iPhone App Dev

Lesson 4

Source Codes

https://github.com/makzan/ios-dev-course-example

Contact

makzan@42games.net

Practice

- Browse the App Store. Find some apps that impress you. Discuss why they are good.
- Do you have any problem that want to solve in a mobile phone? What app do you want to develop?

Practice

- ✓ Design an utility app.
- ✓ Present it to the class in lesson.

Practice

- ✓ Think about how to make use of gestures.
- ✓ Design an app with gesture features.
- ✓ Or discover an app on app store with gesture features.
- ✓ Present it to the class in next lesson.

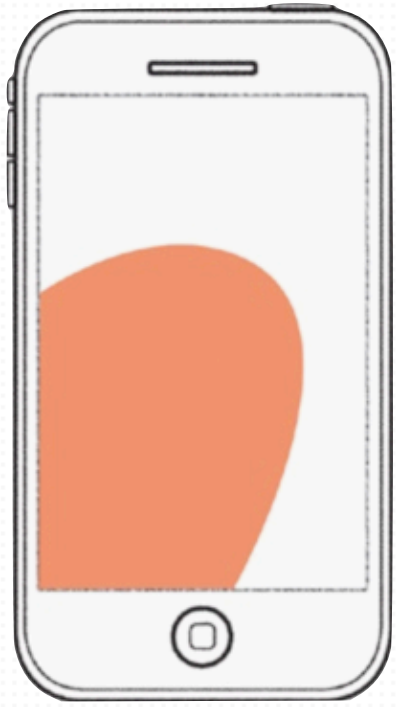
Summary

- Comfortable Thumb Area
- Methods Declaration and Implementation
- Properties Declaration
- Delegates
- View Cycle
- Using TextView
- Using UISlider
- Using UIPicker
- Using UIImageView

Comfortable Tapping Area

Comfortable Thumb Area

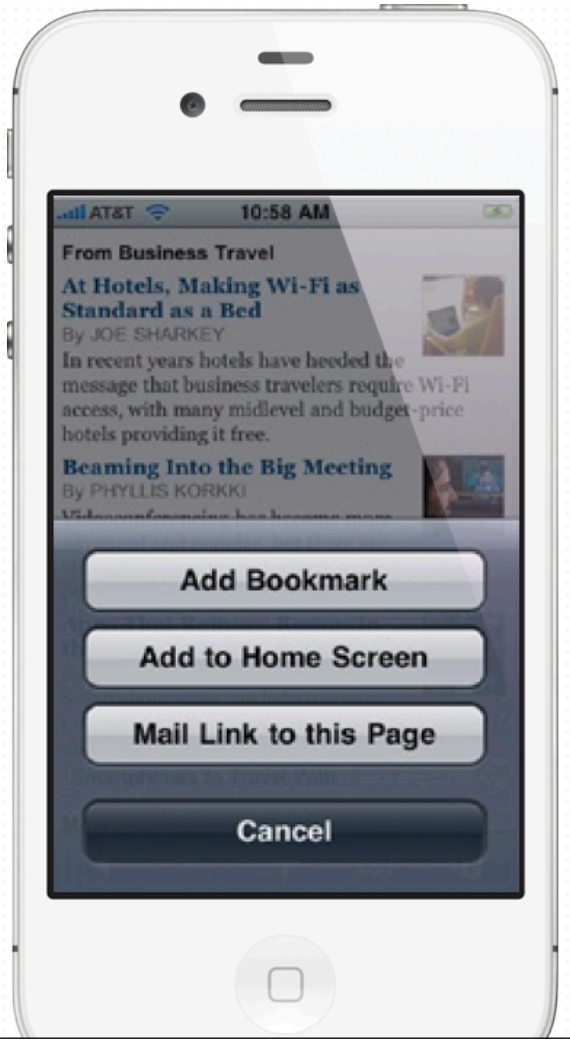
- Put usual buttons at bottom
- Put critical button at top



Graph from TapWorthy book.

Comfortable Thumb Area

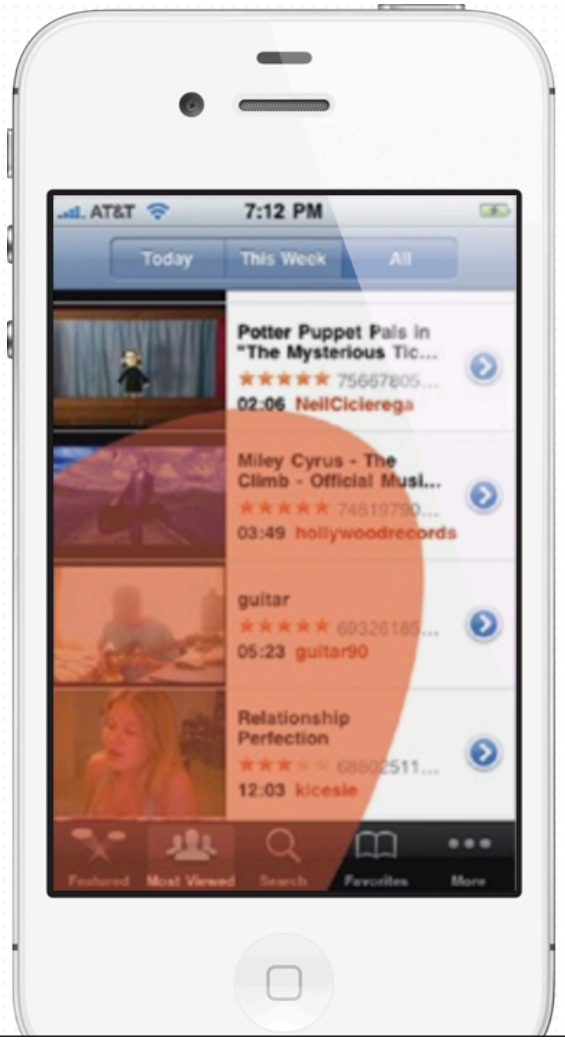
- Put usual buttons at bottom
- Put critical button at top



