

## **Zadania1. Nowa aktywność ( Instancja )**

Napisz aplikację na system Android, która składa się z dwóch aktywności: MainActivity oraz opis. Aplikacja ma za zadanie wyświetlać informacje o malarzu Vincentcie van Goghu.

Szczegółowe wymagania:

### 1. MainActivity:

- Utwórz klasę MainActivity oraz zdefiniuj layout z jednym przyciskiem.
- Znajdź przycisk za pomocą metody findViewById i dodaj nasłuchiwanca zdarzeń (OnClickListener).
- Po kliknięciu przycisku aplikacja ma uruchamiać aktywność opis przy użyciu intencji (Intent).

### 2. opis:

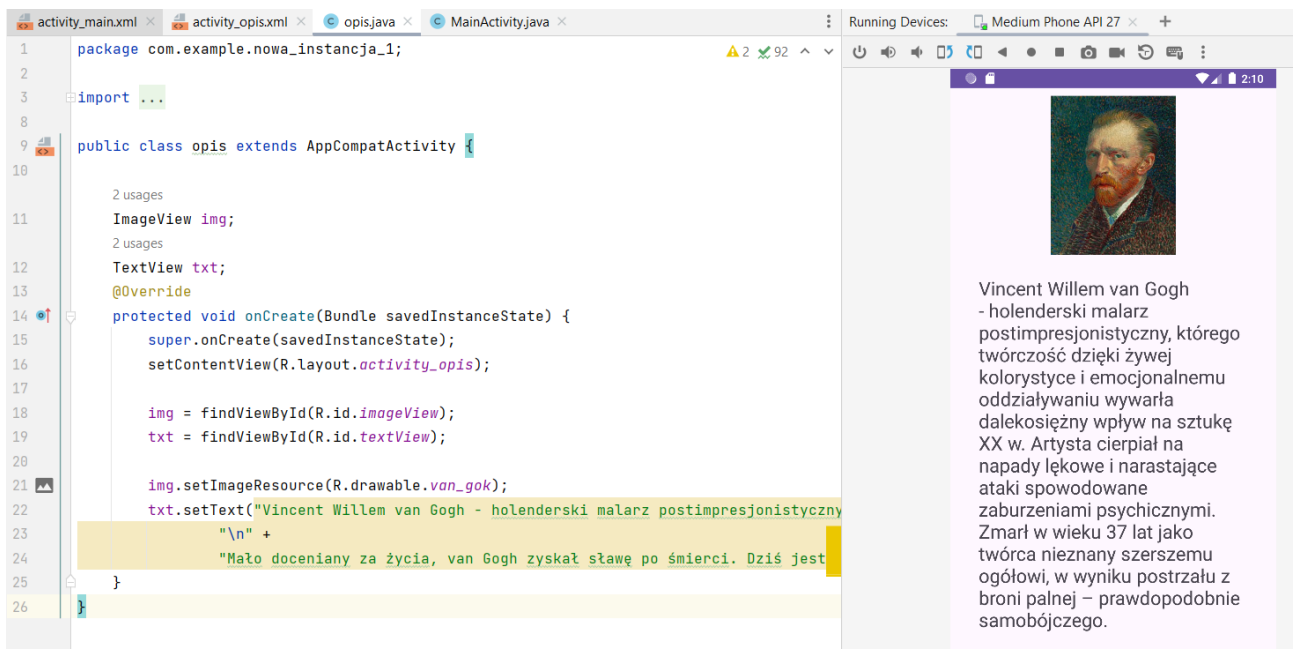
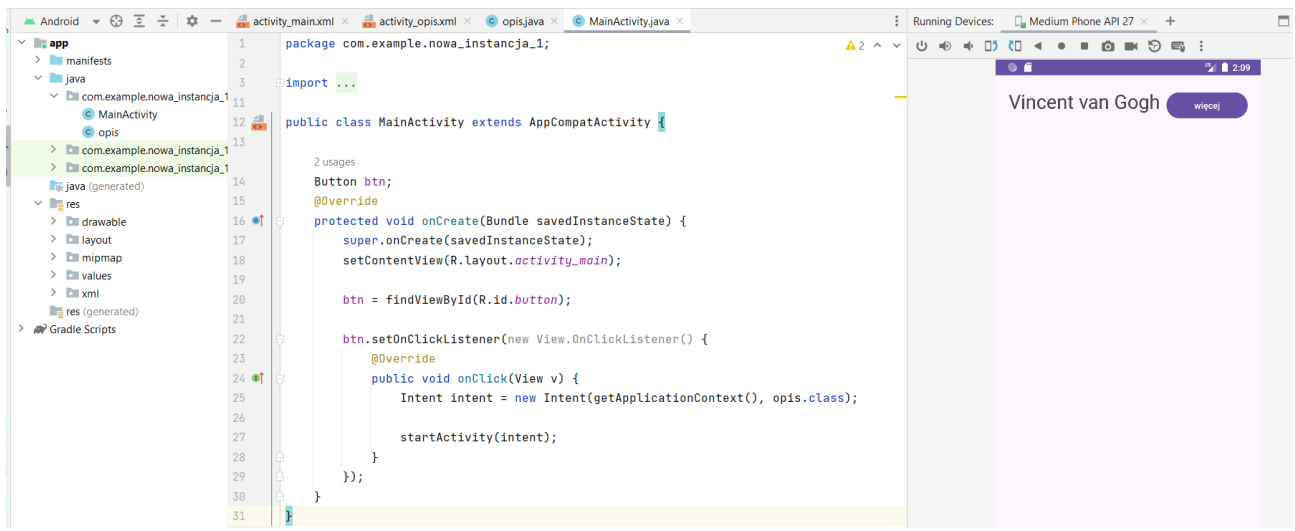
- Utwórz klasę opis, której layout zawiera obraz oraz tekst.
- W klasie opis zdefiniuj dwa elementy interfejsu:
  - ImageView do wyświetlania obrazu.
  - TextView do wyświetlania opisu malarza Vincenta van Gogha.
- W kodzie wstaw obraz malarza za pomocą metody setImageResource() i ustaw opis jego twórczości oraz życia w TextView za pomocą setText().

