

ITC 162 Grading Criteria (please read carefully and ask for clarification if needed).

Does the code run? – 50 points

- No more than 20 points will be given if the code doesn't run.

Correctness of code – 40 points

- Does the code perform per requirements as specified in the assignment?
- Does the program produce the correct prompts to the user?
- Does the program calculate the correct values?
- Does the program display the results correctly?
 - * As applicable for the assignment

Clarity of code – 10 points

- Does the program use good names for variables?
- Is the code commented and spaced so that it is easy to read and understand? Use blank lines as appropriate.

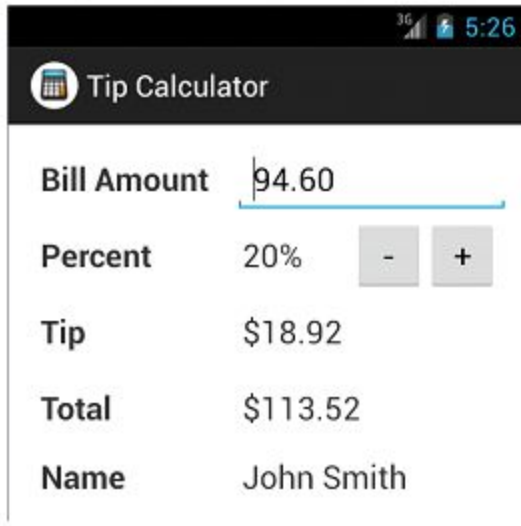
Incorrect submission

- Your assignment will not be graded if it is not submitted in the right place on Canvas or if the submission is not in the file format specified for the assignment.
- You will be allowed to resubmit, however you might lose points due to late submission.

Use an EditTextPreference element

1. In this exercise, you'll modify the Tip Calculator app presented in this chapter so it uses an EditTextPreference element. Although this type of preference wasn't described in this chapter, it works similarly to the CheckBoxPreference element, and you can look it up in the documentation for the API if necessary.
2. When you're done, a test run should look like this:

3.



The screenshot shows the 'Tip Calculator' app interface. At the top, there is a status bar with '3G' signal, a battery icon, and the time '5:26'. Below the status bar is a dark header with a calculator icon and the text 'Tip Calculator'. The main content area has a light gray background and contains the following elements:

- Bill Amount:** A text input field with the value '94.60' and a blue underline.
- Percent:** A label 'Percent' followed by the value '20%'. To the right of '20%' are two gray buttons with black text: a minus sign '-' and a plus sign '+'. The buttons are slightly offset to the right of the '20%' text.
- Tip:** A label 'Tip' followed by the value '\$18.92'.
- Total:** A label 'Total' followed by the value '\$113.52'.
- Name:** A label 'Name' followed by the value 'John Smith'.

4. Start Android Studio and open the project named Week7/ch08_ex4_TipCalculator.
5. Open the preferences.xml file in the xml directory. Add an EditTextPreference element for a setting named Name. This setting should allow the user to enter his or her full name.
6. Run the app and use the Settings activity to enter your name.
7. Open the layout of the activity. Modify the layout, so it includes a fifth row that can display a name.
8. Open the class for the Tip Calculator activity. Modify this code so it gets the name from the preferences and displays that name in the fifth row.