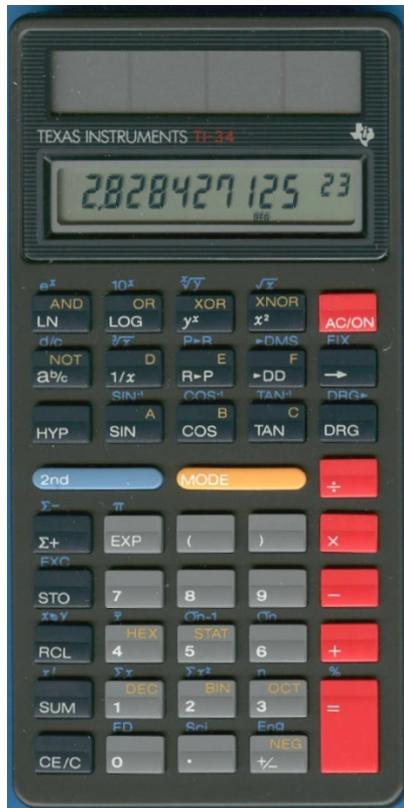


## Zmiana położenia ekranu – zmiana zawartości ekranu

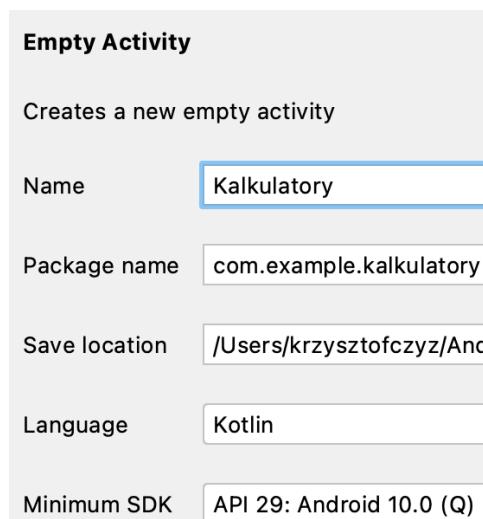
Zmiana zawartości wyświetlanej w zależności od położenia ekranu. W celu zmiany zawartości musimy stworzyć w projekcie dodatkowy folder, w którym umieszczać będziemy nasz przerobiony Layout. Przerobiony Layout oprócz zawartości Layoutu głównego będzie zawierać opis.

Utworzenie Layoutu głównego z ImageView:

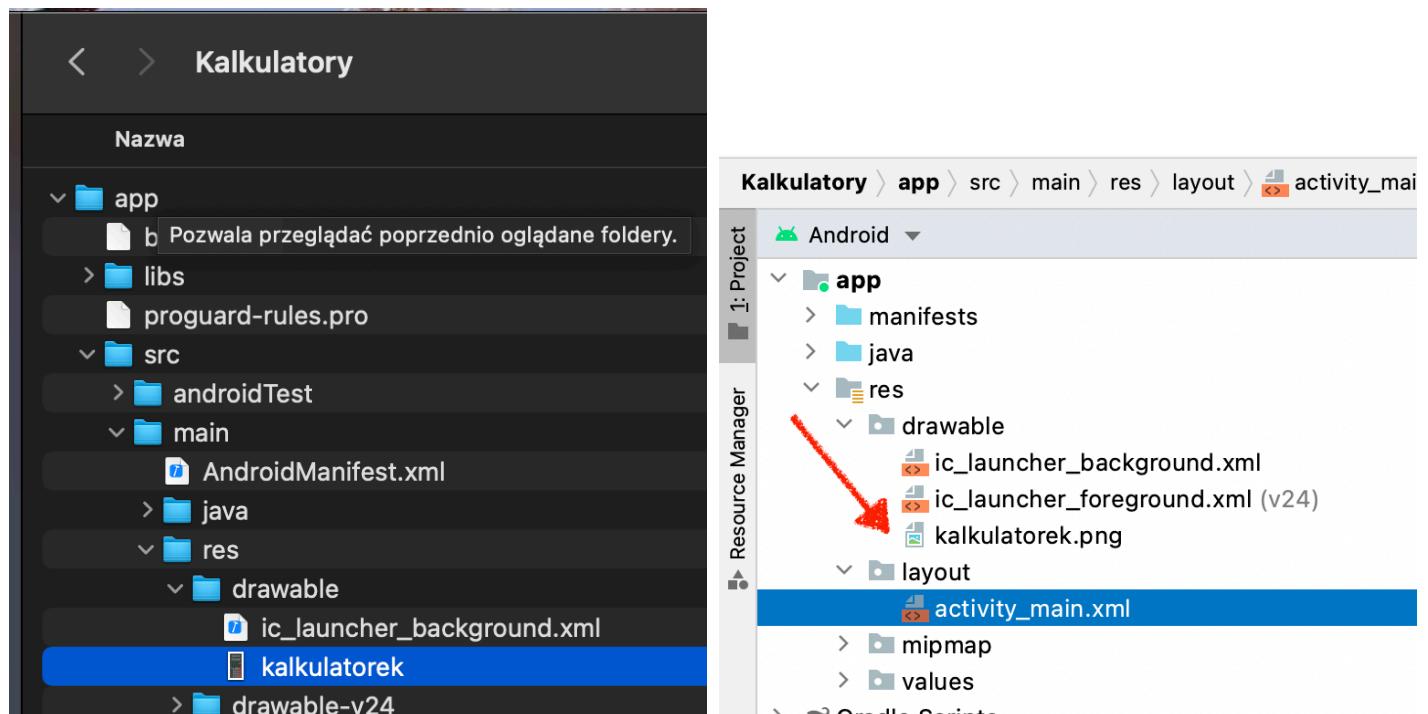
Najpierw przygotowujemy obrazek:



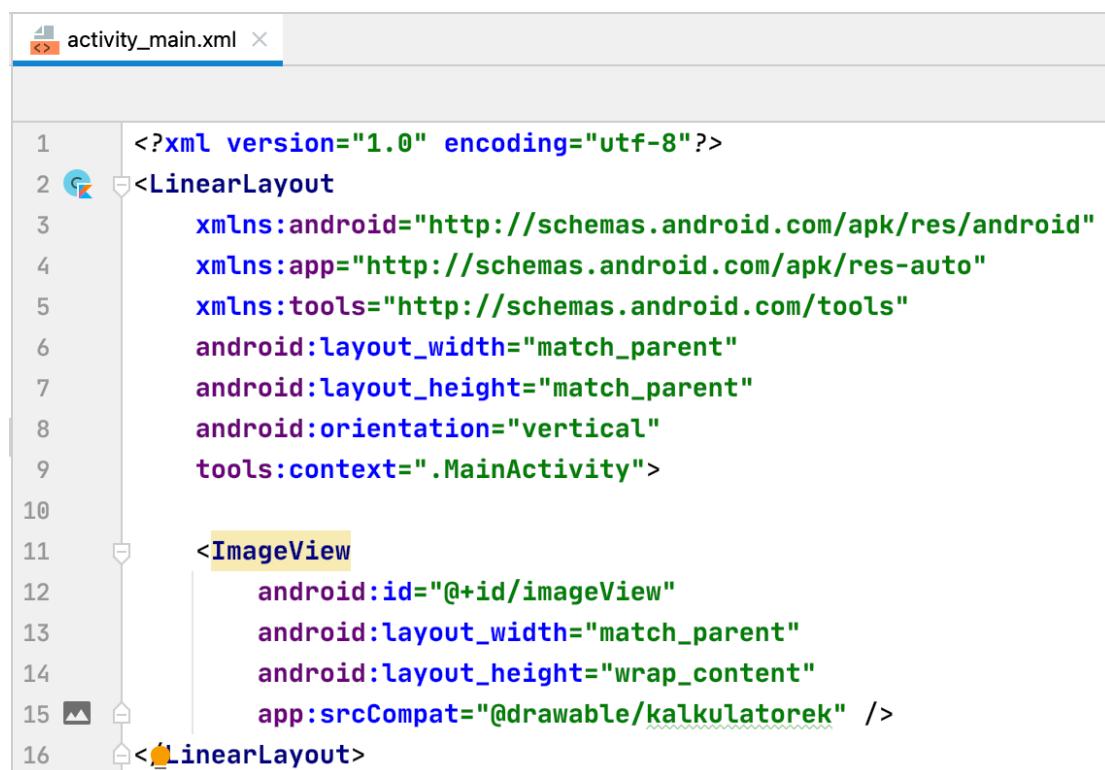
Tworzymy nowy projekt pod nazwą Kalkulatory