Graph from TapWorthy book.

Comfortable Thumb Area

- Put usual buttons at bottom
- Put critical button at top



Graph from TapWorthy book.

Methods Declaration and Implementations

Methods Declaration

- (void) sayHello;
- (BOOL) hasLoggedIn;
- (int) countTotalItems;
- (NSString *) filename;
- (double) sqareOf:(double)number;

Methods Implementation

```
- (void) sayHello {
  // do something to say hello
}
```

Methods Implementation

```
- (BOOL) hasLoggedIn {
    if ( userSessionValid ) return YES;
    return NO;
}
```

Methods Implementation

```
- (double) sqareOf:(double)number {
    return number * number;
}
```

Property Declaration

Property Declaration

```
@property (nonatomic, retain) NSString *username;

@property (nonatomic, assign) NSString *username;

@property (nonatomic, copy) NSString *username;
```

Property Declaration

Interface:

```
@property (nonatomic, retain) NSString *username;
```

Implementation:

```
@synthesis username;
```

Property Declaration

We can use both following syntax to access the username.

self.username

[self username]

Property Declaration

We can use both following syntax to set the username to new value.

```
self.username = @”Steven”;
[self setUsername: @”Steven”];
```

Property Declaration

When

@synthesis username;

Both following works, but they act differently.

self.username = @”Thomas Mak”;

username = @”Thomas Mak”;

Property Declaration

self or not self?

Property Declaration

self.username = @"Thomas Mak";

equals to

[self setUsername:@"Thomas Mak"];

Property Declaration

```
[self setUsername:@”Thomas Mak”];
```

equals to

```
[username release]; // release old username
username = @”Thomas Mak”;
```

Property Declaration

self.username means calling the setting method.

The setter method:

```
- (void)setUsername:(NSString*)newUsername
{
    [username release];
    username = [newUsername retain];
}
```

Property Declaration

The issue of calling

username = @”new name”;

The old username variable is not released.

memory leaked.

Property Declaration

Interface:

```
@property (nonatomic, retain) NSString *username;
```

Implementation:

```
@synthesis username = _username;
```

Property Declaration

When

@synthesis username = _username;

✓ self.username = @”Thomas Mak”;

X username = @”Thomas Mak”;

Property Declaration

@synthesis username = _username;

```
- (void)setUsername:(NSString*)newUsername
{
    [_username release];
    _username = [newUsername retain];
}
```

Property Declaration

Exception

We are safe to not using **self.** on primitive.

@property (nonatomic) int gameScore;

No need to retain and release primitive.

Delegates

Delegates

- UITextField
- UITextFieldDelegates
- FBRequest
- FBRequestDelegates

View Cycle

View Cycle

- init
- viewDidLoad
- viewWillAppear:
- viewWillDisappear:
- viewDidUnload
- dealloc

