Fragmenty

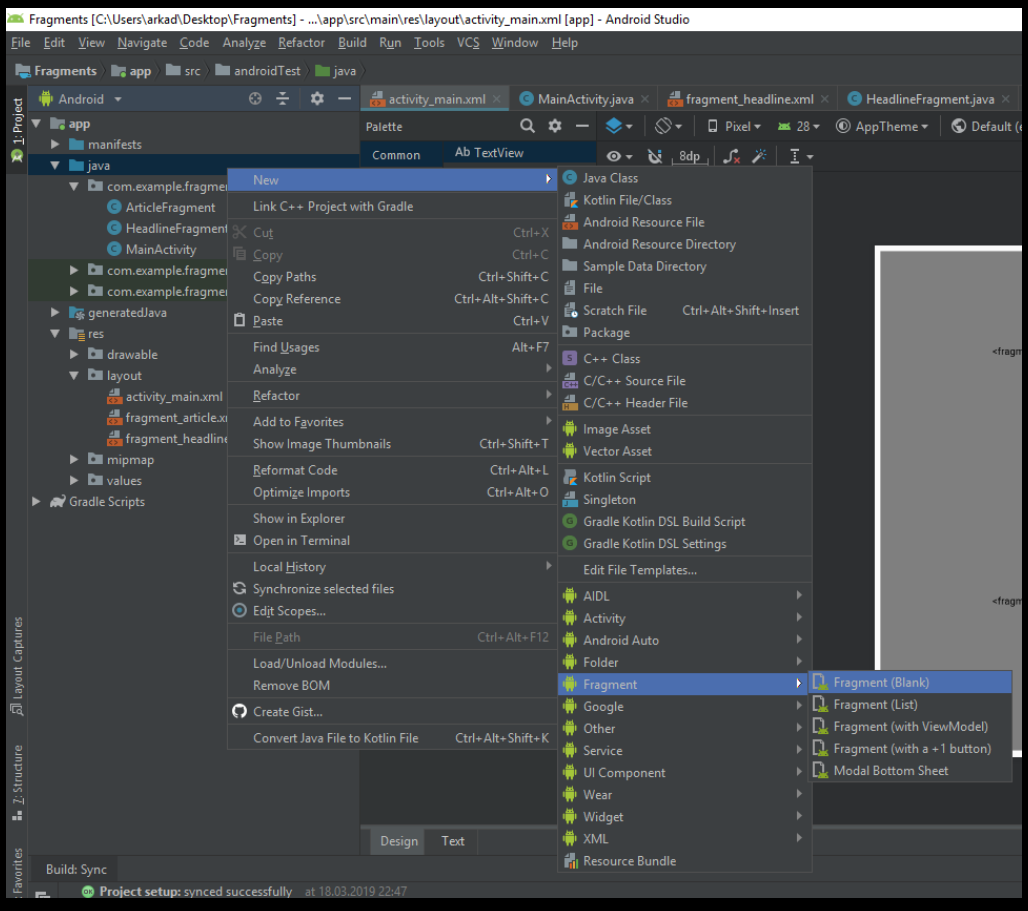
Wprowadzone w Android 3.0 (Honeycomb), obiekty Fragment są świetnym narzędziem, które przydaje się w wielu problematycznych sytuacjach, w których tylko klasy Activity były dostępne. Fragmenty pozwalają na organizowanie komponentów interfejsu projektu dla różnych urządzeń, dając możliwość wyświetlania wielu segmentów interfejsu na dużym ekranie, np. na tablecie lub wyświetlać jeden i połączyć wszystkie ze sobą na mniejszym ekranie.

Pomagają również podzielić kod na łatwe do zarządzania kawałki, bez potrzeby polegania na dużych i skomplikowanych klasach Activity. Jedną z ostatnich i prawdopodobnie najbardziej wartościową funkcją jest to, że fragmenty pozwalają na łatwe poruszanie się w aplikacji i umożliwiają proste komunikowanie się między różnymi sekcjami aplikacji.