Teraz umieszczamy nasz obraz w odpowiednim folderze projektu i dopierołączamy projekt:



Obraz gotowy do użycia. Tworzymy wygląd naszej aktywności głównej:



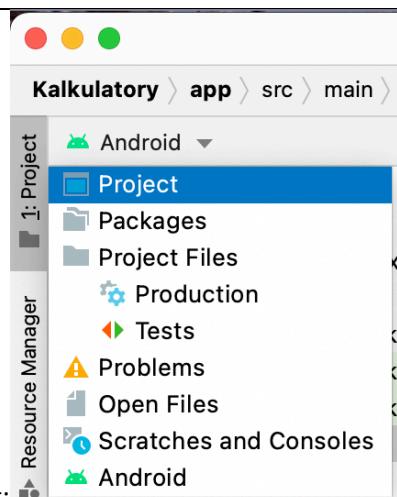
```

1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     android:orientation="vertical"
9     tools:context=".MainActivity">
10
11     <ImageView
12         android:id="@+id/imageView"
13         android:layout_width="match_parent"
14         android:layout_height="wrap_content"
15         app:srcCompat="@drawable/kalkulatorek" />
16     <LinearLayout>

```

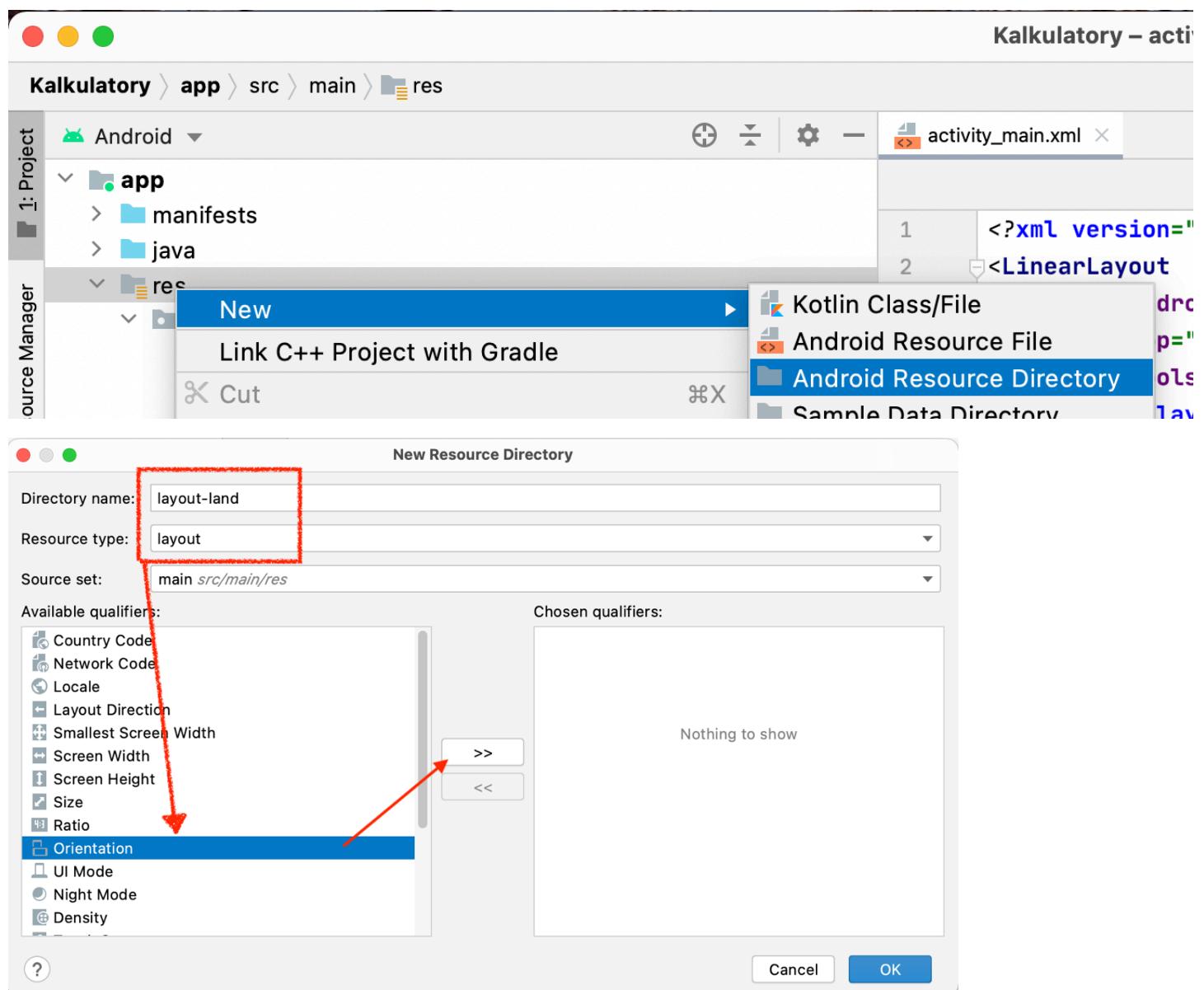
Teraz trzeba stworzyć folder: w którym zostanie umieszczona nasza główna aktywność obsługująca położenie poziome. Android sam podczas obrotu ekranu zmieni zawartość i wybierze miejsce odczytu aktywności. Nowy katalog musi mieć nazwę:

layout-land

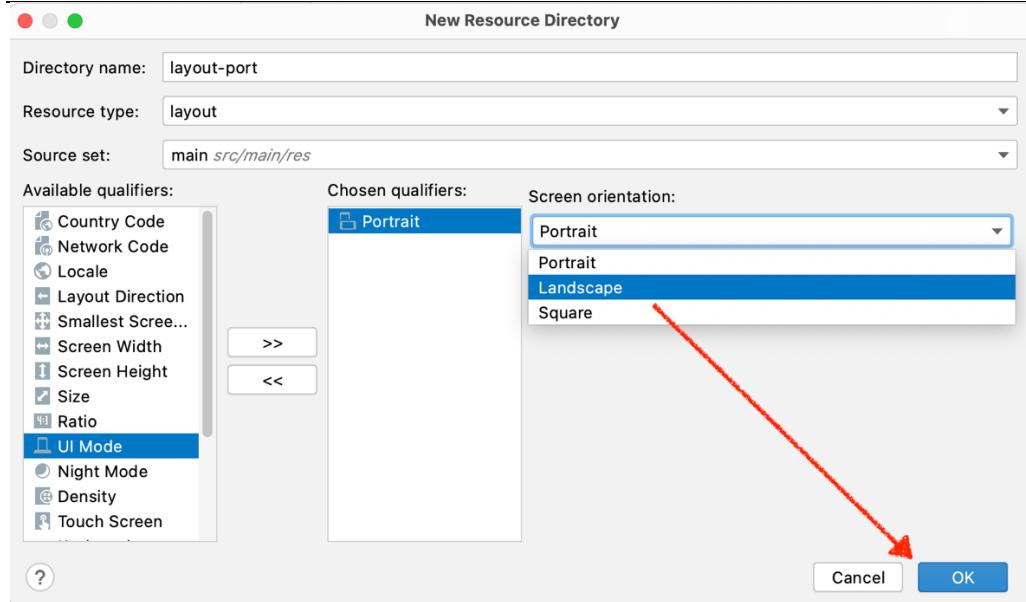


Przełączamy widok na projekt:

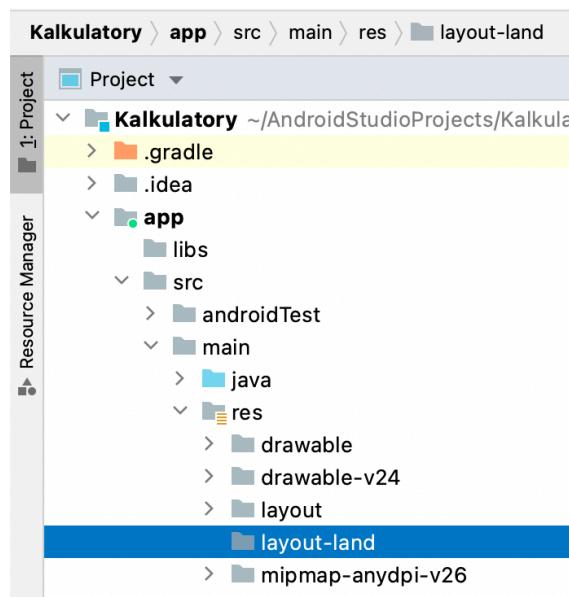
Utworzenie katalogu (klikamy prawym klawiszem w res) :