Using Text View

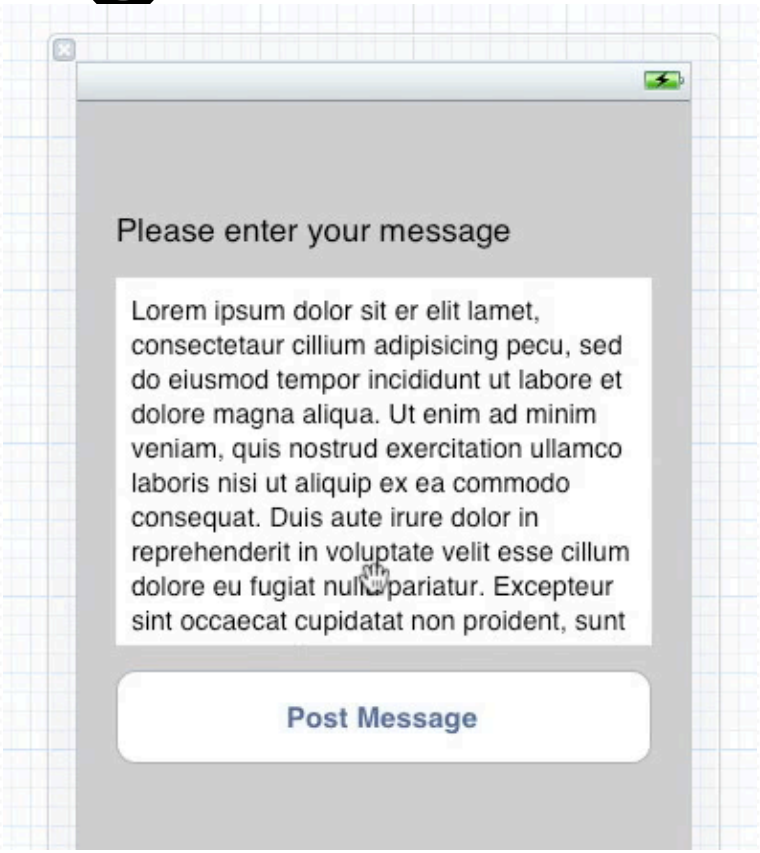
Using UITextView

- UITextViewDelegates Methods
- didStartEditing
- didEndEditing

Using UITextView

- Detect Return Key to end editing
- Move the view to show keyboard

Using UITextView



Prepare a view with UITextView

Connect the UITextView delegate to File's Owner

Using UITextView

```
1 @interface ViewController : UIViewController <UITextViewDelegate>
2
3 @end
```

(Optional) Add the delegates to header.

Using UITextView

```
1 - (BOOL)textView:(UITextView *)textView shouldChangeTextInRange:
(NSRange)range replacementText:(NSString *)text {
2
3     if([text isEqualToString:@"\n"]) {
4         [textView resignFirstResponder];
5         return NO;
6     }
7
8     return YES;
9 }
```

Detect the input character and find the line break.

Using UITextView

```
1  - (void)textViewDidBeginEditing:(UITextView *)textView
2  {
3      CGRect frame = self.view.frame;
4      frame.origin.y = -100;
5      self.view.frame = frame;
6  }
7
8  - (void)textViewDidEndEditing:(UITextView *)textView
9  {
10     CGRect frame = self.view.frame;
11     frame.origin.y = 0;
12     self.view.frame = frame;
13 }
```

Move up the view when the keyboard shows.

Revert the view when the keyboard hides.

Using UITextView

```
1  - (void)textViewDidBeginEditing:(UITextView *)textView
2  {
3      CGRect frame = self.view.frame;
4      frame.origin.y = -100;
5      [UIView animateWithDuration:.3 animations: ^{
6          self.view.frame = frame;
7      }];
8  }
9
10 - (void)textViewDidEndEditing:(UITextView *)textView
11 {
12     CGRect frame = self.view.frame;
13     frame.origin.y = 0;
14     [UIView animateWithDuration:.3 animations: ^{
15         self.view.frame = frame;
16     }];
17 }
```

Bonus, animate the view transition.

Using UITextView

```
1 [textview becomeFirstResponder];
```

(Optional) Focus on the textview by code.