### 3. Zasoby aplikacji:

- Dodaj do projektu obrazek van Gogha (np. plik van\_gok.png) i umieść go w katalogu res/drawable.
- Upewnij się, że obrazek jest poprawnie wyświetlany w aktywności opis.

### 4. Dodatkowe wymagania:

- W aktywności MainActivity przekierowanie do aktywności opis powinno odbywać się za pomocą intencji po kliknięciu przycisku.
- Zadbaj o to, aby aktywność opis była zadeklarowana w pliku AndroidManifest.xml.



## Jak dodać nową aktywność do projektu:

- prawy przycisk myszy na folderze java -> New -> Activity -> Empty Activity i nazwij ją AnimalDetailActivity.

## Zadanie 2 Galeria sztuki.

Napisz aplikację na system Android, która umożliwi użytkownikowi wybór jednego z dwóch słynnych dzieł sztuki oraz wyświetli ich opisy na osobnym ekranie. Aplikacja będzie składać się z dwóch aktywności: **MainActivity** i **galeria**.

### Szczegółowe wymagania:

#### 1. **MainActivity**:

- Utwórz klasę **MainActivity**, której layout zawiera dwa przyciski: jeden dla obrazu "**Krzyk**" Edvarda Muncha, a drugi dla obrazu "**Gwiaździsta noc**" Vincenta van Gogha.
- Każdy z przycisków po kliknięciu ma uruchomić nową aktywność **galeria** i przekazać do niej odpowiedni opis dzieła sztuki za pomocą intencji (Intent).
  - Przy pierwszym przycisku przekazany opis to tekst dotyczący obrazu "Krzyk".
  - Przy drugim przycisku przekazany opis to tekst dotyczący obrazu "Gwiaździsta noc".

#### 2. **galeria**:

- Utwórz klasę **galeria**, której layout zawiera element **TextView**.
- W aktywności **galeria** odbierz opis przekazany z **MainActivity** za pomocą metody `getIntent().getStringExtra()` i wyświetl go w **TextView**.

