

Panel logowania cz.3 – obsługa pozostałych obiektów

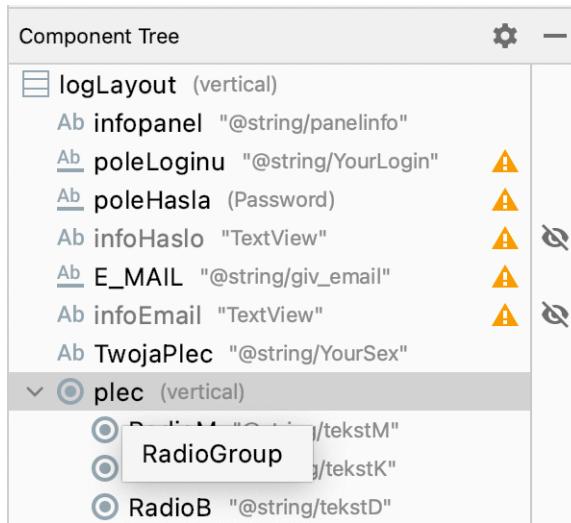
Skorzystamy z panelu logowania z poprzedniego projektu.

Na początku obsłużymy naszą grupę przycisków RadioButton. Aplikacja musi wiedzieć, który przycisk jest kliknięty i przyjąć jego wartość.

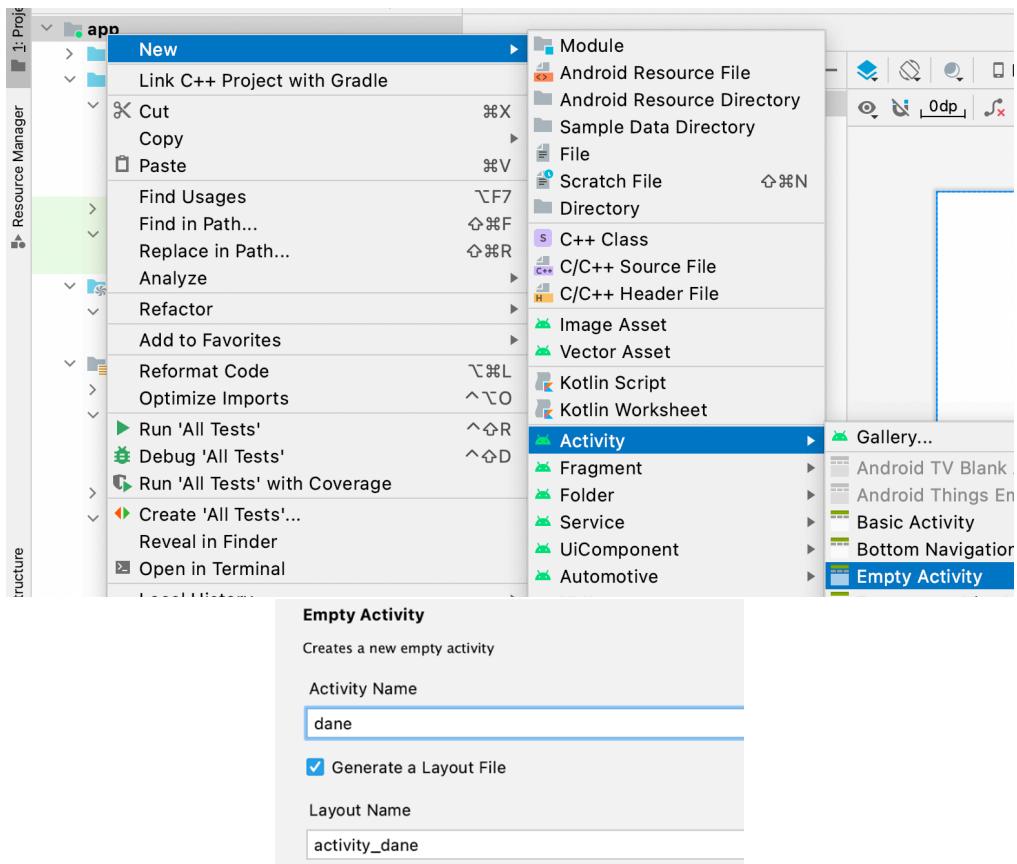
Utworzmy funkcję obsługującą RadioGroup, którą później przeniesiemy do pliku zachowanieHasla.kt .