Using Slider

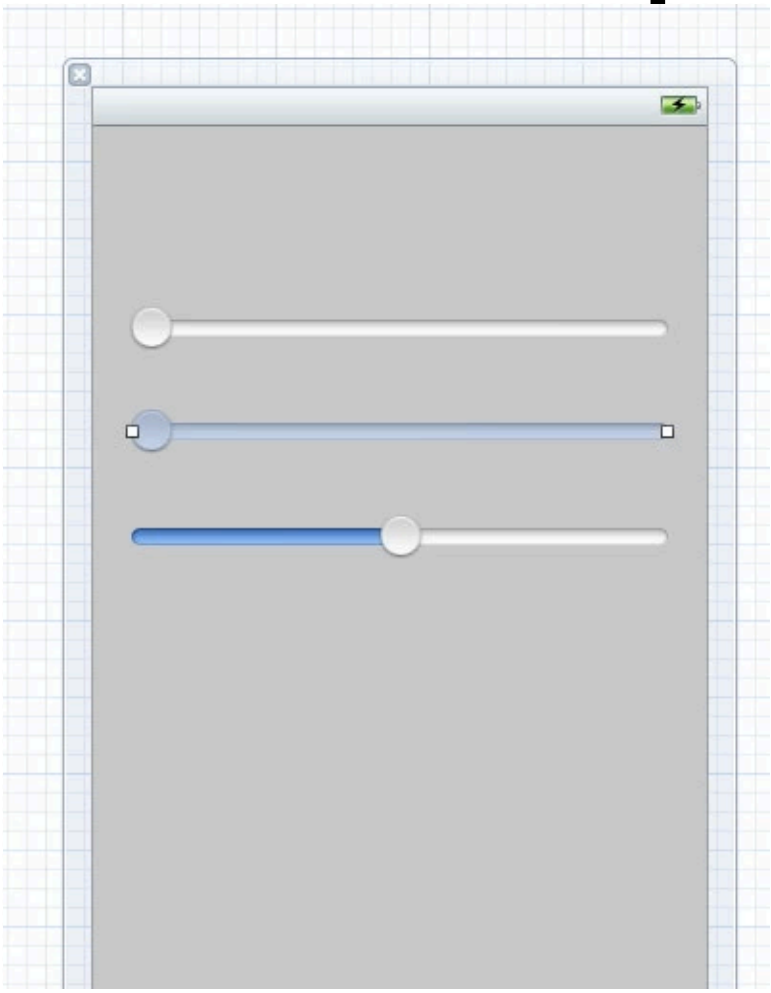
UISlider

- new IBAction
- valueChanged

UISlider Example

- RGB Color Picker
- RGB to Hex

RGB Example

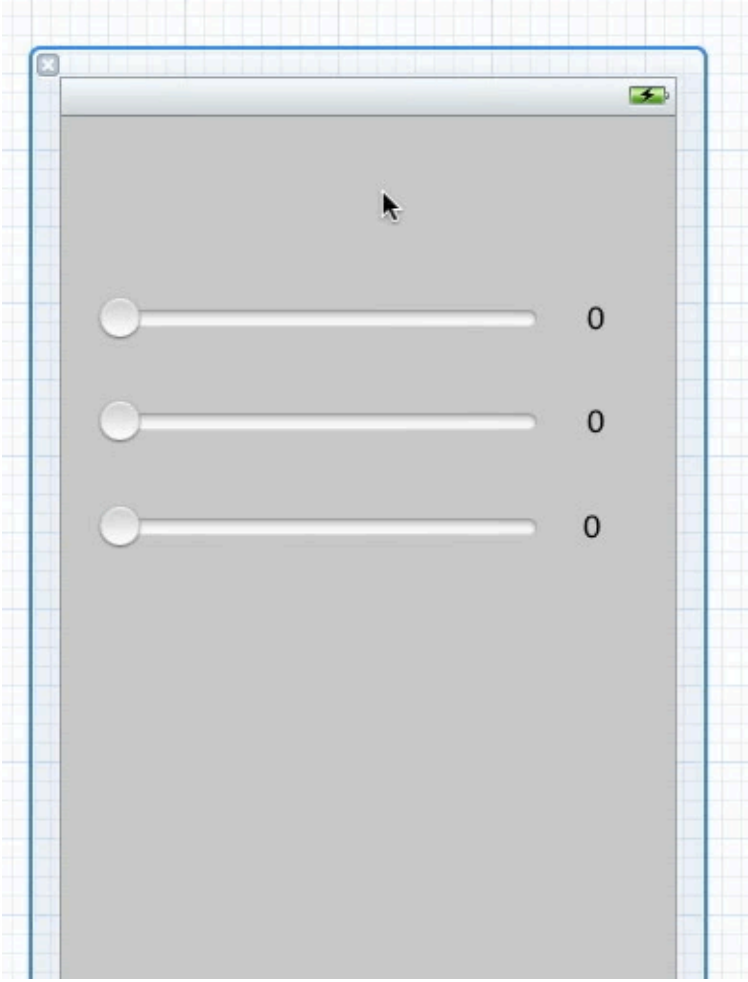


Put 3 UISliders on the view

RGB Example

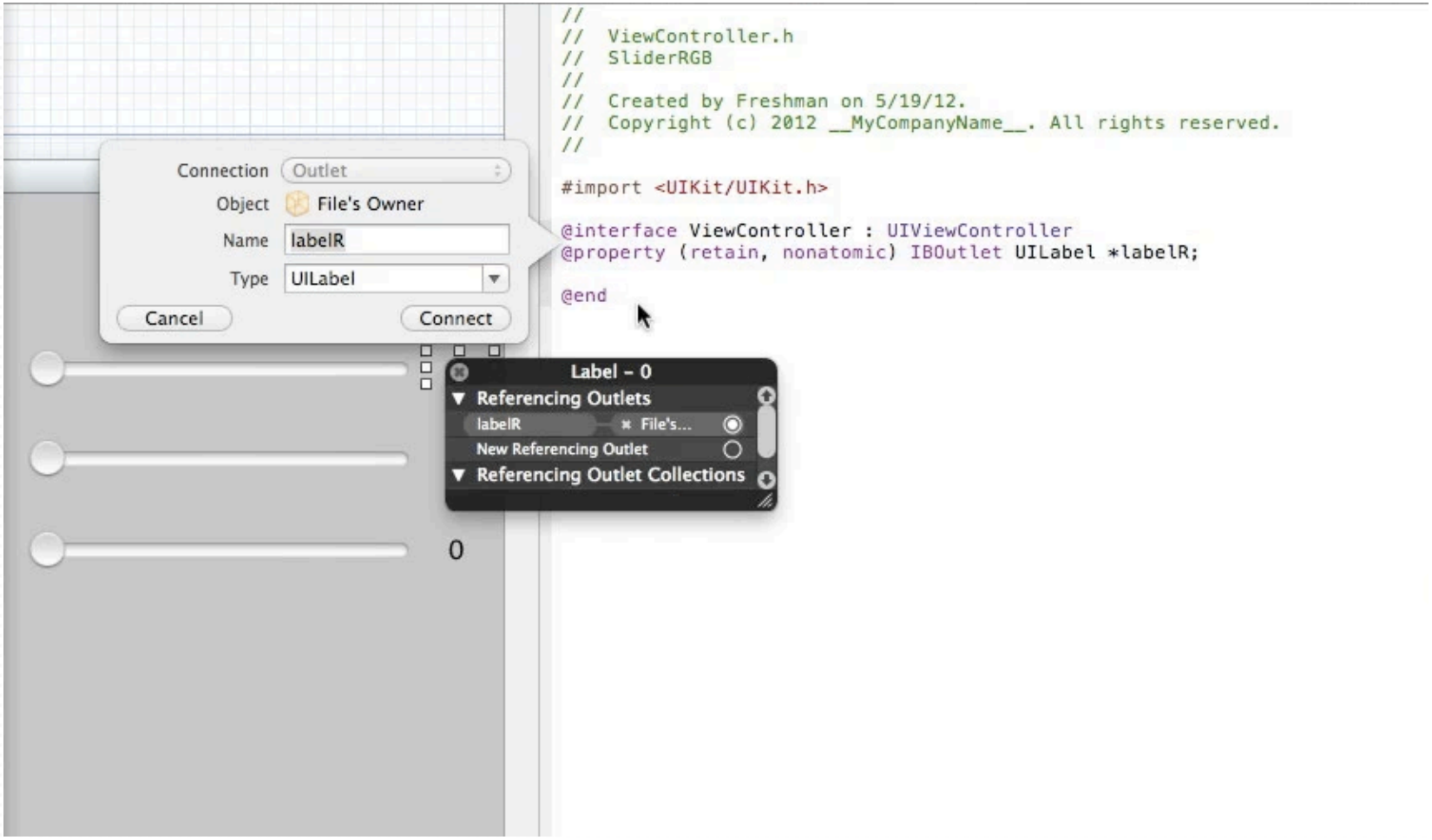
Set the slider range from 0 to 255

RGB Example



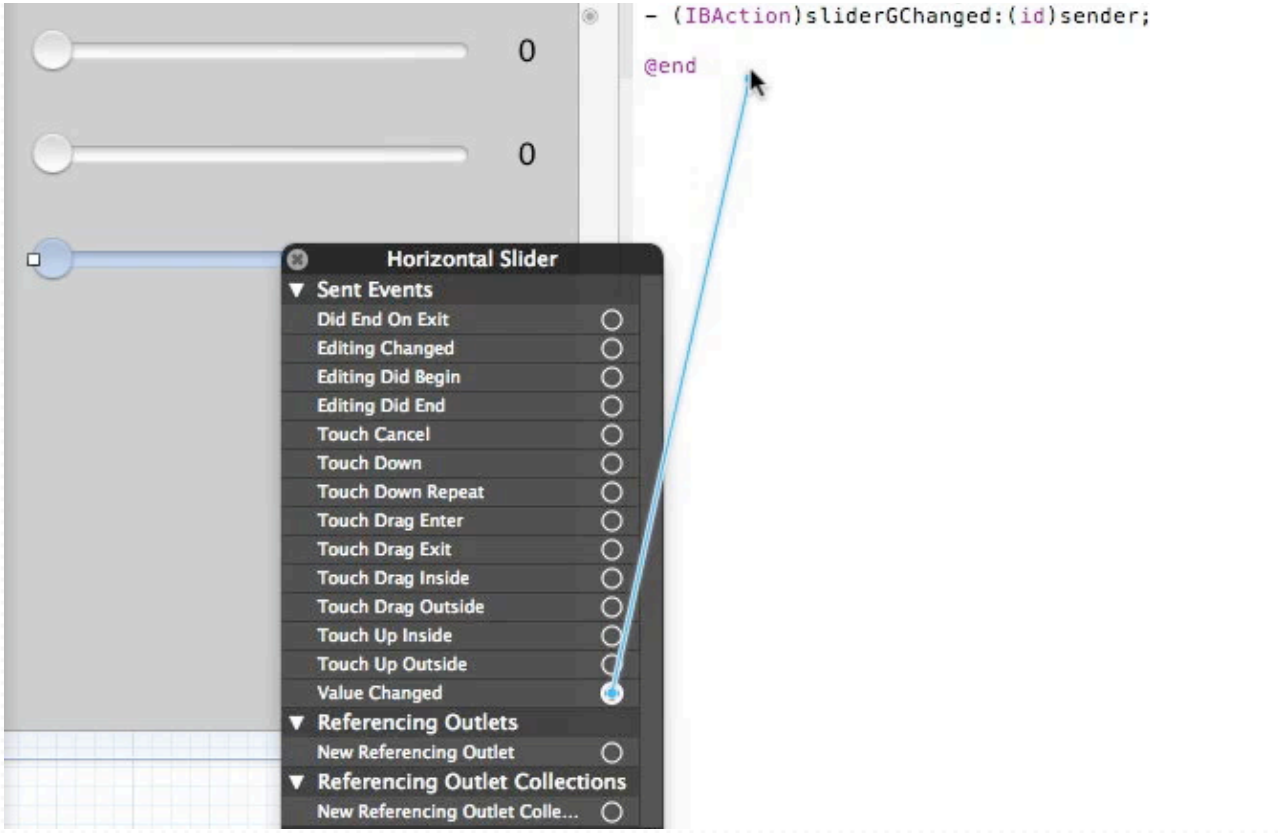
Add 3 UILabels besides the UISliders

RGB Example



Link the labels to be labelR, labelG and labelB

RGB Example



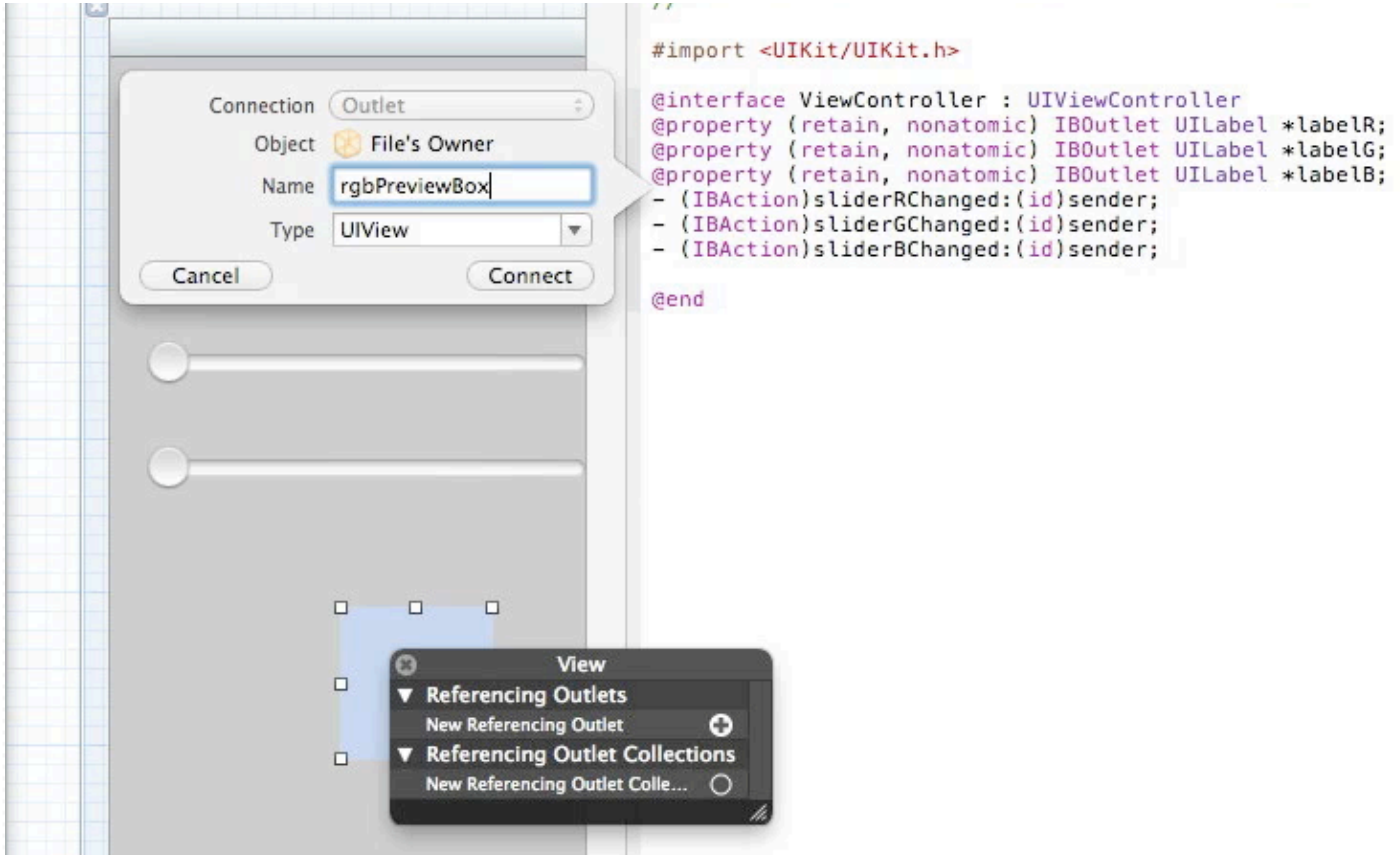
Link the Value Changed event of Slider to sliderRChanged, sliderGChanged, sliderBChanged

RGB Example

```
1  - (IBAction)sliderRChanged:(id)sender {