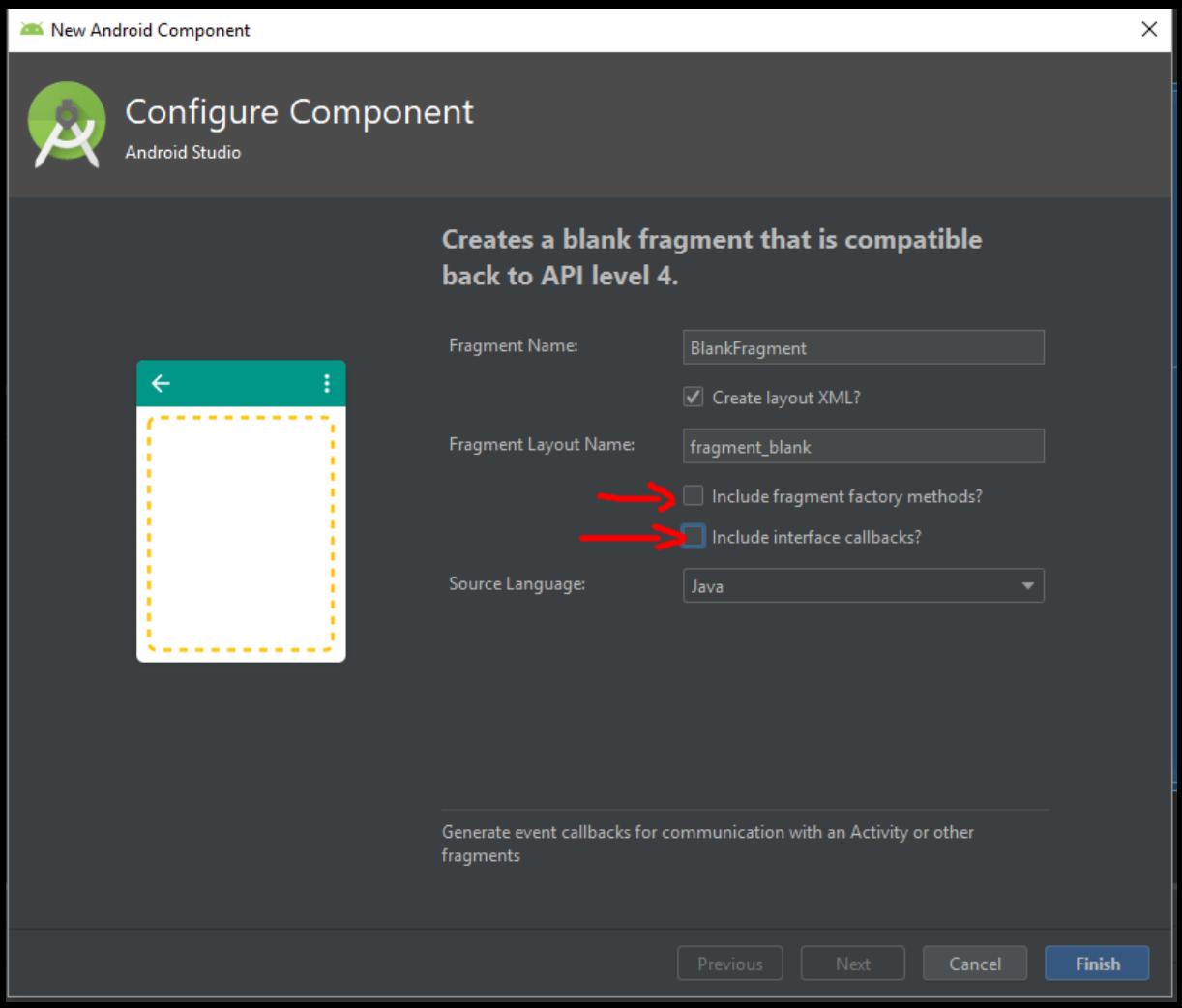
Pierwsza aplikacja wykorzystująca fragmenty