W aktywności LogowanieActivity.kt :

Zmieniamy nazwę RadioGroup na plec

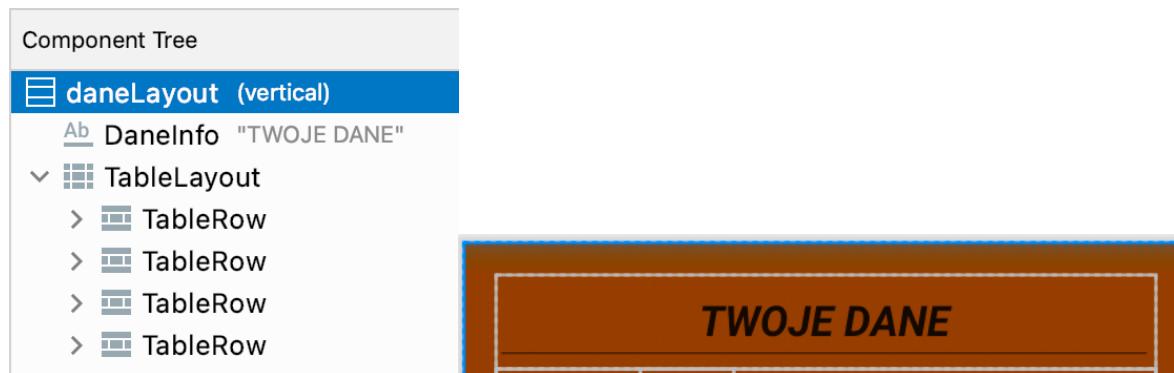


Utworzmy również aktywność do której przekażemy nasze dane z panelu logowania. Dodajemy nową aktywność o nazwie activity_dane.xml



W nowej aktywności zmieniamy tło główne oraz wstawiamy Layout verticalny.

W pierwszej linii layoutu wstawiamy napis główny Layoutu o treści „TWOJE DANE”. W drugiej linii wstawiamy kolejny layout, tym razem tabelaryczny. Pozwoli on na układanie elementów obok siebie.



Do poszczególnych wierszy wstawiamy po dwa TextView.

```
activity_dane.xml
<TableLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent" >
        <TextView...>
        <TextView...>
    </TableRow>
    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent" >
        <TextView...>
        <TextView...>
    </TableRow>
    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent" >
        <TextView...>
        <TextView...>
    </TableRow>
    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent" >
        <TextView...>
        <TextView...>
    </TableRow>
</TableLayout>
</LinearLayout>
```

The screenshot shows the XML code for the 'activity_dane.xml' file. The code defines a 'TableLayout' with five 'TableRow' children. Each 'TableRow' contains two 'TextView' elements. The code is color-coded for syntax highlighting, and some parts are highlighted with yellow boxes, likely indicating selected or edited text.

Będą one zawierały w komórkach tabeli po lewej stronie identyfikację pobranych danych, a po prawej wpisane przez nas dane.

Twoje imię	
Twoje hasło	
Twój e-mail	
Twoja płeć	

Oczywiście hasła nie wolno w aplikacjach pokazywać i w pole obok napisu „Twoje hasło” wypełni się gwiazdkami.

```
74      <TextView  
75          android:id="@+id/YourSex"  
76          android:layout_width="wrap_content"  
77          android:layout_height="wrap_content"  
78          android:text="Twoja płeć: " />
```

```
28      <TextView  
29          android:id="@+id/YourName"  
30          android:layout_width="wrap_content"  
31          android:layout_height="wrap_content"  
32          android:text="Twoje imię: " />
```

```
42      <TextView  
43          android:id="@+id/YourPass"  
44          android:layout_width="wrap_content"  
45          android:layout_height="wrap_content"  
46          android:text="Twoje hasło: "  
47          android:visibility="visible" />
```

```
58      <TextView  
59          android:id="@+id/YourMail"  
60          android:layout_width="wrap_content"  
61          android:layout_height="wrap_content"  
62          android:text="Twój e-mail: " />
```