2      float r = [(UISlider*)sender value];
3      labelR.text = [NSString stringWithFormat:@"%.0f", r];
4  }
5
6  - (IBAction)sliderGChanged:(id)sender {
7      float g = [(UISlider*)sender value];
8      labelG.text = [NSString stringWithFormat:@"%.0f", g];
9  }
10
11 - (IBAction)sliderBChanged:(id)sender {
12     float b = [(UISlider*)sender value];
13     labelB.text = [NSString stringWithFormat:@"%.0f", b];
14 }
```

Change the label text when the slider changes

RGB Example




```
#import <UIKit/UIKit.h>

@interface ViewController : UIViewController
@property (retain, nonatomic) IBOutlet UILabel *labelR;
@property (retain, nonatomic) IBOutlet UILabel *labelG;
@property (retain, nonatomic) IBOutlet UILabel *labelB;
- (IBAction)sliderRChanged:(id)sender;
- (IBAction)sliderGChanged:(id)sender;
- (IBAction)sliderBChanged:(id)sender;

@end
```

Put an UIView in the view, connect it as **rgbPreviewBox**

RGB Example

```
1 @implementation ViewController {
2     float r;
3     float g;
4     float b;
5 }
```

Declare 3 variables to hold the RGB value

RGB Example

```
1 - (void)refreshPreviewBox
2 {
3     rgbPreviewBox.backgroundColor = [UIColor colorWithRed:r/255.0f
4                                                     green:g/255.0f
5                                                      blue:b/255.0f
6                                                     alpha:1];
7 }
```