Krok 1:

Stwórz nowy projekt. Następnie klikając prawym przyciskiem myszy na folder „java” wybieramy tworzenie nowego fragmentu:



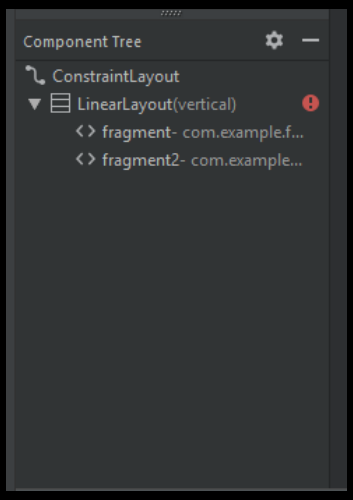
UPEWNIJ SIĘ, ŻE PONIŻSZE OPCJE SĄ ODZNACZONE!

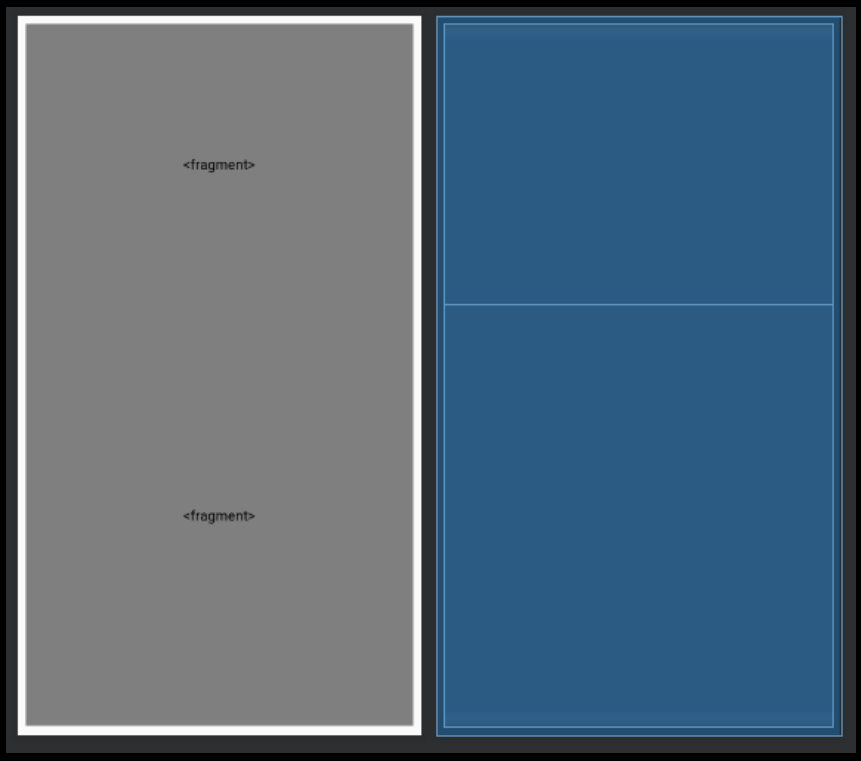


Stwórz dwa fragmenty: „HeadlineFragment” oraz „ArticleFragment” (pamiętaj, że to tylko nazwy, możesz je nazwać jak chcesz)

Krok 2:

Zaprojektuj interfejs MainActivity. W moim przypadku wygląda to tak, że na LinearLayout (vertical) nałożone są dwa elementy fragment (należy użyć opcji szukaj).





