

# Kotlin & Android w Android Studio Giraffe (2022.3.1)

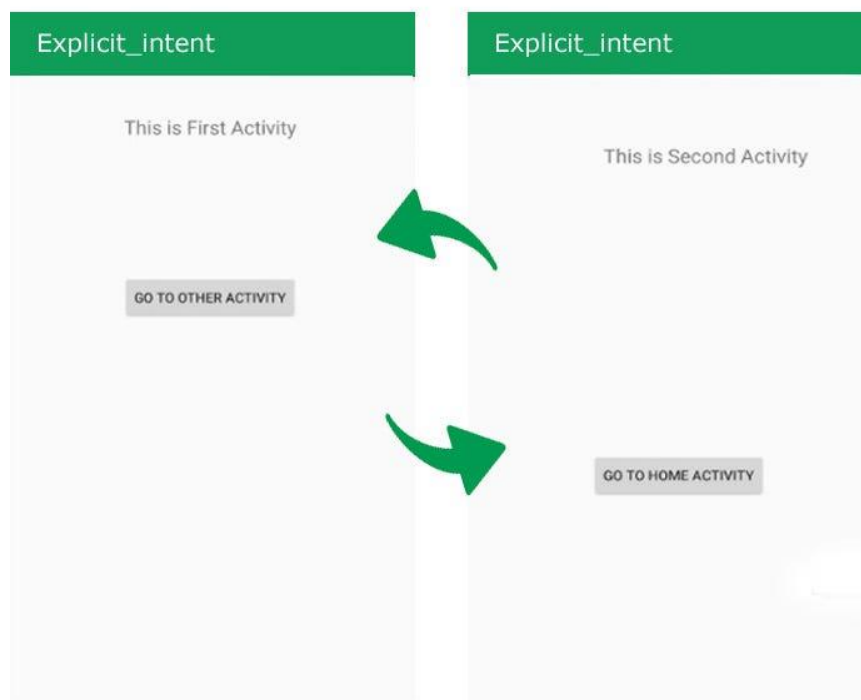
## Intencje, intencje jawne i niejawne

**Intencje** obok Aktywności są jednym z podstawowych komponentów z których zbudowane są Androidowe aplikacje. Mechanizm ten odpowiedzialny jest przede wszystkim za obsługę rozkazów wydawanych przez użytkownika. Za pomocą intencji możemy wprowadzić komunikację pomiędzy aplikacjami (lub mniejszymi komponentami, jak Usługi, Aktywności itp.). Jednak najważniejszym zadaniem tego komponentu jest uruchamianie odpowiednich aplikacji/Aktywności.

Intencje ze względu na typ uruchomienia dzielimy na:

- Jawne - czyli takie, w których jasno precyzujemy obiekt, który chcemy stworzyć.
- Niejawne - są to intencje, w których zawieramy informacje o tym co chcemy zrobić, bez podawania konkretnych klas, które mają to zrealizować.

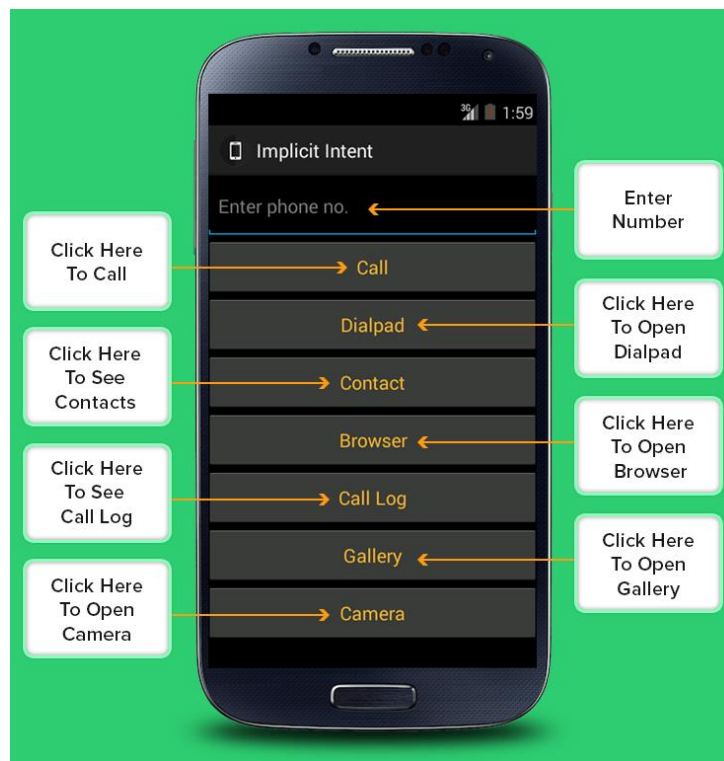
Intencje jawne wykorzystywane są głównie do komunikowania się komponentów aplikacji. Np. klikając przycisk bieżącej aktywności możemy podać adres innej aktywności, aby to właśnie ją uruchomić.