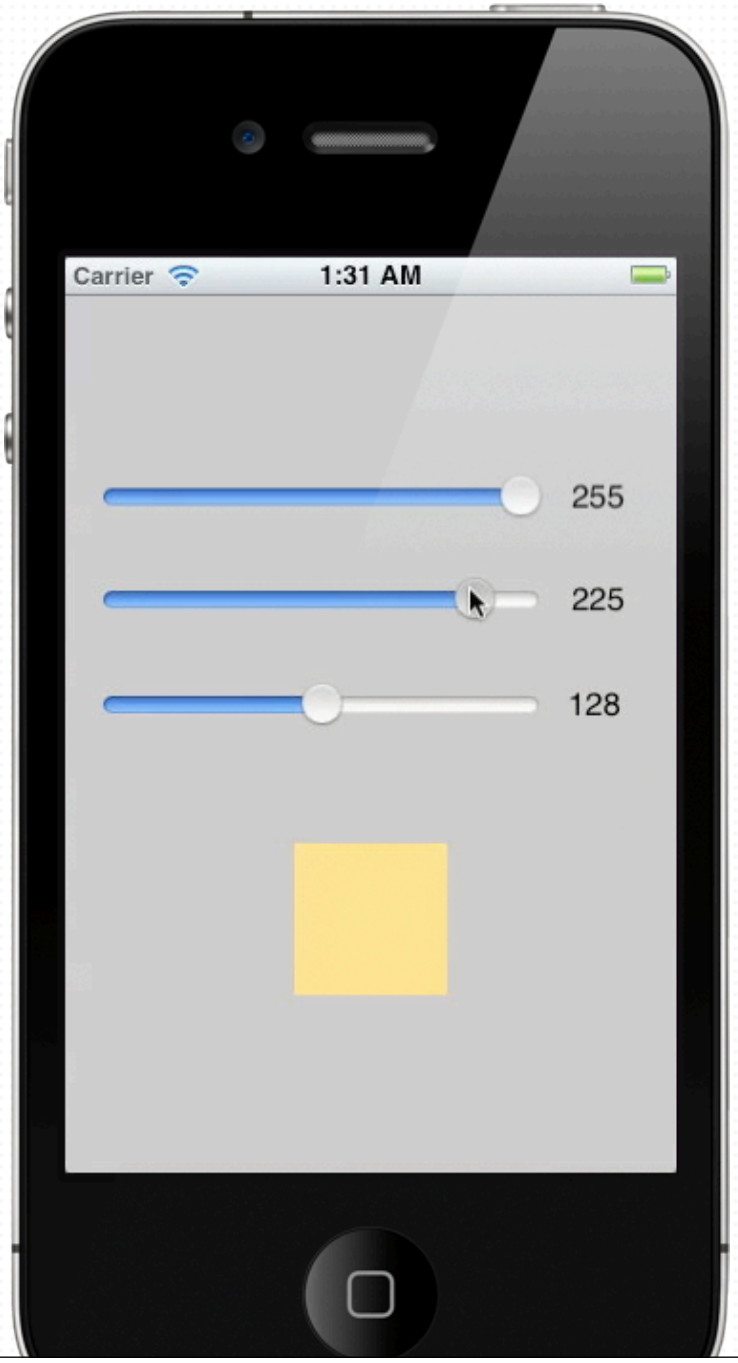
Create a method to change box color according to the stored RGB value.

RGB Example

```
1 - (IBAction)sliderRChanged:(id)sender {
2     r = [(UISlider*)sender value];
3     labelR.text = [NSString stringWithFormat:@"%.0f", r];
4     [self refreshPreviewBox];
5 }
6
7 - (IBAction)sliderGChanged:(id)sender {
8     g = [(UISlider*)sender value];
9     labelG.text = [NSString stringWithFormat:@"%.0f", g];
10    [self refreshPreviewBox];
11 }
12
13 - (IBAction)sliderBChanged:(id)sender {
14    b = [(UISlider*)sender value];
15    labelB.text = [NSString stringWithFormat:@"%.0f", b];
16    [self refreshPreviewBox];
17 }
```

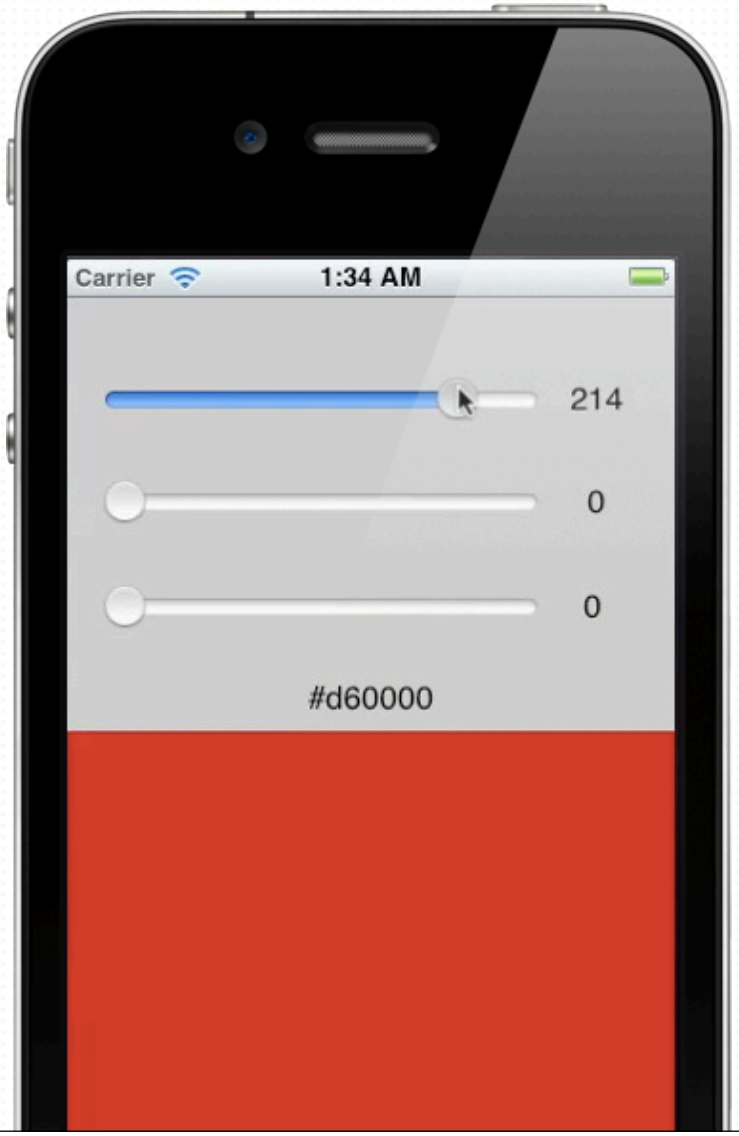
Update the changing event to refresh the color box

