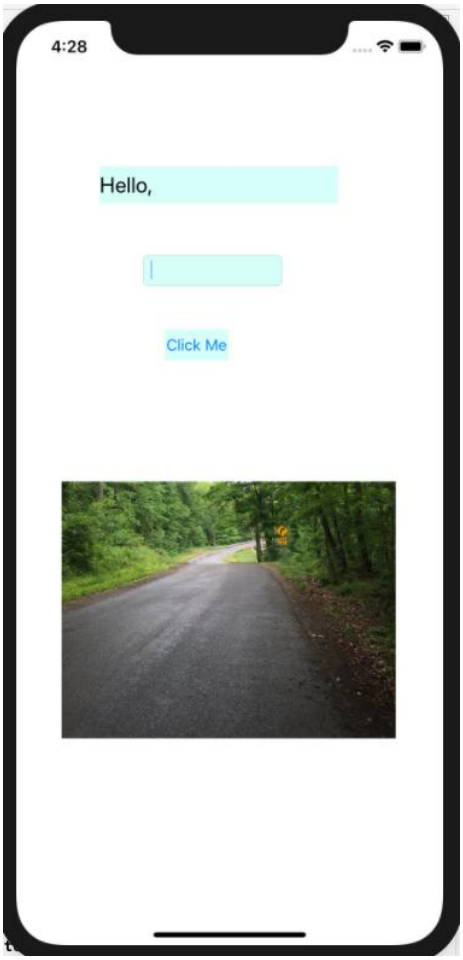
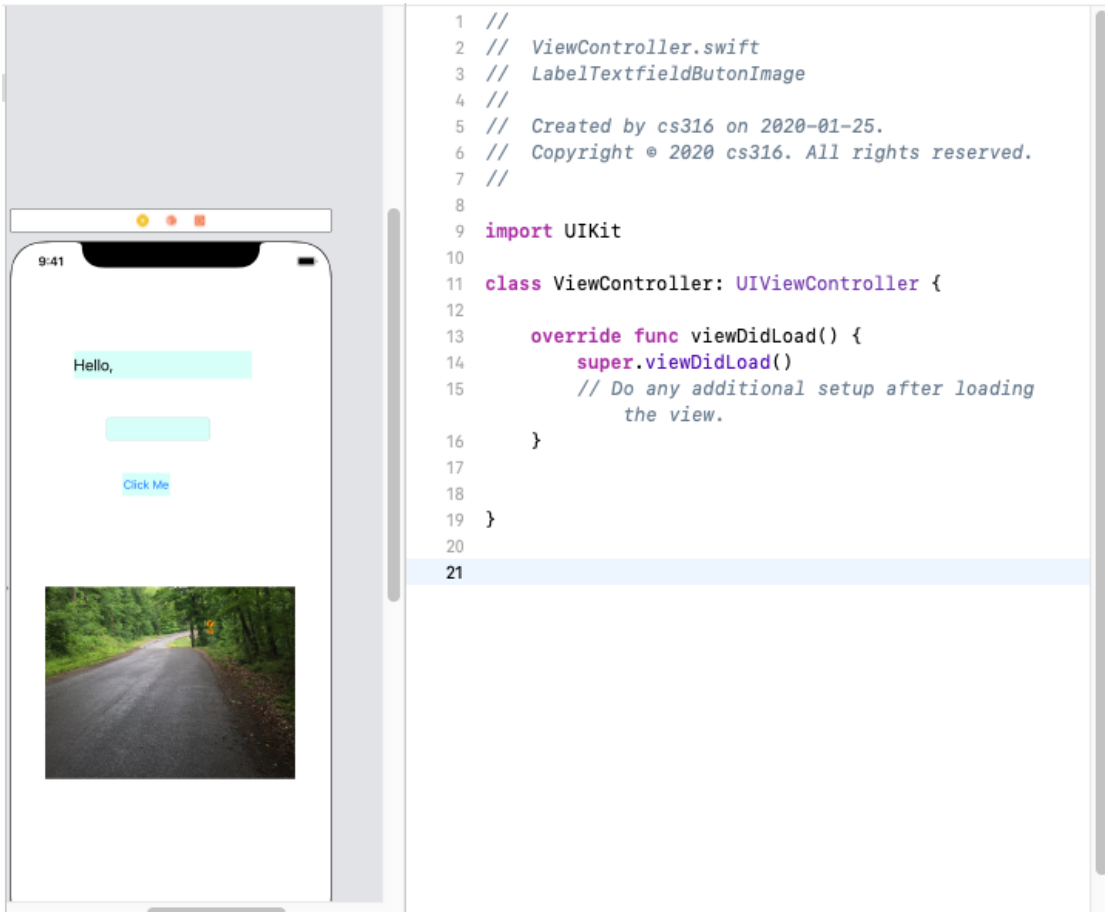
In-class activity:

1. Create a single-view iOS application using one UILabel, UITextField , UIButton and UIImage. With a given image (forest_park.jpg), the layout should look like this below:



2. Connect storyboard GUI components (those used as input, and output outlets as well as triggering some action; in this case, the UILabel is used for output, the UITextField is used for input and the UIButton is used to trigger for an action) to the relevant swift file (i.e. ViewController.swift) .

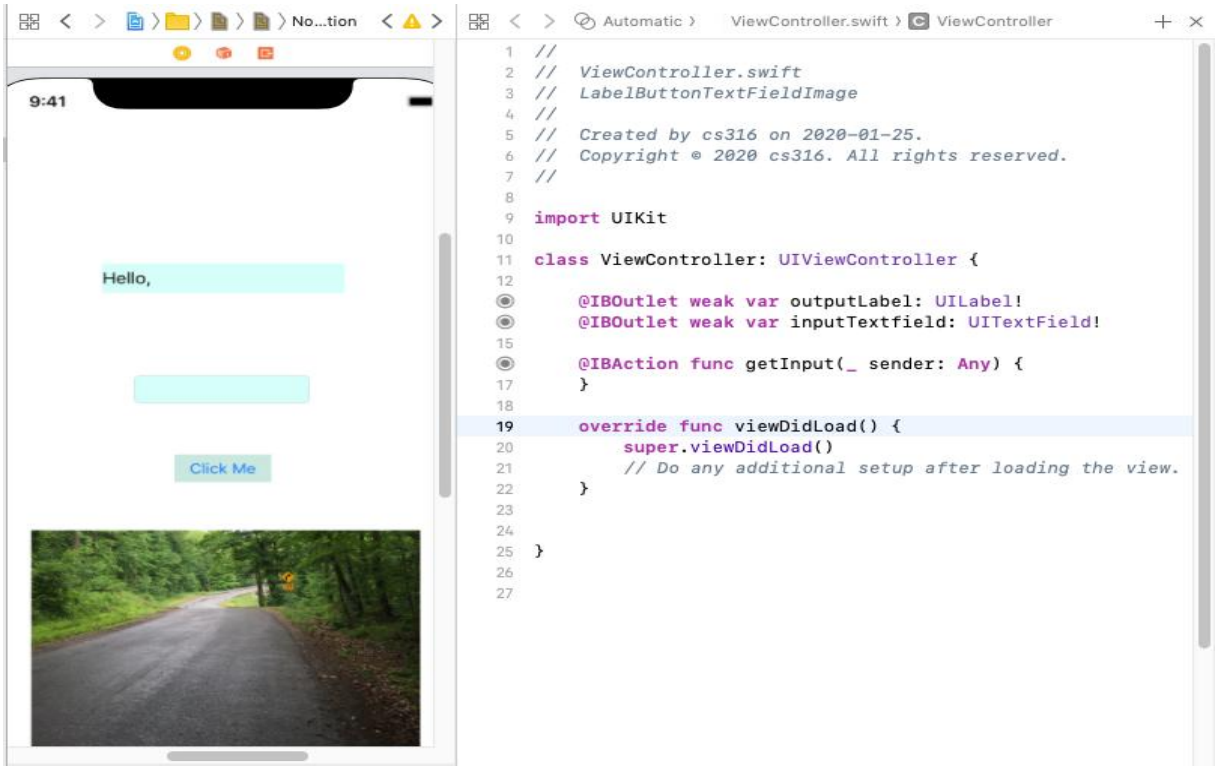
    i. Click on Main.storyboard, then open Assistant Editor. Make sure that the storyboard layout and the ViewController.swift file are displayed side-by-side as follow:



For each outlet connection, Contol-Click on the GUI object to get a pop-up menu, then Contol-Click on “New Referencing Outlet’s circle area” and Control-drag to the file (between the class heading and the viewDidLoad function,(ie. Line 12 in this example). Provide a proper name and then click connect button.

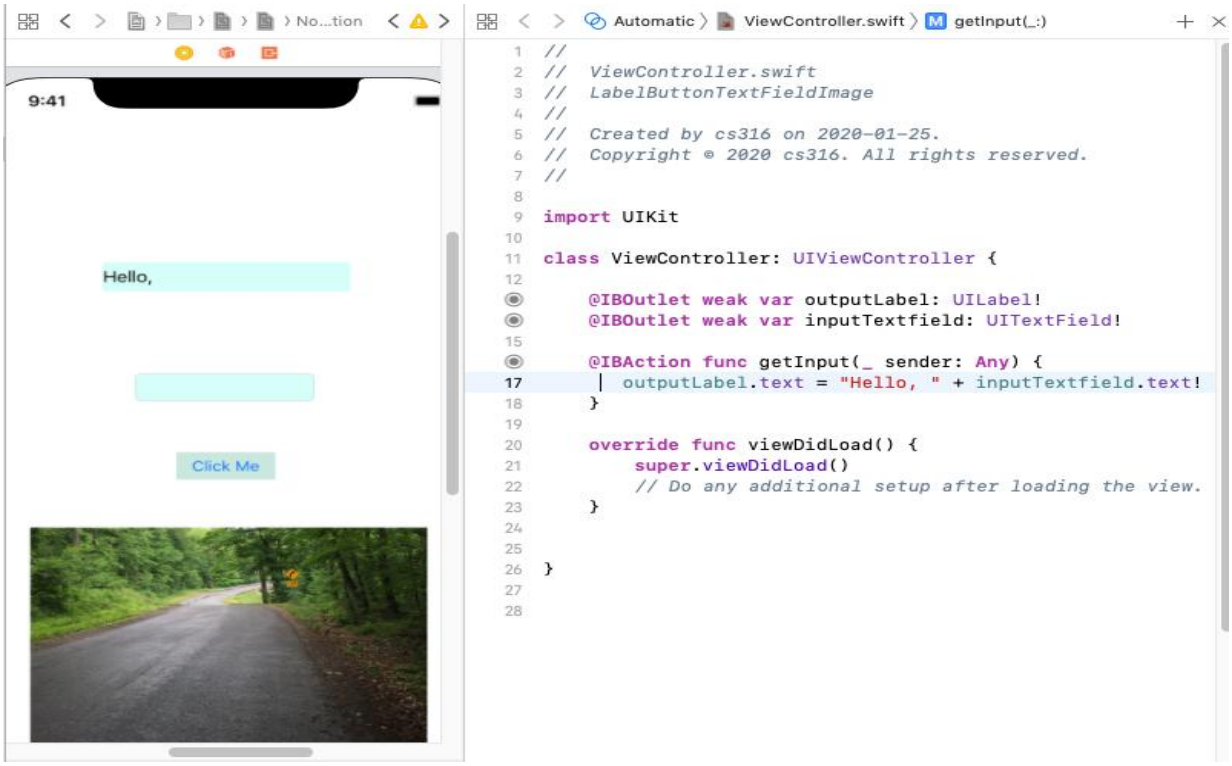
For each action connection, we do very much the same except that we choose a triggering event like “Touch Drag inside” or other event instead of a new referencing outlet. Provide a method name and click the connect button. This method will be executed when the GUI object is “touch-drag-inside”