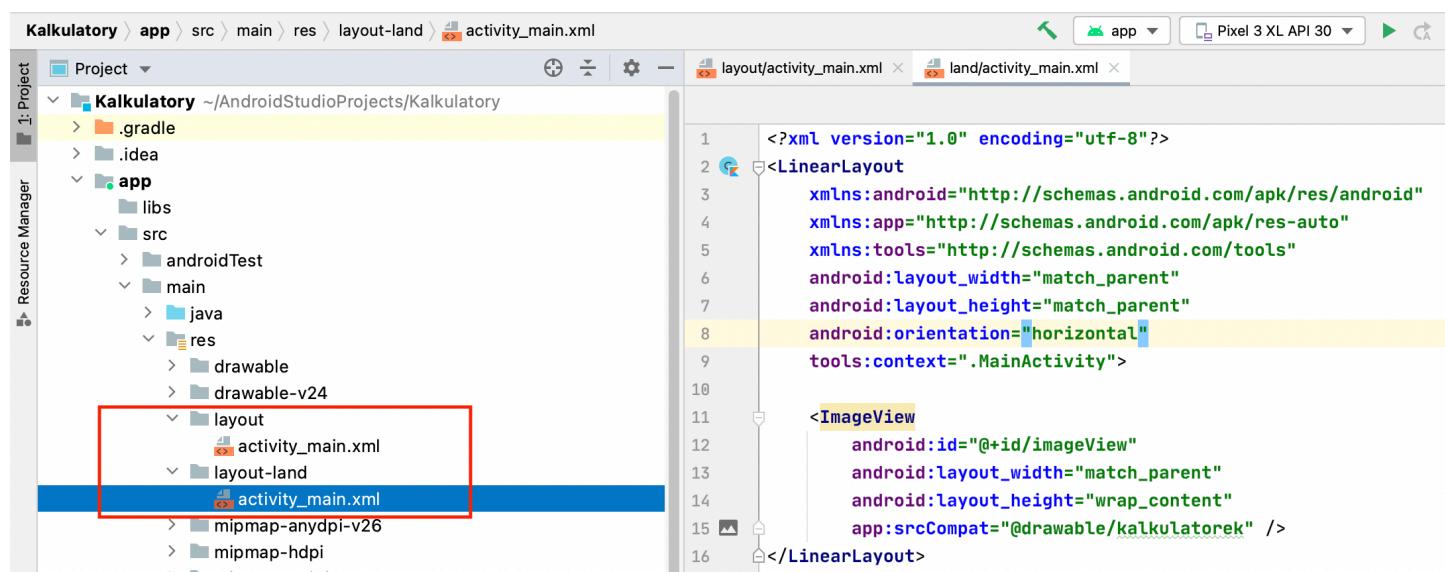
## Kurs Android Studio