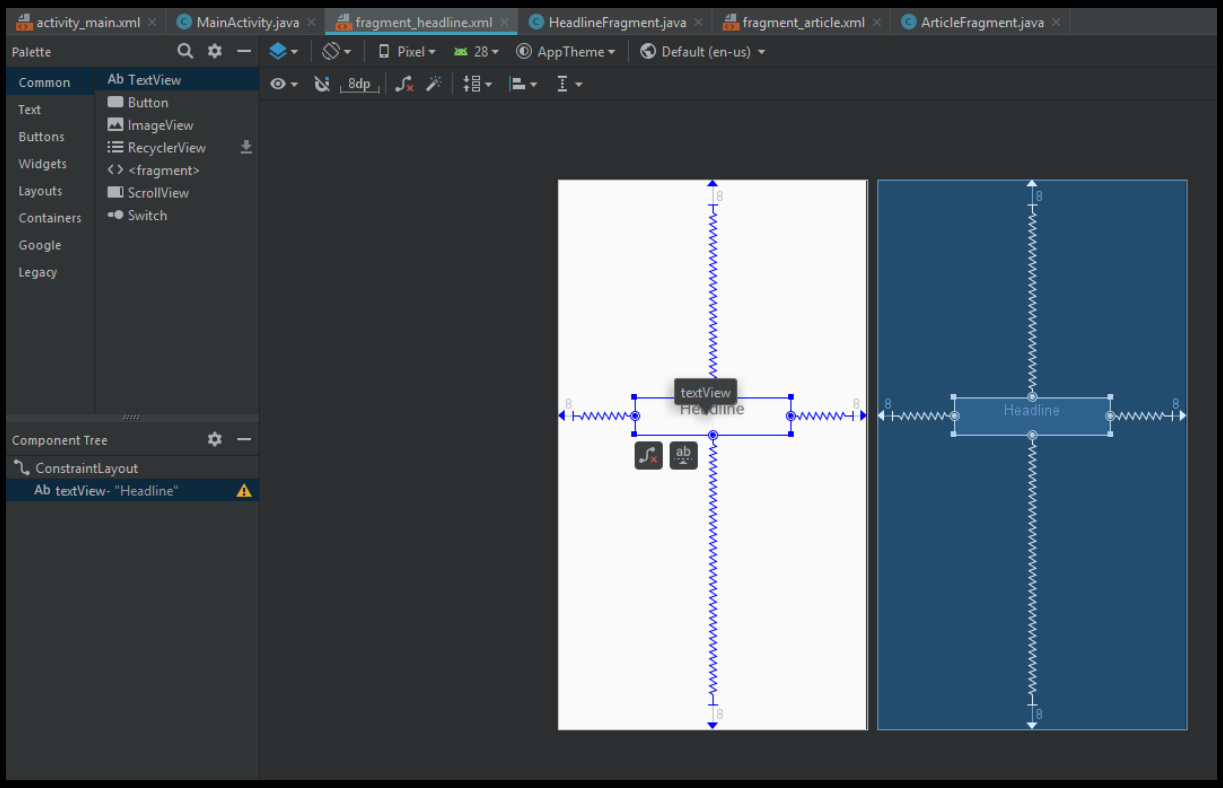
Plik activity_main.xml nie jest zbyt skomplikowany i wygląda tak:

activity_main.xml | MainActivity.java | fragment_headline.xml | HeadlineFragmer

```xml
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    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"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="395dp"
        android:layout_height="715dp"
        android:orientation="vertical"
        tools:layout_editor_absoluteX="8dp"
        tools:layout_editor_absoluteY="8dp">

        <fragment
            android:id="@+id/fragment"
            android:name="com.example.fragments.HeadlineFragment"
            android:layout_width="match_parent"
            android:layout_height="285dp" />

        <fragment
            android:id="@+id/fragment2"
            android:name="com.example.fragments.ArticleFragment"
            android:layout_width="match_parent"
            android:layout_height="429dp" />
    </LinearLayout>
</android.support.constraint.ConstraintLayout>
```

