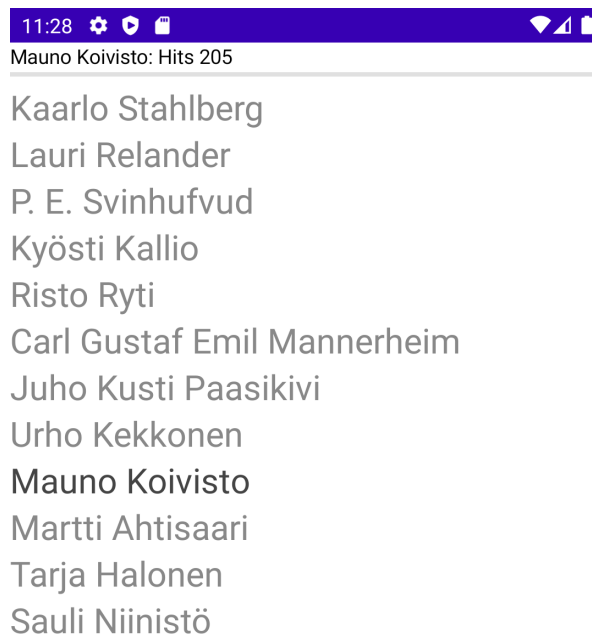


Exercise Retrofit

Wikipedia has a good web service API for making various queries about their database (https://www.mediawiki.org/wiki/API:Main_page). Use your previous lab as your base if you want. Create an application which shows the president list, and when the user clicks on one president, he will also get information about how many links there are to his web page in Wikipedia.



Hint 1: Query like

<https://en.wikipedia.org/w/api.php?action=query&format=json&list=search&srsearch=Trump> will give you a JSON answer which contains a lot of information about the query. The interesting field is named as totalhits. You can put the query to the web browser and see the result.

Hint 2: If you want to follow the communication between your application and the server, use Network Inspector. It can be activated when the application is running (either on your phone or on the emulator) by clicking the App Inspector name on the bottom menu/icon line. If you select a time scale on the Network graph, you can see the query and response in clear text.

Hint 3: Remember to insert permission for the Internet communication to the Android Manifest.xml file

```
<uses-permission android:name="android.permission.INTERNET" />
```

Hint 4: Remember to add those retrofit library references to your build gradle (Module) dependencies block

```
final retrofit_version = '2.9.0'  
implementation "com.squareup.retrofit2:retrofit:$retrofit_version"  
implementation "com.squareup.retrofit2:converter-gson:$retrofit_version"
```

Hint 5: Create first an application that only fetches the hit count for a given president name (using Retrofit and LiveData & ViewModel).

Hint 6: The following ViewModel may be useful:

```
class MyViewModel : ViewModel() {  
    private val repository: WikiRepository = WikiRepository()  
  
    var wikiUiState: Int by mutableStateOf(0)  
    private set  
  
    fun getHits(name: String) {  
        viewModelScope.launch(Dispatchers.IO) {  
            val serverResp = repository.hitCountCheck(name)  
  
            wikiUiState = serverResp.query.searchinfo.totalhits  
        }  
    }  
}
```

Composable function then uses wikiUiState as normal observable state which can be directly drawn to the screen.