na czerwony
- Zmień kolor na zielony
- Uruchom aplikację

_main