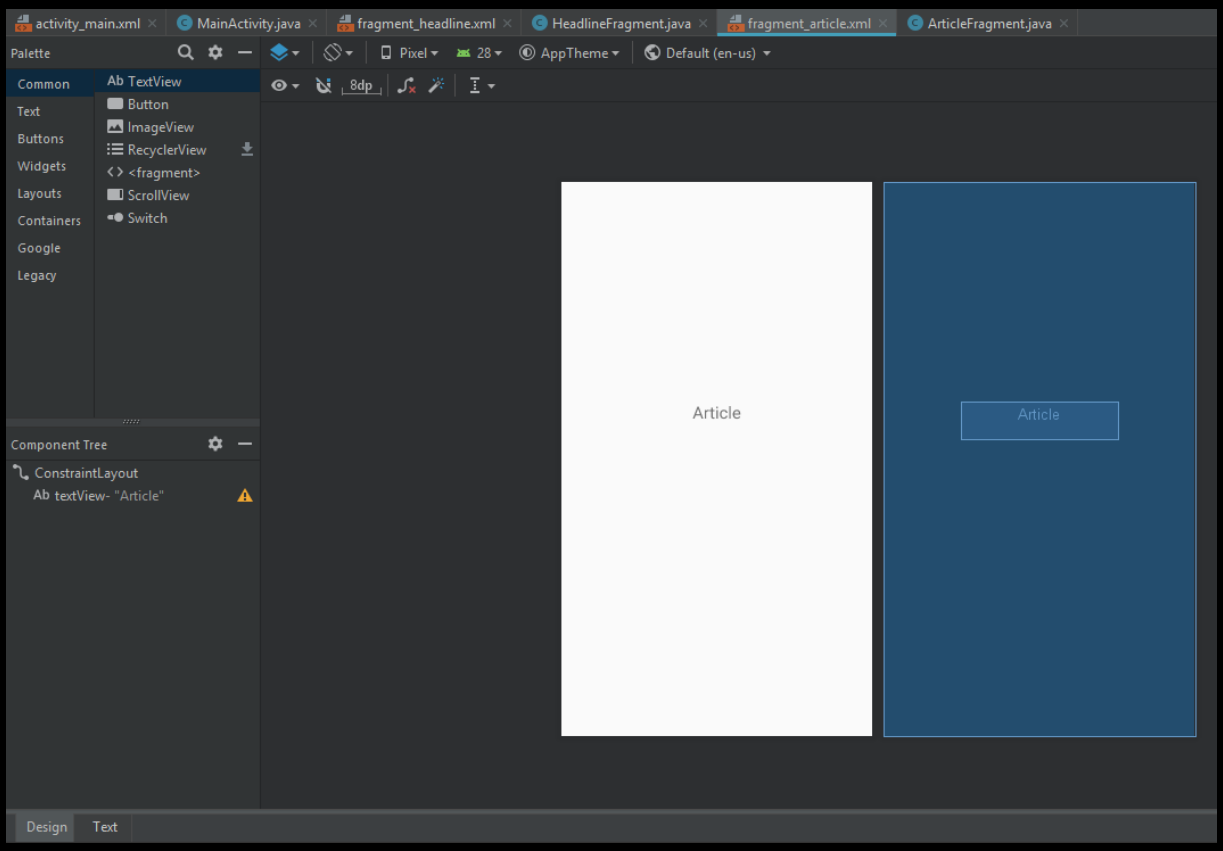
Krok 3

Pora zaprojektować nasze fragmenty. Wyglądają podobnie, HeadlineFragment wygląda tak:




```xml
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    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"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".HeadlineFragment">

    <TextView
        android:id="@+id/textView"
        android:layout_width="208dp"
        android:layout_height="50dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        android:layout_marginBottom="8dp"
        android:text="Headline"
        android:textSize="22sp"
        android:textAlignment="center"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.422" />
</android.support.constraint.ConstraintLayout>
```

A tak wygląda ArticleFragment:




```xml
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/an
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".HeadlineFragment">

    <TextView
        android:id="@+id/textView"
        android:layout_width="208dp"
        android:layout_height="50dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        android:layout_marginBottom="8dp"
        android:text="Article"
        android:textSize="22sp"
        android:textAlignment="center"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.422" />
</android.support.constraint.ConstraintLayout>
```

I to tak naprawdę wszystko. Pora uruchomić naszą aplikację.