

iOS Precompiler 2013

Daniel H Steinberg
editorscut.com
dimsumthinking.com

Four Pillars

- Xcode
- Objective-C
- View Controllers
- Storyboards

Xcode

- Find Xcode
- Create a new project **PreCompiler**
- Single-View iPhone with ARC and Storyboards
- Look around a bit
- Run it

PreCompiler I

Object Library

- Color background black
- Drag in a label titled “Hello, World.”
- Add three buttons: Red, Green, Yellow
- Add constraints and attributes
- Provide Xcode labels
- Run

PreCompiler2

Outlets and Actions

- Use Assistant Editor to create **helloLabel** outlet in **DSTViewController** class extension
- Create actions **redButtonTapped** (etc) in DSTViewController implementation
- Add breakpoints
- Run

PreCompiler3

Properties

- Implement `redButtonTapped` to set `helloLabel`'s `textColor` to red
 - ```
(IBAction) redButtonTapped: (UIButton *)sender {
 self.helloLabel.textColor = [UIColor redColor];
}
```
- Repeat for the other colors
- Run

PreCompiler4

# Messages

- Disable a button after it is tapped
  - Create this method
    - ```
(void)disableButton:(UIButton *)tappedButton {  
    tappedButton.enabled = NO;  
}
```
 - Call it like this
 - ```
(IBAction)redButtonTapped:(UIButton *)sender {
 self.helloLabel.textColor = [UIColor redColor];
 [self disableButton:sender];
}
```
  - Run
- PreCompiler5

# Collections

- Create an Outlet collection named **colorButtons** for the three buttons.
- Reset the buttons before disabling the tapped button using block enumeration

```
- (void)disableButton:(UIButton *)tappedButton {
 [self.colorButtons
 enumerateObjectsUsingBlock:^(UIButton *colorButton,
 NSInteger idx, BOOL *stop) {
 colorButton.enabled = YES;
 }];
 tappedButton.enabled = NO;
}
```

- Run  
PreCompiler6



# Gestures

- Add a left swipe gesture recognizer to our scene.
- Connect it to an action named **enableAllButtons**.
  - ```
(IBAction)enableAllButtons:(UISwipeGestureRecognizer *)sender {  
    [self disableButton:nil];  
}
```
- Run

PreCompiler7

Segue

- Add an up swipe gesture recognizer to our scene.
- Create a second scene with a text field and a cancel and save button
- Create a segue from the gesture recognizer to the second scene
- Run

PreCompiler8

Custom Class

- Create a subclass of **UIViewController** named **DSTInputViewController**
- Give it public properties named **backgroundColor** and **signInName**. Create an outlet for the textfield named **inputField**.
- Use the Identity Inspector to set the input VC's type.
- In **viewDidLoad** set the background color to the property.
- Run
PreCompiler9

Forwarding Info

- Set the Input View Controller's **backgroundColor** in **prepareForSegue:sender:**

```
- (void)prepareForSegue:(UIStoryboardSegue *)segue
    sender:(id)sender {
    if ([segue.identifier isEqualToString:@"showInput"]) {
        DSTInputViewController *inputVC
        = segue.destinationViewController;
        inputVC.backgroundColor = self.helloLabel.textColor;
    }
}
```

- Run

PreCompiler10

Moving Backwards

- Stub out these methods
 - (**IBAction**)cancelInput:(**UIStoryboardSegue ***)segue {}
 - (**IBAction**)completeInput:(**UIStoryboardSegue ***)segue {}
- Create unwind segues from the two buttons. Name them cancelInput and completeInput
- Run

PreCompiler I I

Backwarding Info

- In `prepareForSegue:sender:` set `signIn`

```
-(void)prepareForSegue:(UIStoryboardSegue *)segue  
    sender:(id)sender {  
    if ([segue.identifier isEqualToString:@"completeInput"]) {  
        self.signInString = self.inputField.text;  
    }  
}
```

- Implement `completeInput`

```
-(IBAction)completeInput:(UIStoryboardSegue *)segue {  
    DSTInputViewController *inputVC = segue.sourceViewController;  
    self.helloLabel.textColor = inputVC.backgroundColor;  
    self.helloLabel.text = inputVC.signInString;  
}
```

- Run
PreCompiler 2

Delegates

- Set the text field's delegate to be the input VC
- Implement this delegate method
 - ```
(BOOL)textFieldShouldReturn:(UITextField *)textField {
 [self performSegueWithIdentifier:@"completeInput"
 sender:self];
 return YES;
}
```
- Run

# iOS Precompiler 2013

Daniel H Steinberg  
[editorscut.com](http://editorscut.com)  
[dimsumthinking.com](http://dimsumthinking.com)