**Soft-Boiled Eggs**

**Comment out code you wish considered for partial credit.**

Sometimes the context is not needed at all. Sometime it is. Try

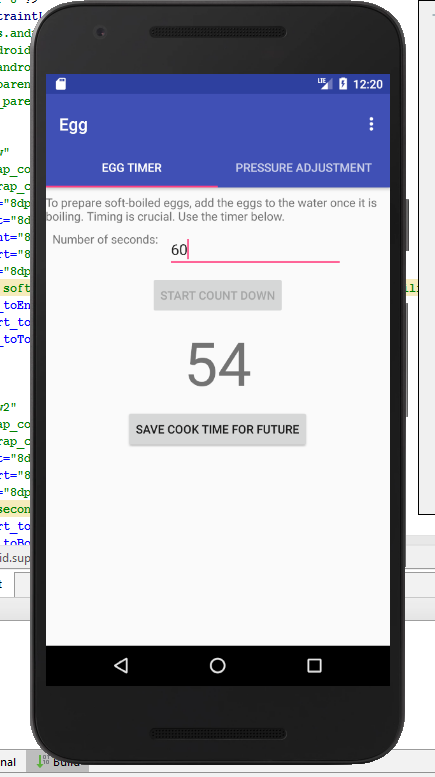
this 🡪 getContext() 🡪 getApplicationContext()

Create a Tabbed Activity with Action Bar Tabs Navigation Style

Eliminate the Floating Icon.

Have the tabs navigation display the names “Egg Timer” and “Pressure Adjustment”.

Eliminate references to third tab.



**Tab 1 – Egg Timer:**

The first tab layout should have TextView, then a TextView & EditText, then a Button, then a TextView (with large text size), and finally another Button. (They should have horizontal and vertical constraints.)

The first TextView is simply a brief header. Something like: “To prepare soft-boiled eggs, add the eggs to the water once it is boiling. Timing is crucial. Use the timer below.”

The next TextView says simply “Number of seconds: ” and is followed by an EditText that allows only positive numbers to be entered.

The first Button should say “Start Count Down” will establish a count down timer to the time indicated by the user. Recall the count down timer works in milli-seconds so all second values will have to be multiplied by 1000. The button should be disabled until the count down is complete and then be re-enabled.

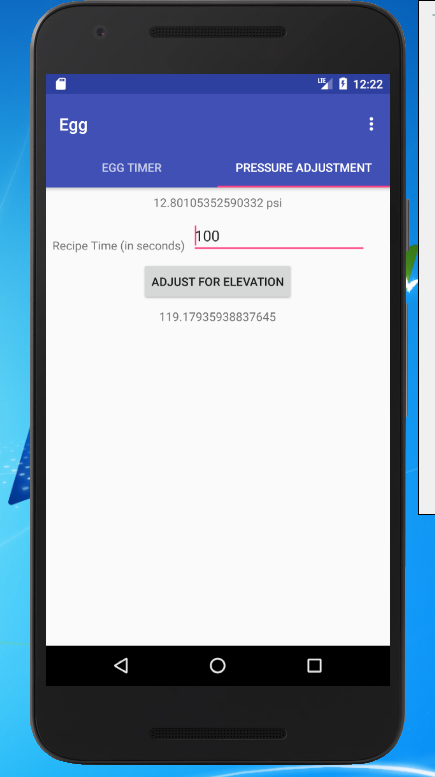
If the user types/clicks “Enter” in the Edit text it should have the same effect as clicking the button that starts the timer.

Every second (1000 milliseconds) the time should be decremented and displayed in a TextView.

When the count-down is complete, play the sound file submarine\_diving.mp3.

The second button should say “Save Cook Time for Future” and when clicked, it should store the user’s cook time as a Shared Preference. (Don’t forget to **apply**!). When the Fragment is being created, the SharedPreference should be read and placed in the Time EditText. Use a default amount of 300 if the user has not saved a preferred time. (Of course you will want to test with shorter times than 300 – which five minutes.)

**Tab 2 – Pressure Adjustment:**



The second tab layout should have a TextView to display the pressure, a TextView & EditText (allow positive integers only) to enter the Recipe Time (in seconds), a Button that says “Adjust for Elevation” that initiates a calculation of an “adjusted time”, and finally a TextView to display the adjusted time. (They should have horizontal and vertical constraints.)

In my code I extended Fragment and implemented Sensor Event Listener and it worked

**public class** PressureAdjustmentFragment **extends** Fragment **implements** SensorEventListener

Have a pressure sensor. Have its reading displayed in the top TextView after you have converted from hectoPascals to psi (pounds-per-square-inch)

1 Hectopascal = 0.0145038 psi

|  |  |  |
| --- | --- | --- |
| **Air Pressure**  **(hectoPascal)** | **Air Pressure**  **(psi)** | **factor** |
| 1013.5 | 14.7 | 0 |
| 965.3 | 14.0 | 0.0707 |
| 896.3 | 13.0 | 0.1717 |
| 827.4 | 12.0 | 0.2727 |
| 758.4 | 11.0 | 0.3737 |
| 689.5 | 10.0 | 0.4747 |

The EditText is for the user to enter a time in seconds called for in a recipe which is presumably written for sea-level which corresponds roughly to 14.7 psi.

When the first button is clicked, take the amount the user entered in the EditText and calculate the factor (based on the Pressure in psi) and then the new time. Display the new time in the TextView below.

(I was taking my pressure from the above display and splitting it on a space – since I had a number then a space and then “psi”.)

factor = 0.101 ( 14.7 – press\_psi )

new\_time = orig\_time \* (1+factor)

If the user types/clicks Enter in the EditText, that should have the same effect as clicking on the Calculate button.