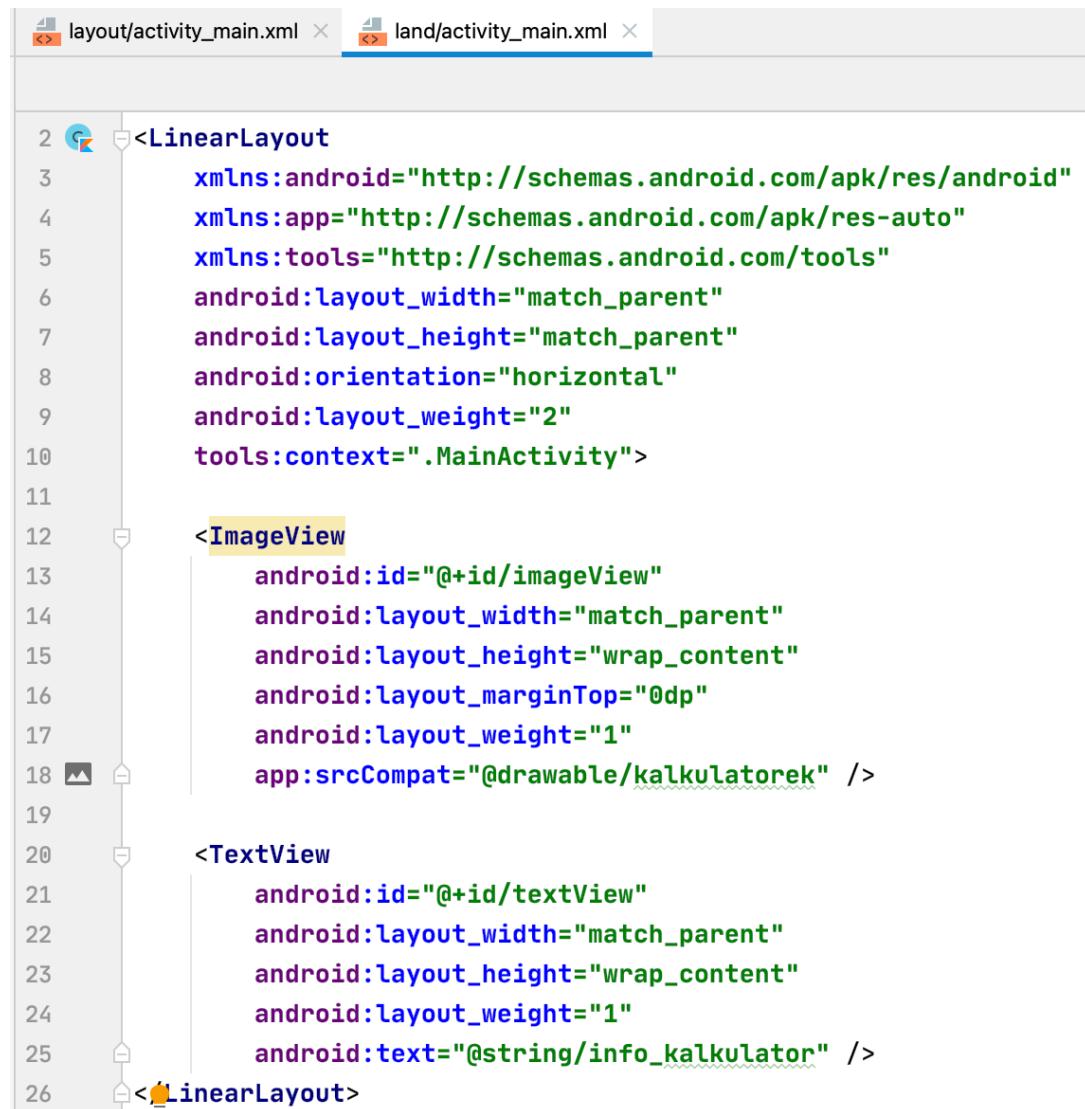
W nowym folderze:



W tym katalogu tworzymy aktywność o nazwie takiej samej jak nasza główna aktywność **main\_activity.xml** i wklejamy do niej zawartość głównej aktywności:



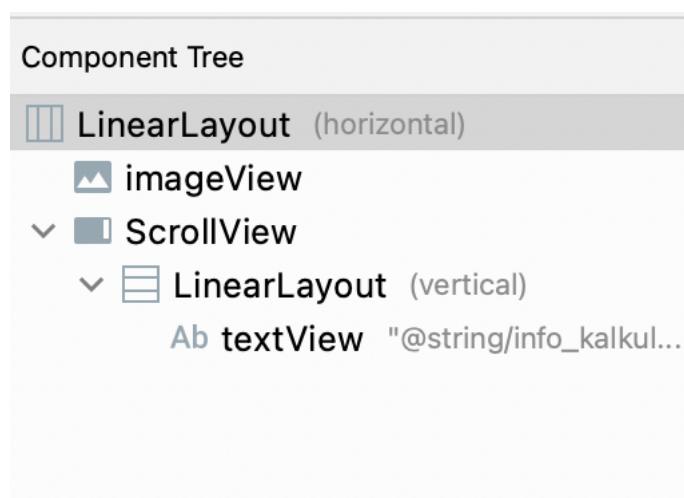
Zmieniamy orientację LinearLayout i dodajemy ScrollView w którym zamieszczamy TextView:

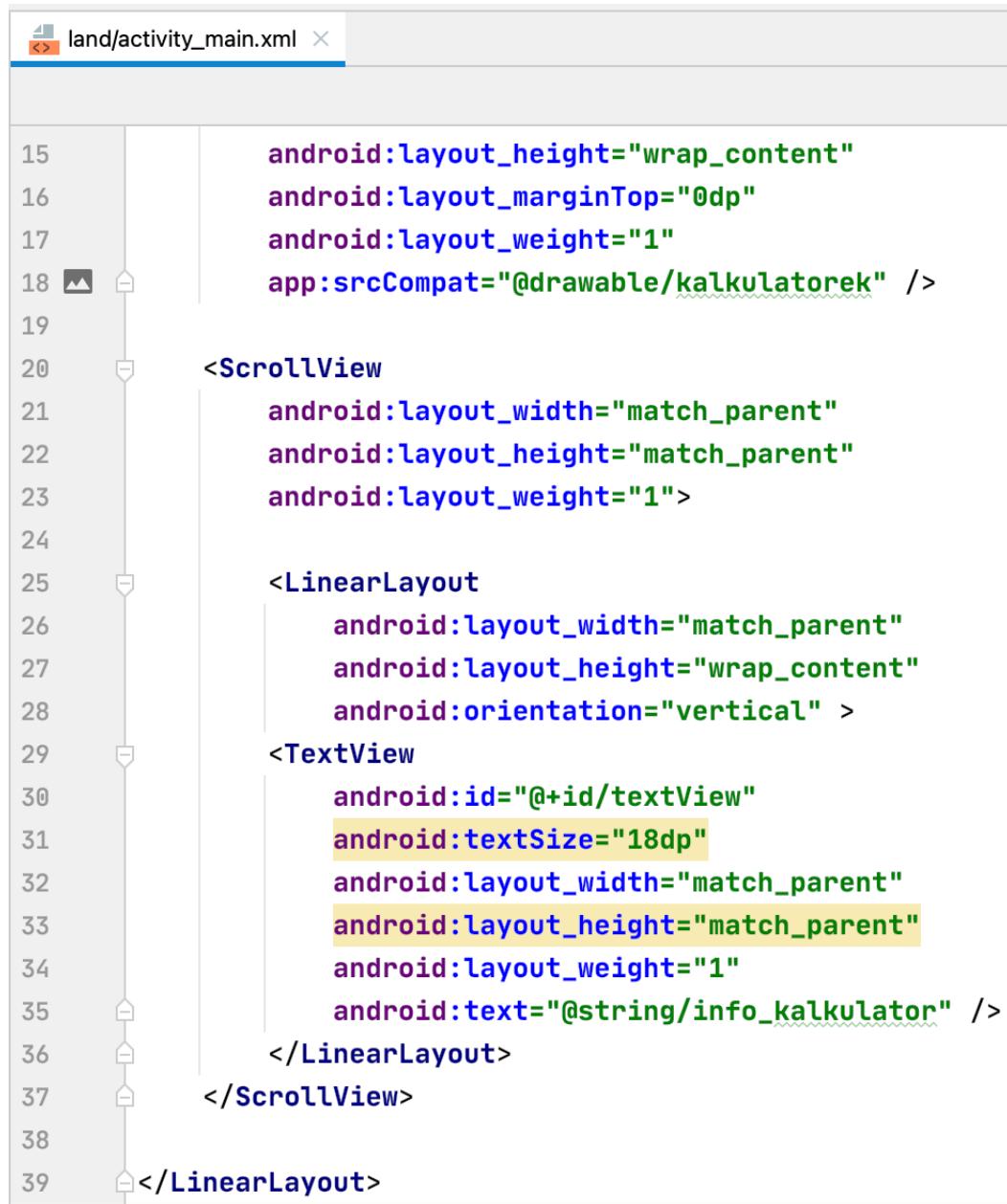


```
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     android:orientation="horizontal"
9     android:layout_weight="2"
10    tools:context=".MainActivity">
11
12    <ImageView
13        android:id="@+id/imageView"
14        android:layout_width="match_parent"
15        android:layout_height="wrap_content"
16        android:layout_marginTop="0dp"
17        android:layout_weight="1"
18        app:srcCompat="@drawable/kalkulatorka" />
19
20    <TextView
21        android:id="@+id/textView"
22        android:layout_width="match_parent"
23        android:layout_height="wrap_content"
24        android:layout_weight="1"
25        android:text="@string/info_kalkulator" />
26</LinearLayout>
```

Można sprawdzić zachowanie obu położień ekranu. Brak możliwości przesuwania tekstu. Należy dodać komponent ScrollView a przy okazji nieco zwiększyć czcionkę.

Uzyskamy nowe drzewo komponentów:





```
15     android:layout_height="wrap_content"
16     android:layout_marginTop="0dp"
17     android:layout_weight="1"
18     app:srcCompat="@drawable/kalkulatorek" />
19
20 <ScrollView
21     android:layout_width="match_parent"
22     android:layout_height="match_parent"
23     android:layout_weight="1">
24
25     <LinearLayout
26         android:layout_width="match_parent"
27         android:layout_height="wrap_content"
28         android:orientation="vertical" >
29         <TextView
30             android:id="@+id/textView"
31             android:textSize="18dp"
32             android:layout_width="match_parent"
33             android:layout_height="match_parent"
34             android:layout_weight="1"
35             android:text="@string/info_kalkulator" />
36         </LinearLayout>
37     </ScrollView>
38
39 </LinearLayout>
```