When you finish, your screen should look like this:



ii. Now complete the “Action” function (I called it “getInput”) with appropriate swift statement(s) to display the input text on the Label.

outputLable.text = “Hello, “ + inputTextfield.text



iii. Run you app, you should get a similar output:

iv. Next adopt a protocol named UITexttFieldDelegate. Firstly, change the class heading to like below:

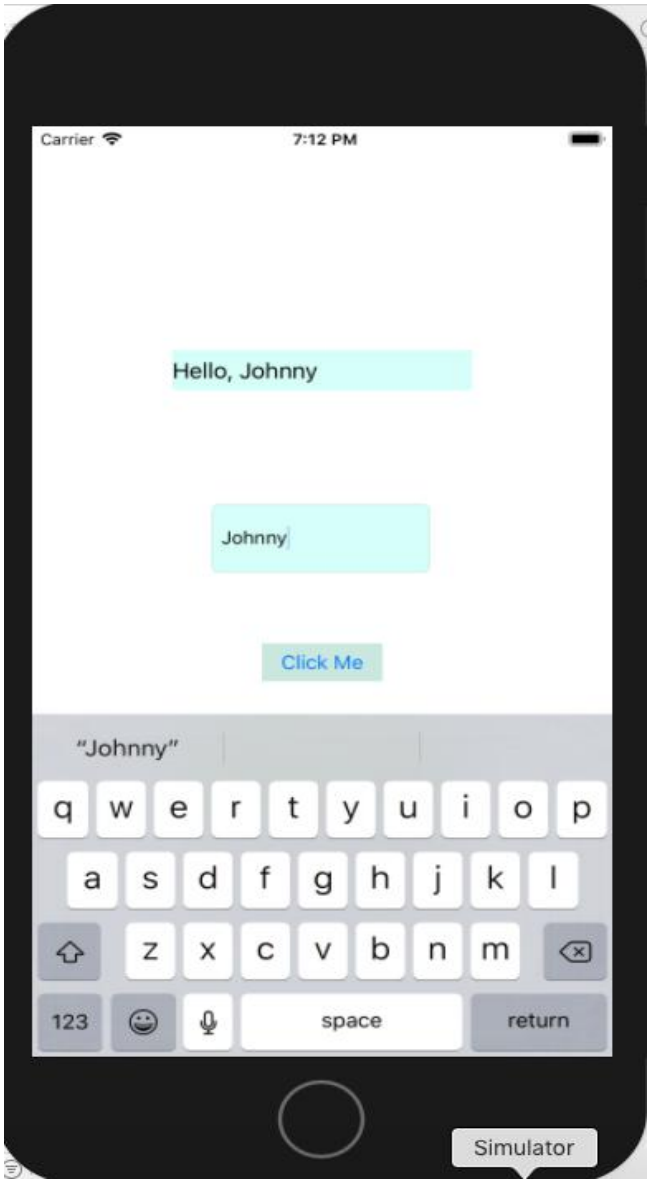
```
class ViewController: UIViewController, UITextFieldDelegate {
```

Then implements the following methods:

```
override func viewDidLoad() {
    super.viewDidLoad()
    self.inputTextfield.delegate = self    //***
  }

  override func didReceiveMemoryWarning() {
    super.didReceiveMemoryWarning()
    // Dispose of any resources that can be recreated.
  }
```

```
override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?) {
    self.view.endEditing(true)

}

func textFieldShouldReturn(_ textField: UITextField) -> Bool {
    inputTextfield.resignFirstResponder()
    return true
}
```

v. Run the program:



Note: If the Software Keyboard doesn't show and we need it, we an get it by going to the Simulator's menu → Hardware → Keyboard → Toggle Software keyboard.