

Assignment 1

[Submit Assignment](#)

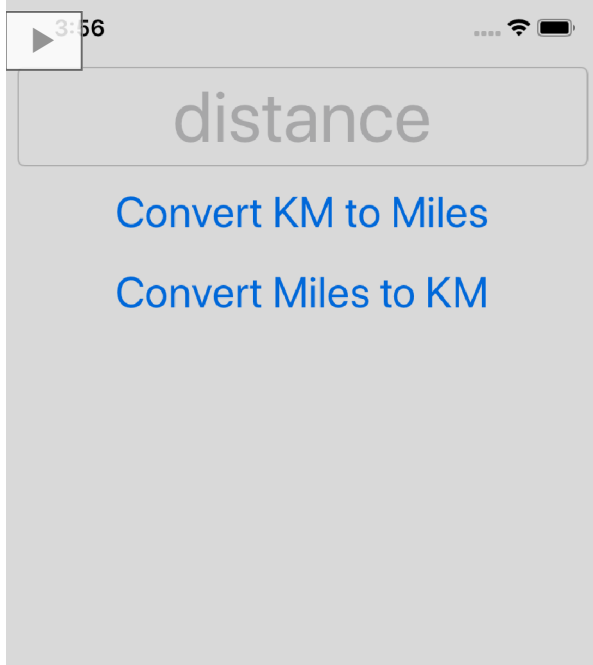
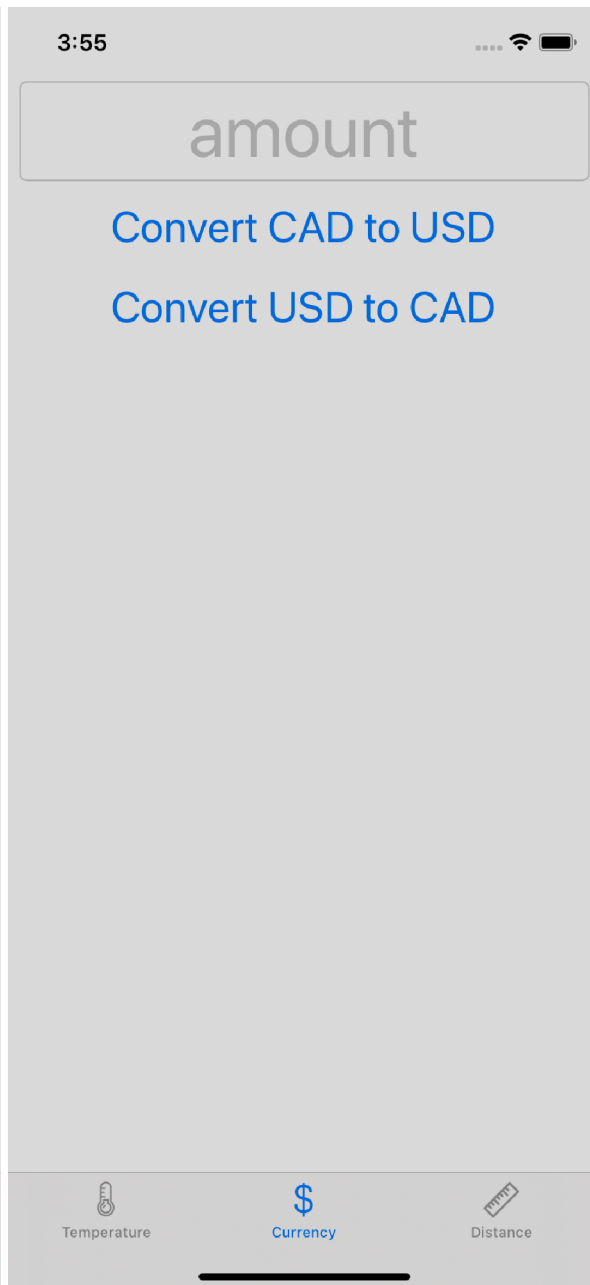
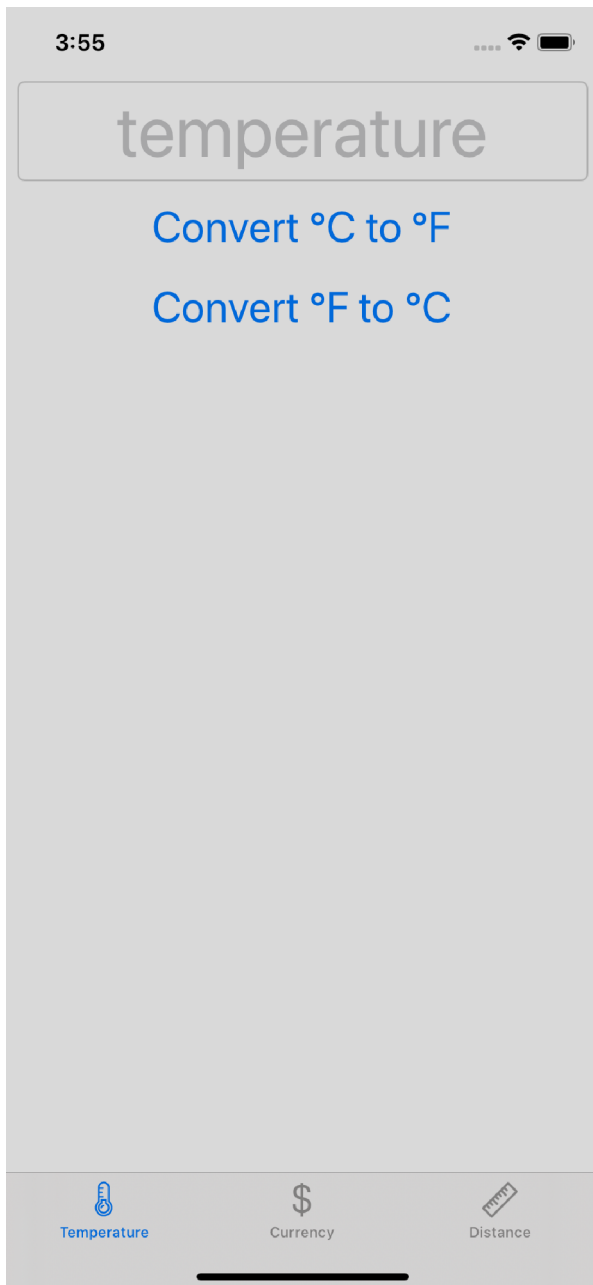
Due	Thursday by 11:59pm	Points	10	Submitting	a file upload	File Types	zip
------------	---------------------	---------------	----	-------------------	---------------	-------------------	-----