Intencje niejawne nie określają konkretnego komponentu, który ma zostać uruchomiony i pozwalają systemowi Android wskazać odpowiedni komponent na podstawie akcji i danych wywołującej intencji. Przykładami takich intencji mogą być:

- Otwieranie adresów URL lub określonych stron internetowych.
- Wysyłanie e-maili lub wiadomości SMS.
- Wykonywanie połączeń telefonicznych lub inicjowanie połączeń wideo.
- Udostępnianie treści na platformach mediów społecznościowych.

- Otwieranie map lub aplikacji nawigacyjnych dla określonych lokalizacji.



## Tworzenie intencji

### Intencje jawne

Aby utworzyć intencję jawną, jednym z argumentów konstruktora Intencji jest obiekt typu Class (np. `SecondActivity.class`) wskazujący na klasę, której obiekt chcemy utworzyć. Najprostszy przykład to:

```
// Executed in an Activity, so 'this' is the Context
val explicitIntent = Intent(packageContext: this, SecondActivity::class.java)
startActivity(explicitIntent)
```

Intencja, jak łatwo się domyślić, uruchomi aktywność `SecondActivity`.

### Intencje niejawne

W celu utworzenia intencji niejawnej musimy określić dwie informacje - co chcemy zrobić i na jakich danych chcemy tą czynność wykonać. W takim wypadku to system (za pomocą Filtrów Intencji) decyduje jaka Aktywność ma być uruchomiona. Przykład:

```
val intentGoogle = Intent(Intent.ACTION_VIEW, Uri.parse(uriString: "http://google.com"))
startActivity(intentGoogle)
```

W tym wypadku informujemy nasz system o tym, że chcemy zobaczyć dane (`Intent.ACTION_VIEW`) zapisane pod adresem URI (w tym wypadku adres URL strony).

Inny przykład intencji niejawnej:

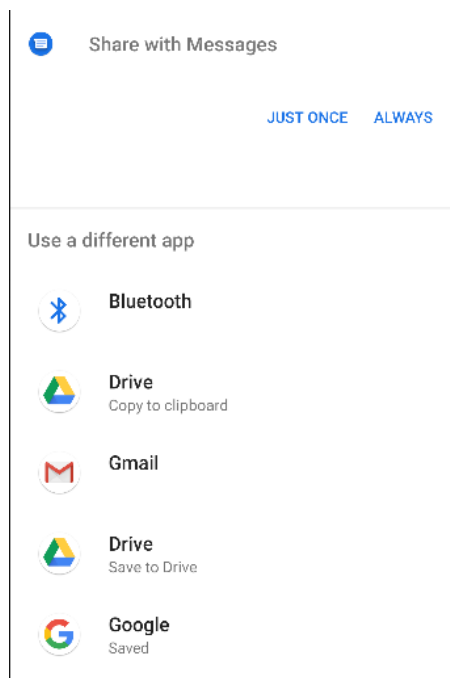
```

val textMessage = "My message to share."
// Create the text message with a string.
val sendIntent = Intent().apply { this: Intent
    action = Intent.ACTION_SEND
    putExtra(Intent.EXTRA_TEXT, textMessage)
    type = "text/plain"
}
// Try to invoke the intent.
try {
    startActivity(sendIntent)
} catch (e: ActivityNotFoundException) {
    // Define what your app should do if no activity can handle the intent.
}

```

W powyższym kodzie informujemy nasz system o tym, że chcemy udostępnić (wysłać) dane (*Intent.ACTION\_SEND*) typu tekst (*text/plain*).

