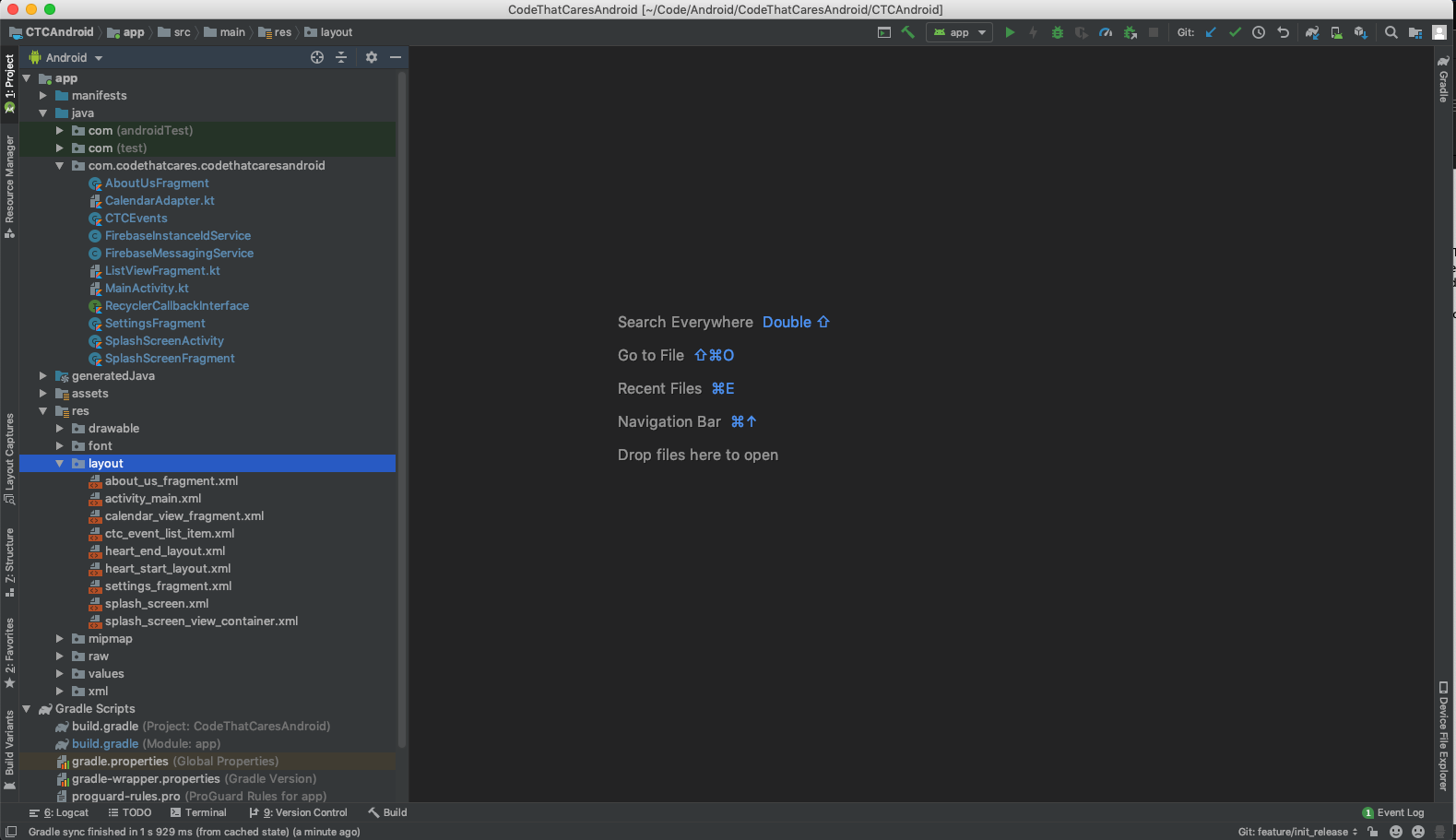
Welcome to Code that Cares! This is a small tutorial to get you introduced to Android Studio, we will be going through some basics of android development, as well as your IDE. If you do not have android studio, you can download it from this link [here](https://developer.android.com/studio).

**Getting to know android studio:**



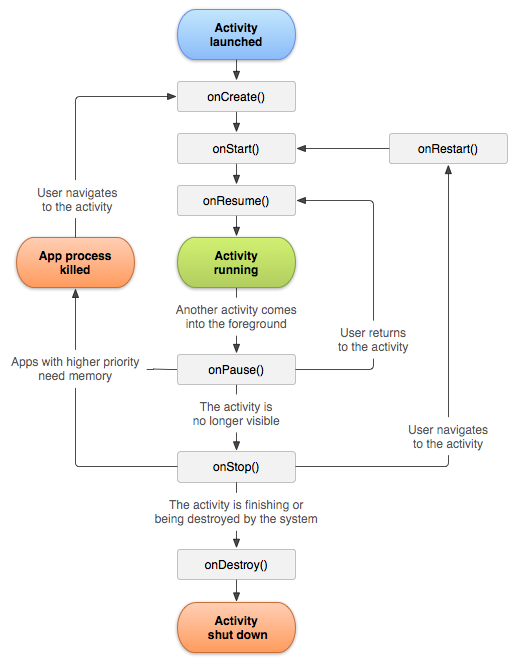
This is what a typical android studio project might look like. The java files are under the ‘java’ folder and the layouts are described in the layouts folder. In order to run your project, you will hit the big green arrow at the top to run it on your phone connected via USB. If android studio does not see your phone, check out this [link](https://developer.android.com/studio/debug/dev-options) to enable developer mode on your android phone. If you do not have an android phone, you will need to [install an android](https://developer.android.com/studio/run/emulator) emulator to test the projects. There are many other useful tools in android studio, but I find the best way to learn is to dive right in. You’ll learn many more of the features as you continue your journey in android development.