#### 3. **Zasoby aplikacji**:

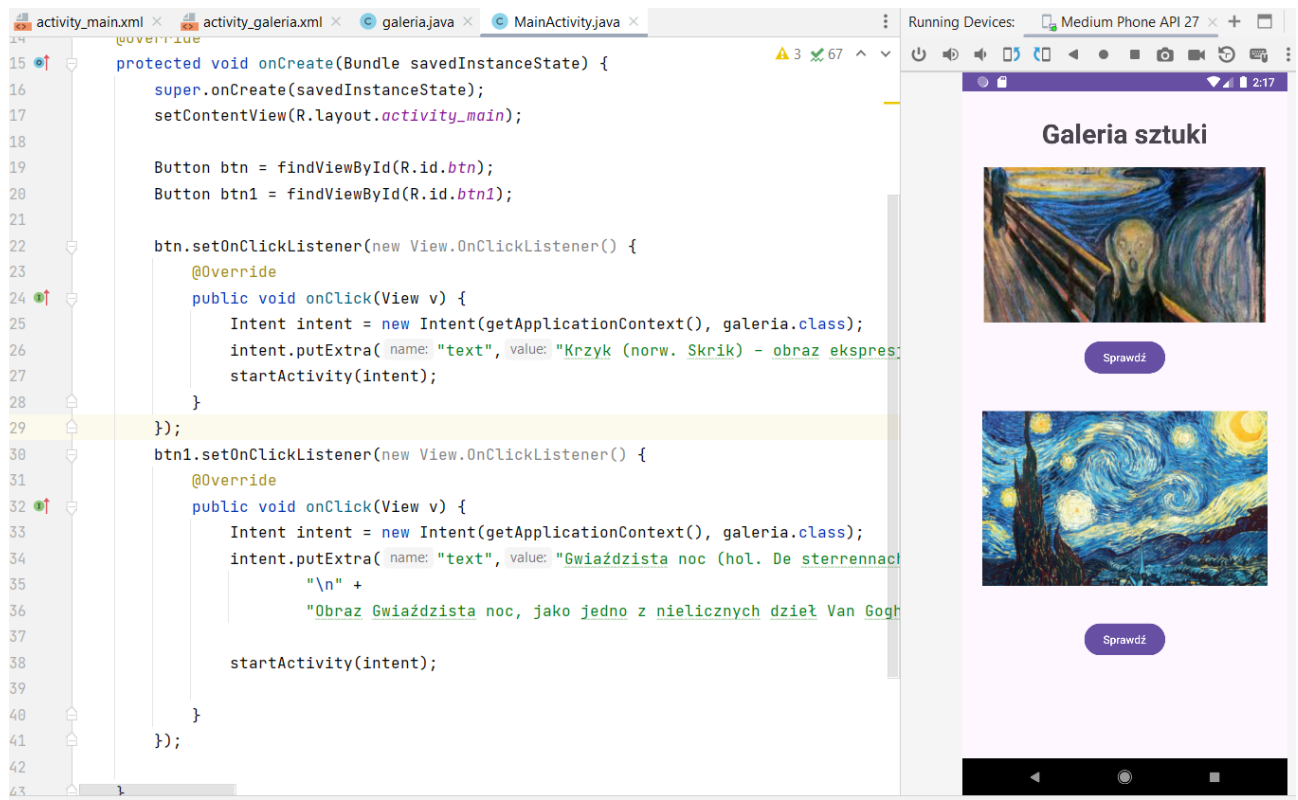
- Stwórz odpowiednie layouty XML dla obu aktywności: **activity\_main.xml** (z dwoma przyciskami) oraz **activity\_galeria.xml** (z jednym elementem **TextView**).

#### 4. **Obsługa intencji**:

- Każdy przycisk w **MainActivity** powinien wywołać metodę `startActivity()` z odpowiednią intencją zawierającą opis wybranego dzieła sztuki.

### Dodatkowe wymagania (opcjonalne):

- W aktywności **galeria** możesz dodać więcej elementów graficznych, np. wyświetlanie obrazków dzieł sztuki, używając **ImageView**.
- Upewnij się, że obie aktywności są poprawnie zadeklarowane w pliku **AndroidManifest.xml**.



`Intent.putExtra(String key, Serializable value);`

Parametry:

- **key:** Klucz (ciąg znaków), który identyfikuje wartość przekazywaną do intencji.
- **value:** Wartość, która ma być przekazana. Może to być np. String, int, boolean, Parcelable lub inne typy danych wspierane przez intencje.



### Zadanie 3. Galeria zwierząt nietypowych.

Stwórz aplikację na Androida w języku Java, która wyświetla galerię obrazków zwierząt. Użytkownik może klikać na obrazki, aby zobaczyć powiększoną wersję obrazka i nazwę zwierzęcia w nowej aktywności.