Po wywołaniu funkcji *startActivity()* system sprawdza wszystkie zainstalowane aplikacje, aby określić, które z nich poradzą sobie z tego rodzaju intencją. Jeśli istnieje tylko jedna aplikacja, która może to obsłużyć, zostanie ona natychmiast otwarta i otrzyma intencję. Jeśli żadna aplikacja nie może obsłużyć wywołanej intencji, nasza aplikacja może przechwycić wyjątek *ActivityNotFoundException*. Jeśli intencję może zostać obsłużona przez wiele różnych aplikacji, system wyświetli okno dialogowe, dzięki czemu użytkownik może wybrać, której aplikacji chce użyć.



Przykłady innych zastosowań intencji możemy znaleźć w dokumentacji.

### **ZADANIE**

Korzystając z projektu z poprzedniego laboratorium, utwórz w klasie *MainActivity* nową intencję, która uruchomi wybrany przez Ciebie film na YouTube.

W tym celu dodaj nowy przycisk (za przyciskiem „Dzień dobry!”):

- zmień jego nazwę na *button\_YT* oraz styl na *Widget.Button*,

- utwórz w *strings.xml* stringa dla przycisku: `<string name="yt_btn">YouTube</string>` i ustaw go jako tekst wyświetlany,

- zmień wygląd przycisku: wypełnienie: czerwone, kolor tekstu: biały, rozmiar czcionki: 18sp, czcionka: pogrubiona, padding: 5dp.

Dodaj w *MainActivity.kt* kod odpowiedzialny za obsługę akcji kliknięcia na przycisku:

```
findViewById<Button>(R.id.button_YT).setOnClickListener { it: View!  
    val address = "http://youtube.com"  
    val intentYT = Intent(Intent.ACTION_VIEW, Uri.parse(address))  
    try {  
        startActivity(intentYT)  
    } catch (e: ActivityNotFoundException) {  
        var message = Toast.makeText(applicationContext, text: "Nie można uruchomić intencji!",  
            Toast.LENGTH_SHORT)  
        message.show()  
    }  
}
```

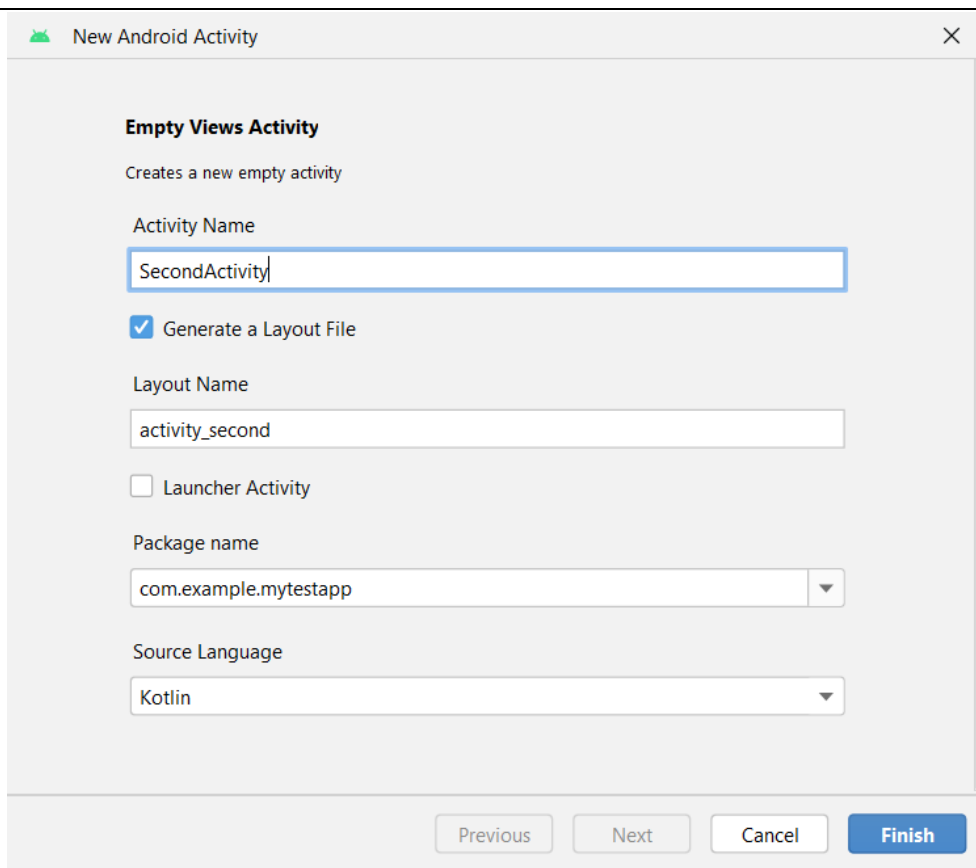
Uruchom aplikację i przetestuj działanie nowego przycisku.

