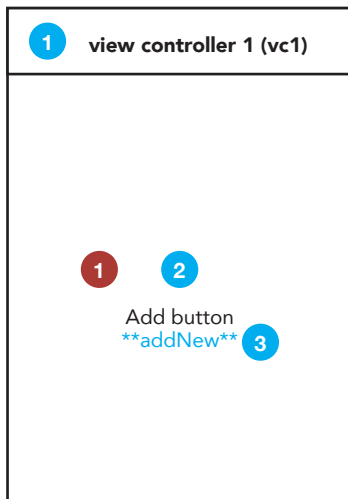


DELEGATE

(gets info)



- 1 Add button pressed action, 2 set the AddViewController delegate to reference this view controller 1

```
@IBAction func addNew(sender: AnyObject) { 3
    // Identify view controller we will transition to and get information from
    let vc = self.storyboard?.instantiateViewControllerWithIdentifier("addVC") as AddViewController
    // Point the AddViewController delegate variable to the current ViewController
    vc.delegate = self
    // Switch to the AddViewController (modally in this case)
    self.presentViewController(vc, animated: true, completion: nil)
}
```

- 5 Add the protocol 4 we defined in the AddViewController to our ViewController.

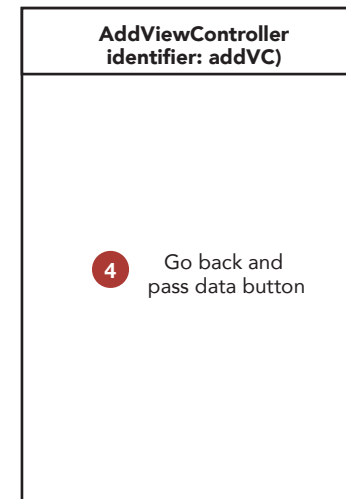
```
class ViewController: addVCDelegate { 4
```

- 6 Define method that the AddViewController will use to communicate with us.

```
func addVCDismissed(newItem: String) { 6
    // newItem is the data we passed from our other view controller.
    // We can now do anything we like with this data, like update our labels/table views.
    println(newItem)
}
```

DELEGATING OBJECT

(passes info)



- 2 Define the protocol 4 the first View Controller will adhere to

```
protocol addVCDelegate { 4
    func addVCDismissed(newItem: String) 6
}

class AddViewController: UIViewController {
```

- 3 Define the delegate 5 that will communicate with the first View Controller (this is what **vc.delegate** was setting in step 1)

```
class AddViewController: UIViewController {
    var delegate: addVCDelegate? 5
```

- 4 This is the final step that needs to happen as we go back to our first view controller. 1 The delegate property 5 lets us run the method 6 in our first view controller passing it our desired data.

```
@IBAction func addGoBack(sender: AnyObject) {
    // Run the function in the first view controller passing "some text"
    // "Some text" can be anything: a label value, string, array, etc
    self.delegate?.addVCDismissed("some text") 6
    // Dismiss this view controller and go back
    self.dismissViewControllerAnimated(true, completion: nil)
}
```