**Android Studio Projects**

A simple android application consists of java/kotlin classes, resource files for layouts, and gradle files.

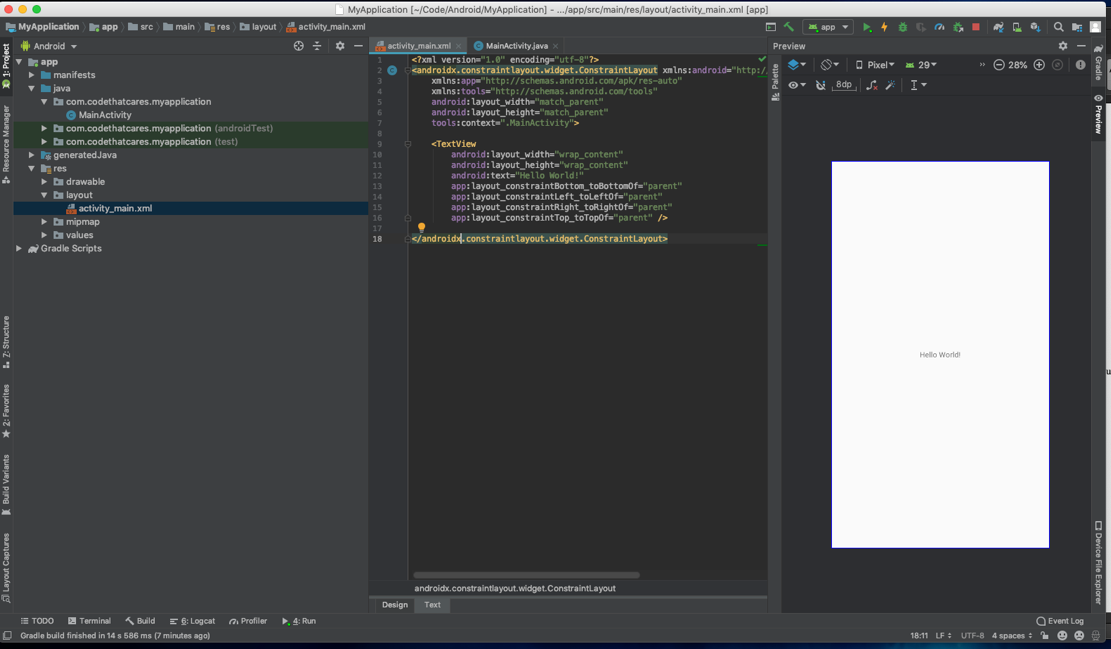
* Gradle: Gradle is a dependency manager. Gradle controls what packages are imported into the project, the java version, and other information such as SDK version and the version of the app. You will likely not have to do much in the gradle files outside of adding libraries you would like to use in your project, but it helps to know what it is.
* Resource files: Resource files (XML) define many things, such as projects layouts, themes, colors, string literals, and vector drawables. The main use of resource files is for layouts, which define what components should be where on the screen.
* Java files: Java files are where you will programmatically interact with the views defined in the resource files, as well as data. Java files in android projects can inherit lots of functionality based on what they extend. Right now, we will only focus on the Activity component, but others such as Services and Receivers exist and will become important as you make more complex projects.

**Activities**

Activities are one ‘screen’ in android. Take an email application for example there could be one activity for viewing a list of your emails, one for viewing a specific email, and one for creating an email. Activities have their own lifecycle, so certain methods will get called at certain points of that screen’s lifecycle. In order to better understand everything brought up in this mini tutorial, we will be making a hello world app.

**Making Hello World**

* Open android studio and select ‘Start a new android studio project’
* Select ‘Empty Activity’
* Name the application HelloWorld, save it where you wish
* Select your desired language, likely java unless you know kotlin
* Select the min sdk to be 22
* Make sure ‘uses androidx artificats’ is selected and ‘this project supports instant apps’ is NOT selected
* Click finish
* Allow android studio to do its thing and make the necessary files
* The default empty project is a Hello World application, press run and see what it looks like!
* Lets review what the code does to get us to this point. Open up 2 files
  + MainActivity.java under java/<packagename>/MainActivity
  + activity\_main.xml under rese/layout/activity\_main.xml
* **activity\_main**: Looking at the resource file, make sure you have the ‘text’ tab selected. Android studio includes a ‘Design’ tab, however it is pure trash and you shouldn’t use it.
* Your editor should look like this



* This is what a layout resource file. There is one root and then element and many child elements. This root is a [ConstraintLayout](https://developer.android.com/reference/android/support/constraint/ConstraintLayout), which is a more complex but is very powerful. For the future tutorials we will be using more basic [LinearLayout](https://developer.android.com/guide/topics/ui/layout/linear) and [RelativeLayout](https://developer.android.com/guide/topics/ui/layout/relative).
* The child element is a TextView which does exactly what it sounds like, displays text. In this case the text is Hello World
* Now lets examine what the java file **MainActivity** does
* There is a function called onCreate(Bundle savedInstances) that gets called when the activity
  + Within this method the support class is called, and the resource file that defines the layout is passed via setContentView(R.layout.activity\_main)
* All of the other methods that are contained in Activity (such as onStop() onPause() and onDestroy()) are still called at their appropriate times, but but since they aren’t overridden the one in the super class is called.
* That’s the end on the intro tutorial, the next one will be a making a list of recent earthquakes.