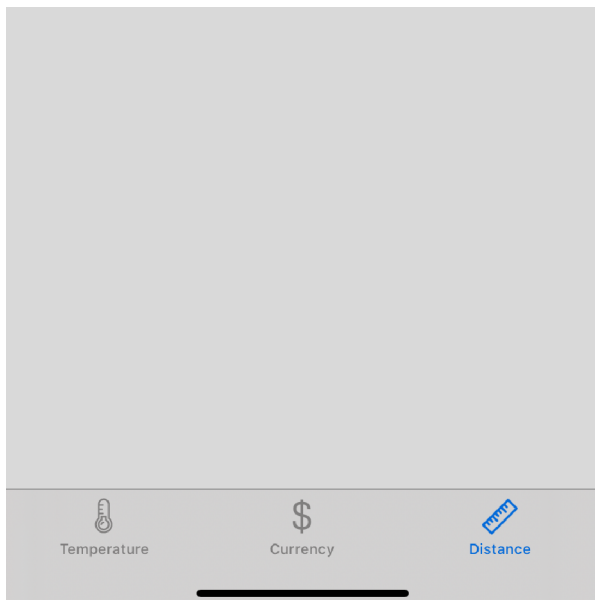
Assignment to be done individually. Write an app that allows the user to make various conversions between different units of measure. The app must be properly commented. Each method you create must have a brief description preceding it. Provide comments for any tricky code. Good comments should not restate code, but explain the code and potentially explain why the code is doing what it is doing.

The user will be able to enter a value on each tab and press either convert button to be presented with the converted value. On each tab, use a gesture recognizer to allow the user to dismiss the keyboard by tapping the background.

The app has 3 tabs that look approximately like below. There is also a label on each tab under the 2nd button. This label is used to display the results of the conversions.







When complete, compress your project folder into a single zip file and submit to the Dropbox.

Marking Scheme

Adequate and appropriate comments	2
GUI elements	3
View controller implementations	5
Total	10

Converter

- Need a Mac? See [Student Resources](#).
- Create an iOS Application using the Tabbed App template.
 - Product Name: Converter
 - Language: Swift
 - Include Unit Tests: Unchecked
 - Include UI Tests: Unchecked
- Images for app: [01_converter_images.zip](#). Add the images to the image asset catalog and use them for the tab bar items.
- Use the provided app icons.

TemperatureViewController

- Rename FirstViewController to TemperatureViewController.
- Add the following statement of authorship as a comment to the top of TemperatureViewController.swift replacing Student Name and 123456789 with your name and student number. **Failure to include a statement of authorship will result in a grade of 0.** These two lines must be the first two lines of the file.

- I, Student Name, student number 123456789, certify that this material is my original work. No other person's work has been used without due acknowledgement and I have not made my work available to anyone else.
- To get the ° (degree) symbol, use the Emoji & Symbols tool, which is on the Edit menu.
- Use the appropriate built in Measurement and Unit class for conversion.

CurrencyViewController

- Rename SecondViewController to CurrencyViewController.
- For the currency conversion, hard code the conversion factor at \$1.00 USD = \$0.746897 CAD.

DistanceViewController

- Add a 3rd view controller scene named DistanceViewController.
- Use the appropriate built in Measurement and Unit class for conversion.

View Controllers

- Each of the 3 view controllers operates in the same way.
- They will each have 2 outlets, 1 for the text field and the other for the output label.
- They will each have 3 action methods, 2 for conversion and 1 to dismiss the keyboard.
- A number formatter will need to be used to format the output correctly. The temperature and distance view controllers output the results with 1 decimal; the currency view controller outputs the results with 2 decimals.
- Each conversion method will attempt to convert the input into a usable double and then proceed with the conversion.
- If the input can not be converted into a usable double, display an error to the user along with the value that can't be converted.

Main.storyboard



Use auto layout constraints to ensure that the UI looks correct on any sized iPhone.

- Set the font size for the text field to 50.
- Set the font size to 30 for the buttons.
- Set the font size to 25 for the result label.
- The text field and result label must have their text centred.
- Show the decimal keyboard for the text field.

Running

While running, the tabs should look approximately like this:



