

CIS 3515 Assignment 1

Instructions:

Ensure that we have a functional development environment.

1. If this is your personal machine, ensure that you have a recent version of the Java Development Kit installed. Visit <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html> and download the latest version for your platform.
2. Install Android Studio. Visit <https://developer.android.com/studio> and download Android Studio for your platform.
 - The setup will install the IDE, SDK, and set up a default virtual (emulated) device.
 - If you are using a machine that already has Android studio installed, move on to the next step.
3. Test the environment by creating a new Android project.
 1. Launch Android Studio
 2. If presented with the Quick Start dialog, select **Start a new Android Studio Project**
 3. If presented with the development interface, start a new project by clicking **File -> New Project...**
 4. On the *Choose your project* screen, select **Empty Activity**. Click on **Next**
 5. For *Application name*, enter **HelloWorld**
 - For Package Name, enter **edu.temple.helloworld**
 - For Save *location*, enter a location where you have write permission, and where your project files will be safe from future deletion, such as your user directory
 - For *Language*, select **Kotlin**
 - Select API 27 as the Minimum API level.
 - Click on **Finish**
 6. Once your application has been configured by Android Studio, test it by clicking on the Run App button (the green **play** button in the toolbar)
 7. If presented with a dialog to *Select Deployment Target* dialog, ensure an **Available Emulator** is selected. If no Emulators are listed, launch the Device Manager and follow the steps to create a new virtual device (Ask the TA for assistance).
4. In the Project view of your IDE under **app**, expand the **res->values** folder and open the **strings.xml** file
5. Add a new XML item under **resources** resembling the following. Use the following

information:

1. *name*: my_greeting
2. *value*: Hello, World. This is my first app!

Eg:

```
<resources>
    <string name="app_name">Hello World</string>
    <string name="my_greeting">Hello, World. This is my first app!</string>
</resources>
```

6. In the **Project** view, expand the **res->layout** folder and open the **activity_main.xml** file
7. Perform one of the following and save the file
 1. Select the “Hello World” TextView that was automatically generated for you
 2. In the *Attributes* pane, set the TextView’s ID attribute to “**textview**”
 3. In the *Attributes* pane, replace the Hello World **text** value with the following:
“**@string/my_greeting**”
or
 4. Click on the **Code** tab above the Attributes pane (top-right of window), find the **TextView** tag, and replace the value of the **android:text** property with **@string/my_greeting**
8. Run your application again and note the change to the displayed text
9. Share your project to GitHub and commit all work done so far with a descriptive message.
Make the repository private and add the TA to the project as a Contributor so that they have access
VCS → Share Project on GitHub
10. Once completed, Add a button to the layout and arrange it within the layout so that it is centered horizontally and vertically.
11. Ensure that the button’s ID attribute is set to “**button**”
12. Change the button’s **text** attribute “*Click Me*”
13. Change that button’s **click** behavior to have it update the text attribute on the TextView object to read “You have clicked the button” by adding the following code to your activity’s onCreate() method as the last statements:

```
val textView = findViewById<TextView>(R.id.textview)
findViewById<Button>(R.id.button).setOnClickListener{textView.text = "You clicked
the button"}
```

Make sure the code above comes ***after*** the call to setContentView().

14. Commit the new changes to GitHub with a descriptive message.

Submit your GitHub project URL to Canvas before the deadline. No Commits are allowed after the deadline. Any commits after the submission deadline will incur a penalty, potentially leading to no points being awarded for the assignment.

Rubric

Error-free layout with centered button	20%
Using string resource (@string/my_greeting)	20%
Click behavior for button implemented	30%
Project shared to GitHub with 2 commits	30%