Assignment 2

Re-submit Assignment

Due Jan 26 by 11:59pm **Points** 100 **Submitting** a website url

Objectives

1. Design a single activity app with an appropriate layout (will be covered in week 1 and 2)
2. Implement multiple Android Views that update each other via any state change (will be covered in week 2)
3. Unit test via JUnit and UI test via Espresso (will be covered in week 3)

Overview

For your first Android app, your job is to design and develop a tip calculator to demonstrate to a client or potential employer your abilities.

Requirements

1. The calculator shall have the following inputs:
    1. Bill total
    2. Tip amount
    3. Number of people in the party that will split the bill
2. The calculator shall have the following outputs:
    1. The total amount
    2. The total amount per person
3. The calculator's design shall be easy to use and aesthetically pleasing
4. Using advanced widgets for the inputs for Tip amount and Number of people in the party to split the bill (e.g. Sliders or Spinners) will earn you more points than using only EditText views.
5. You must include at least 5 unit test cases to demonstrate correct functionality. Android's **documentation** **(https://developer.android.com/training/testing/start/index.html)** provides concise examples of how to perform testing.
6. You must include at least 3-5 Espresso integration UI tests.

Grading

1. Layout - 35%: does the Tip Calculator have all the needed Views (widgets and buttons)? Are they aligned well and work on different sized devices?
2. Logic - 30%: does the Tip Calculator work as expected?
3. Tests - 20%: are tests included? Do the tests cover all cases and demonstrate correct functionality?
4. Style - 10%: See the style section below.
5. Screen shots- 5%: Include at least two screen shots in your repository that demonstrate your app's functionality. You may also record a short video or clip if you prefer.

Style

You will be graded on;

1. Your code is easy to understand. This includes naming of file, variable, and method names.
2. Logic is not duplicated (i.e. don't have 10 separate methods that essentially do the same thing. Think about how you can write less code to accomplish more).
3. Are you referring to strings via the strings.xml file?
4. Are you using Material Design elements, including a ScrollView?
5. Are your test cases named well and grouped into methods in a way that makes sense?

Repository and Submission

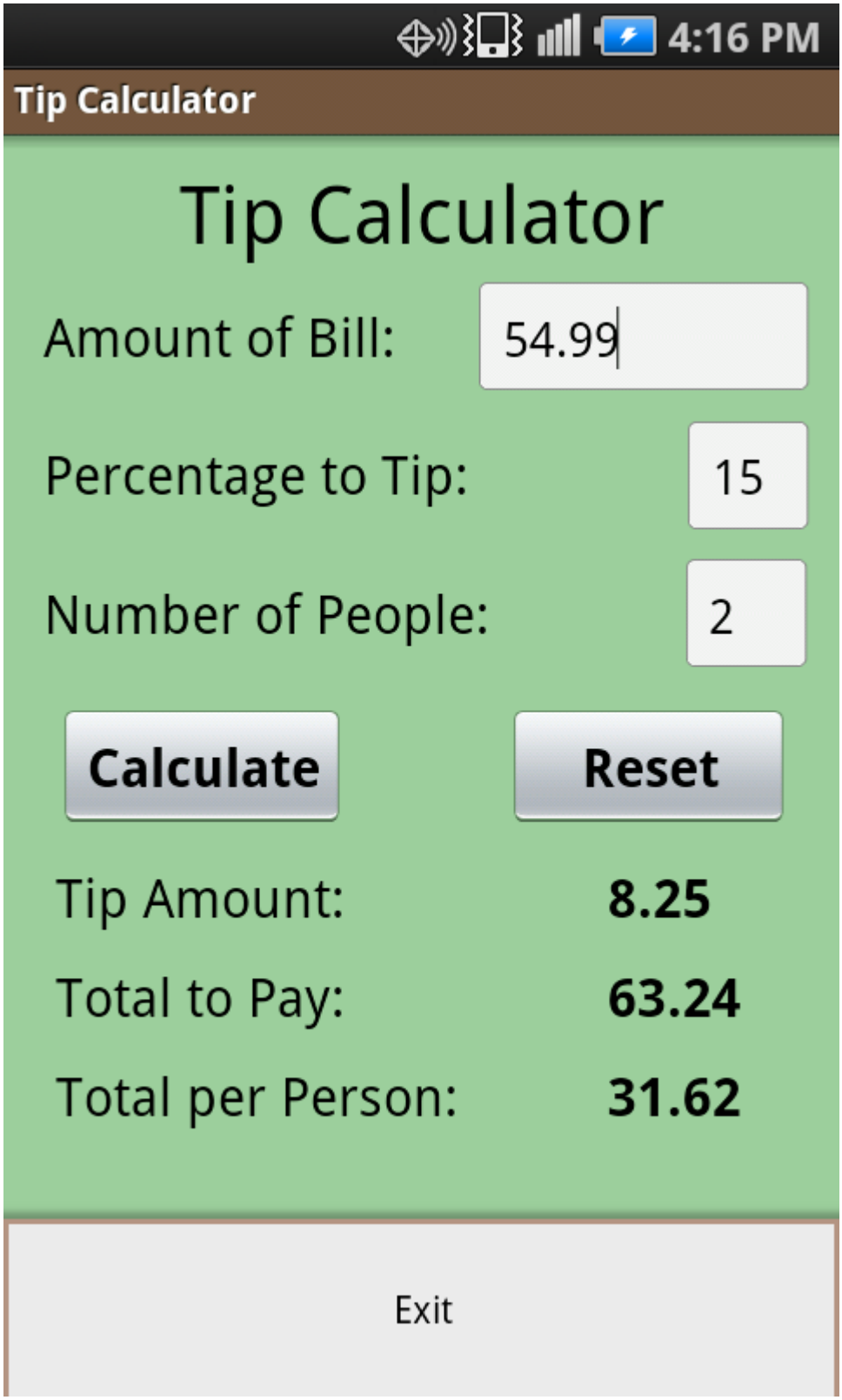
You will use the same repository for all assignments. Accept the assignments repository using the link below. An empty repository will then be generated for you. Create a folder named Assignment2 in your repository and use that folder to house this project.

https://classroom.github.com/a/cHfEI2ay (https://classroom.github.com/a/cHfEI2ay)

To submit:

1. Commit and push your repository to master on GitHub.
2. Create a tag/release for Assignment2 submission on GitHub.
3. Submit the URL to your release in Canvas.

Example (but do not constrain yourself to it as it only meets the very minimum of requirements - doesn't use any advanced widgets):



TipCalculator Rubric

Criteria	Ratings	Pts
Layout Layout - 35%: does the Tip Calculator have all the needed Views (widgets and buttons)? Are they aligned well and work on different sized devices?		35.0 pts
Logic Logic - 30%: does the Tip Calculator work as expected?		30.0 pts
Tests Tests - 20%: are tests included? Do the tests cover all cases and demonstrate correct functionality?		20.0 pts
Style Style - 10%: Your code is easy to understand. This includes naming of file, variable, and method names. Logic is not duplicated (i.e. don't have 10 separate methods that essentially do the same thing. Think about how you can write less code to accomplish more). Are you referring to strings via the strings.xml file? Are you using Material Design elements, including a ScrollView? Are your test cases named well and grouped into methods in a way that makes sense?		10.0 pts
Screen Shots Screen Shots or Video - 5%: Include at least two screen shots in your repository that demonstrate your app's functionality. You may also record a short video or clip if you prefer.		5.0 pts
		Total Points: 100.0