{EPITECH}

TRINITY

BOOTSTRAP

TRINITY

Creating & building an Android project

You'll call Github REST API and especially the endpoint GET repository *(for instance* `repos/googlesamples/android-Camera2Basic`*)* then print the received data. You'll need to use Retrofit for your API calls.

First, you have to prepare your HTTP client:

- ✓ install Android Studio with the latest version of the Kotlin plugin installed.
- ✓ create a new project with an *Empty Activity* and the language Kotlin.
- ✓ create a data class `Repository` with some properties (`id`, `name`, `full_name` and `html_url` are required).
- ✓ create an interface for your endpoints.
- ✓ create a class who builds the *Retrofit* client and implement your interface.

Look at Retrofit documentation to know how to create the interface.
You have to use built-in annotations.

Now let's get the data form Github and show it:

- ✓ in the layout `activity_main` add `TextViews` and other components if necessary to show your data.
- ✓ in the `MainActivity` class you can now get data and set them in the view.

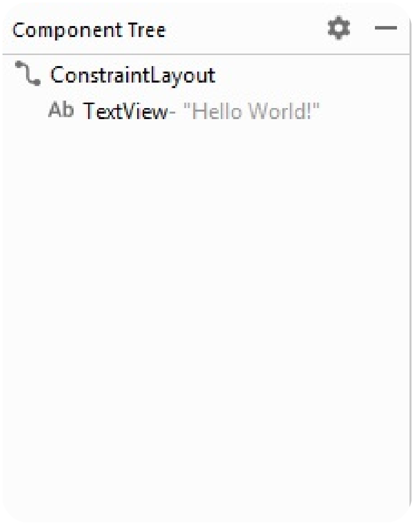
Now you're able to show a repository, you can add features to train yourself, like a list of repositories and the detail when you click on one element.

In this exercise, you only used `Activities` but `fragments` exist too.

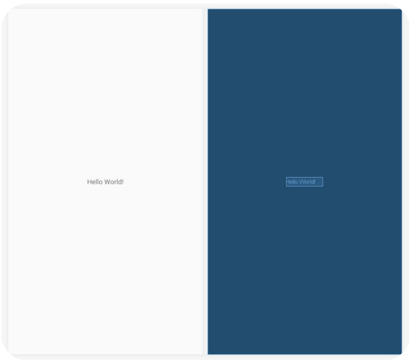
Your UI is very basic. To improve it, you have to understand how works user interface in Android and the XML syntax.

The user interface uses a hierarchy of Layouts and Widgets. You can see Layouts as containers positioning Widgets on the screen. Widgets are components like `TextView`, `Button` or `ImageView`.

Previously you used the file `activity_main.xml`. You can see a component tree in this file.



In this tree, the layout `ContraintLayout` is the container and the widget `TextView` is the component. You can change the position of elements with the properties. By default the layout takes the size of his parent (the screen in this example) so the widget is in the middle of the screen/layout.



Properties are useful to move elements, but also to custom background color or visibility for instance. Some elements have specific properties like `TextView` with `textColor`, `textSize`,...

If you need help to create your interface, you can use material design.

Now, create a nice and handy interface.

v 2.1

{EPITECH}