W powyższym przykładzie wykorzystany został dwuparametrowy konstruktor *Intent(action, uri)*. Parametr *ACTION\_VIEW* mówi, że „chcemy zobaczyć” dane dostarczone do intencji i system Android ma je zwizualizować używając do tego celu najlepszej dostępnej aplikacji. Z kolei drugi parametr to dane w postaci adresu *URI*, który został sparsowany do stringa. Linijka *startActivity()* uruchamia aktywność adekwatną do naszej intencji.

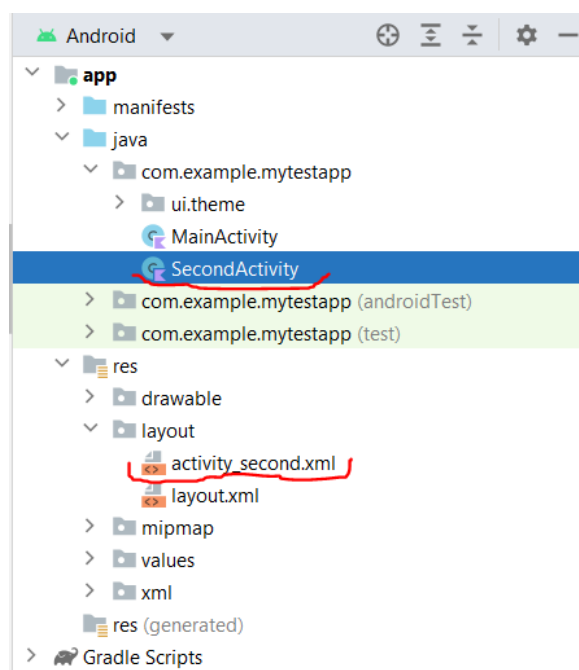
Nasz przycisk wywołuje niejawną intencję, uruchamiając film z podanego adresu YouTube.

### **ZADANIE**

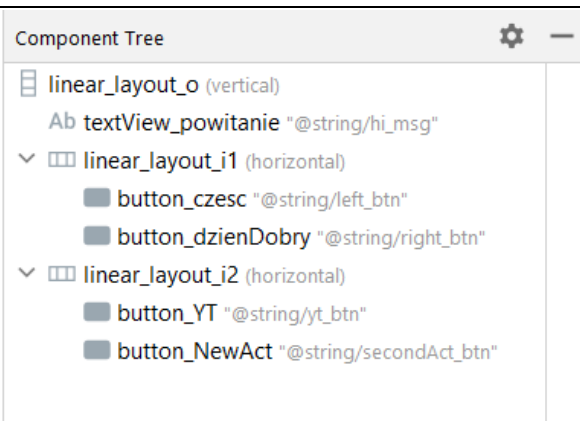
Dodaj nową aktywność *EmptyActivity* do projektu. W tym celu kliknij PPM na pakiecie, w którym znajduje się główna aktywność, a następnie wybierz *New -> Activity -> Empty View Activity*. Dalej wprowadź następujące ustawienia:



Kliknij *Finish*. W tym momencie do naszego projektu została dodana nowa klasa aktywności oraz dedykowany jej plik ustawień layoutu:



Zmodyfikuj wygląd *MainActivity* dodając nowy przycisk i layout:



Dodaj obsługę zdarzenia kliknięcia dla nowego przycisku:

```
findViewById<Button>(R.id.button_NewAct).setOnClickListener { it: View!  
    val intentSA = Intent(applicationContext, SecondActivity::class.java)  
    startActivity(intentSA)  
}
```

Uruchom aplikację i przetestuj jej działanie.

W powyższym kodzie utworzona została intencja, która wywoła *SecondActivity* znajdującą się w kontekście naszej aplikacji (*applicationContext*).

### **ZADANIE**

Dopasuj wygląd wszystkich przycisków: wypełnienie i kolor tekstu wg rysunku, rozmiar czcionki: 18dp, czcionka: pogrubiona, padding: 5dp, marginesy: 5dp. Zmień tło aktywności głównej.