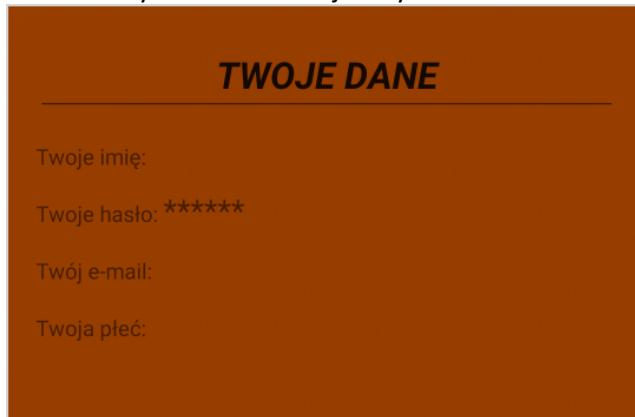
Pozostałe pola zawierają zmienione nazwy oraz powiększoną czcionkę i padding górnny:

```
33      <TextView  
34          android:id="@+id/name_view"  
  
44          android:id="@+id/YourPass"  
  
66          android:id="@+id/email_view"  
  
82          android:id="@+id/plec_view"  
83          android:layout_width="match_parent"  
84          android:layout_height="wrap_content"  
85          android:paddingTop="10dp"  
86          android:textSize="20sp" />
```

Pole z hasłem dodatkowo zawiera wypełnioną komórkę z gwiazdkami

```
53          android:text="*****"
```

Powinniśmy widzieć interfejs aktywności:



W pliku **dane.kt** tworzymy zmienne, które będą przechowywać wartości z **activity_logowanie.xml** oraz linie kodu, które będą przechwytywać dane i uzupełniać obiekty **TextView** w pustych polach tabeli.

Jest to możliwe dzięki zastosowaniu metody intencji `hasExtra` oraz `getCharSequenceExtra`.

Uzupełniony kod w pliku powinien mieć postać:

```

1 package com.example.logowanie2
2
3 import androidx.appcompat.app.AppCompatActivity
4 import android.os.Bundle
5 import android.widget.EditText
6 import android.widget.TextView
7
8 class dane : AppCompatActivity() {
9     override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11         setContentView(R.layout.activity_dane)
12         //przekazane dane
13         var nazwaView = findViewById<TextView>(R.id.name_view)
14         if(intent.hasExtra( name: "poleLoginu")) nazwaView.setText(intent.getCharSequenceExtra( name: "poleLoginu"))
15         var mailView = findViewById<TextView>(R.id.email_view)
16         if(intent.hasExtra( name: "E_MAIL")) mailView.setText(intent.getCharSequenceExtra( name: "E_MAIL"))
17         var plecView = findViewById<TextView>(R.id.plec_view)
18         if(intent.hasExtra( name: "plec")) plecView.setText(intent.getCharSequenceExtra( name: "plec"))
19     }
20 }
```

W pliku **logowanie_activity.kt** musimy dorobić zachowanie przycisków **RadioButton** w taki sposób, aby uruchomienie przycisku powodowało utworzenie nowej intencji i przekierowało do niej dane. (Jest to najważniejszy kawałek kodu związany z przekazaniem danych do nowej aktywności)

```

24 //oprogramowanie RadioGroupu
25 var plec = findViewById<RadioGroup>(R.id.plec)
26 plec.setOnCheckedChangeListener{ group,checkedId ->
27     run{ this: LogowanieActivity
28         //tworzymy nową intencję
29         val intencjaAktywowana: Intent = Intent(applicationContext,dane::class.java)
30         //tworzymy zmienną odczytującą aktywny RadioButton
31         var AktywnyRadioB : RadioButton = findViewById(checkedId) //identyfikuje klikniętego przycisku
32         //uruchamiamy utworzoną intencję aktywującą i przekazujemy jej wartości mail i płeć
33         intencjaAktywowana.putExtra( name: "E_MAIL", email.text)
34         intencjaAktywowana.putExtra( name: "poleLoginu", login.text)
35         intencjaAktywowana.putExtra( name: "plec", AktywnyRadioB.text)
36         startActivity(intencjaAktywowana)
37     }
}
```

Warto sprawdzić, czy mamy zaimplementowane wszystkie importy:

```
LogowanieActivity.kt
1
2
3 import android.content.Intent
4 import android.os.Bundle
5 import android.text.Editable
6 import android.text.TextWatcher
7 import android.widget.EditText
8 import android.widget.RadioButton
9 import android.widget.RadioGroup
10 import android.widget.TextView
11 import androidx.appcompat.app.AppCompatActivity
12
```

Po wykonaniu wszystkich powyższych czynności powinniśmy zobaczyć nową aktywność z wpisanymi przez nas danymi. (oczywiście, jak ktoś chce to również może przekazać sobie swoje hasło i je pokazać – warto to zrobić na etapie programowania).