Wymagania:

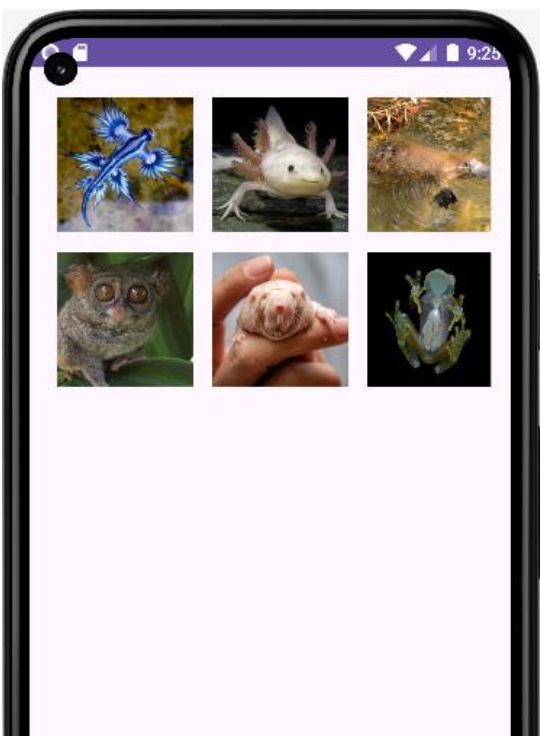
1. MainActivity:

- Layout: Użyj TableLayout, który wyświetli kilka obrazków zwierząt w siatce.
- Funkcjonalność:
  - Każdy obrazek powinien być klikalny.
  - Kliknięcie na obrazek otwiera nową aktywność (AnimalDetailActivity), która wyświetla powiększony obrazek i nazwę zwierzęcia.

2. AnimalDetailActivity:

- Layout:
  - Zawiera ImageView do wyświetlania powiększonego obrazka.
  - Zawiera TextView do wyświetlania nazwy zwierzęcia.
- Funkcjonalność:
  - Odbiera dane (obrazek i nazwę zwierzęcia) przekazane z MainActivity.

Rys 1.



Rys 2.



## Fragment pliku activity\_main.xml

```
<android.widget.TableLayout xmlns:android="http://schemas.android.co
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="2"
    android:padding="16dp">

    <TableRow>
        <ImageView
            android:id="@+id/imageView1"
            android:src="@drawable/niebieski_smok"
            android:contentDescription="niebieski smok"
            style="@style/obraz" />

        <ImageView
            android:id="@+id/imageView2"
            style="@style/obraz"
            android:contentDescription="karlik szponiasty"
            android:src="@drawable/karlik_szponiasty" />

        <ImageView
            android:id="@+id/imageView3"
            style="@style/obraz"
            android:contentDescription="dziobak"
            android:src="@drawable/dziobak" />
    </TableRow>
```

## Plik activity\_animal\_detail.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">

    <ImageView
        android:id="@+id/animalImageView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:src="@drawable/niebieski_smok" />

    <TextView
        android:id="@+id/animalNameTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/animalImageView"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:text="niebieski smok"
        android:textSize="18sp" />

</RelativeLayout>
```

## Plik style.xml

```
<resources>
  <style name="obraz">
    <item name="android:layout_width">110dp</item>
    <item name="android:layout_height">110dp</item>
    <item name="android:layout_margin">8dp</item>
    <item name="android:scaleType">fitXY</item>
  </style>
</resources>
```

## Plik MainActivity.java

```
int[] imageViewIds = {R.id.imageView1, R.id.imageView2, R.id.imageView3, R.id.imageView4, R.id.imageView5, R.id.imageView6};
1 usage
int[] imageResIds = {R.drawable.niebieski_smok, R.drawable.karlik_szponiasty, R.drawable.dziobak, R.drawable.wyrak, R.drawable
1 usage
String[] animalNames = {"niebieski smok", "karlik szponiasty", "dziobak", "wyrak", "golec", "żaba szklana"};

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    for (int i = 0; i < imageViewIds.length; i++) {
        ImageView imageView = findViewById(imageViewIds[i]);
        int imageResId = imageResIds[i];
        String animalName = animalNames[i];

        imageView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openAnimalDetail(imageResId, animalName);
            }
        });
    }
}
```

## Plik AnimalDetailActivity.java

```
ImageView animal;  
2 usages  
TextView name;  
  
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_animal_detail);  
  
    animal = findViewById(R.id.animalImageView);  
    name = findViewById(R.id.animalNameTextView);  
  
    int imageResId = getIntent().getIntExtra("imageResId", 0);  
    String animalName = getIntent().getStringExtra("animalName");  
  
    animal.setImageResource(imageResId);  
    name.setText(animalName);  
}
```