

Tworzenie ArrayAdaptera

Adapter jest mostem między interfejsem aplikacji, a źródłem danych. Pomaga nam w konwertowaniu danych na widoki. W tym zadaniu pracujemy głównie na ArrayAdapter, który przyjmuje typ **String**. Oto jego krótka definicja:

```
ArrayAdapter<String> nazwa_adaptera = new ArrayAdapter<String>
                                     (this, element_layoutu, id_elementu, dane);
```

this – kontekst, z którego tworzony będzie adapter. „this” wskazuje na obecny widok – w naszym przypadku zazwyczaj jest to MainActivity.

element_layoutu – jest to osobny element XML zdefiniowany w katalogu res/layout, który przechowuje informacje o wyglądzie elementu, który będzie wstawiany do widoku (ListView).

id_elementu – jest to ID elementu z **element_layoutu**, konkretnego TextView.

dane – tablica/lista z danymi, które mają być zamieszczone w widoku.

Adapter podpinamy do RecyclerViewa, w naszym przypadku jest to ListView. Sposób podpinania adaptera wygląda tak:

```
(ListView).setAdapter(nazwa_adaptera);
```

Nasłuchiwanie zdarzeń w ListView

Aby wykryć nasłuchiwanie zdarzeń, w tym przypadku onClick, na elementach listy należy dodać nasłuchiвач **onItemClickListener** na cały list View.

Definicja nasłuchiвачa wygląda następująco.

```
(ListView).setOnClickListener(parent, view, position, id) → {});
```

parent – rodzic/lista (w tym przypadku ListView).

view – jeden konkretny wiersz listy, który został naciśnięty.

position – indeks klikniętego elementu listy, zaczyna się od zera, pozwala na wybranie zawartości z tablicy pod odpowiednim indeksem.

id – identyfikator wiersza (na naszym przykładzie w sumie to samo co **position**).

Zadanie

Twoim zadaniem jest w pełni odtworzyć aplikację z [egzaminu zawodowego](#). Układ, widok, kolory, zastosowane elementy powinny zgadzać się z wymogami aplikacji. Ocena będzie na podstawie [zasad oceniania](#) dla tego arkusza (R.3) oraz dodatkowo punktowane będzie rozumowanie napisanego kodu.