What we get so far



RGB Example

Add a **hexLabel** UILabel to the view.



RGB Example

```
1 - (void)refreshHexLabel
2 {
3     hexLabel.text = [NSString stringWithFormat:@"#%02X%02X%02X",(int)r,
(int)g, (int)b];
4 }
```

(Bonus) a method to show the RGB color in hex format

RGB Example

```
1 - (IBAction)sliderRChanged:(id)sender {
2     r = [(UISlider*)sender value];
3     labelR.text = [NSString stringWithFormat:@"%.0f", r];
4     [self refreshPreviewBox];
5     [self refreshHexLabel];
6 }
```

Add the code to refresh Hex label when slider changes.

RGB Example

```
 1 - (void)refreshHexLabel
 2 {
 3     hexLabel.text = [NSString stringWithFormat:@"#%02X%02X%02X",(int)r,
(int)g, (int)b];
 4
 5     // change the label color if it is difficult to see.
 6     if (r+g+b < 255 * 3 / 2)
 7     {
 8         hexLabel.textColor = [UIColor whiteColor];
 9     }
10     else
11     {
12         hexLabel.textColor = [UIColor blackColor];
13     }
14 }
```

(Bonus) change hex label color to make it clear.

RGB Example

Result:

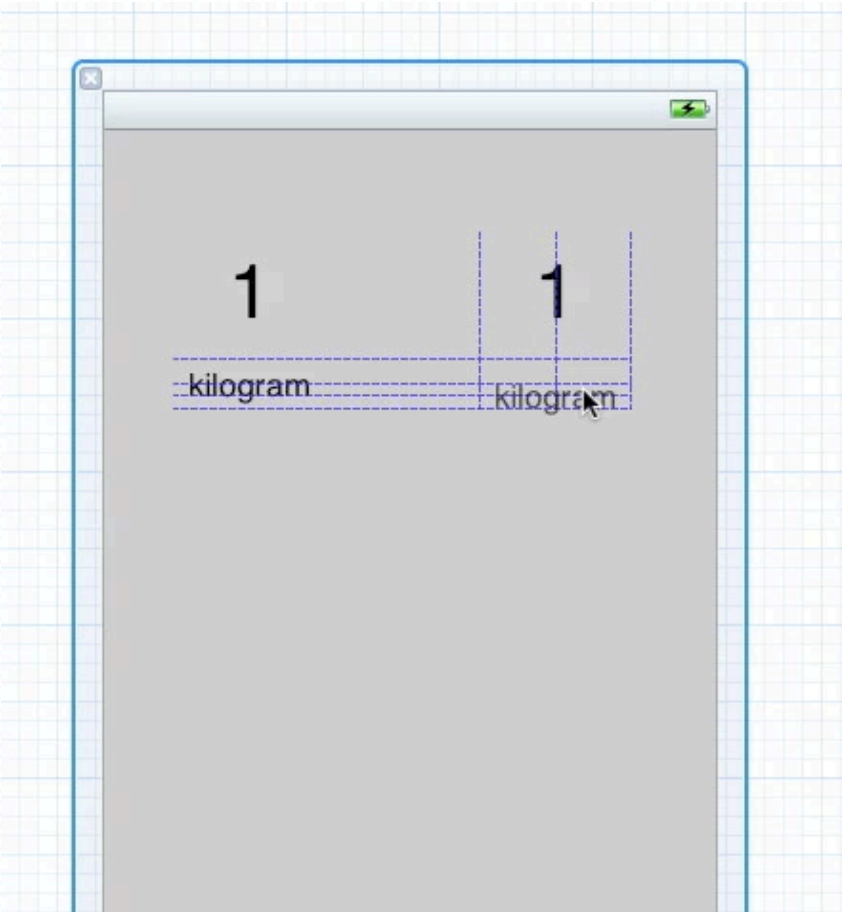
an RGB to hex converter
with UISlider

Using Picker

UIPicker Example

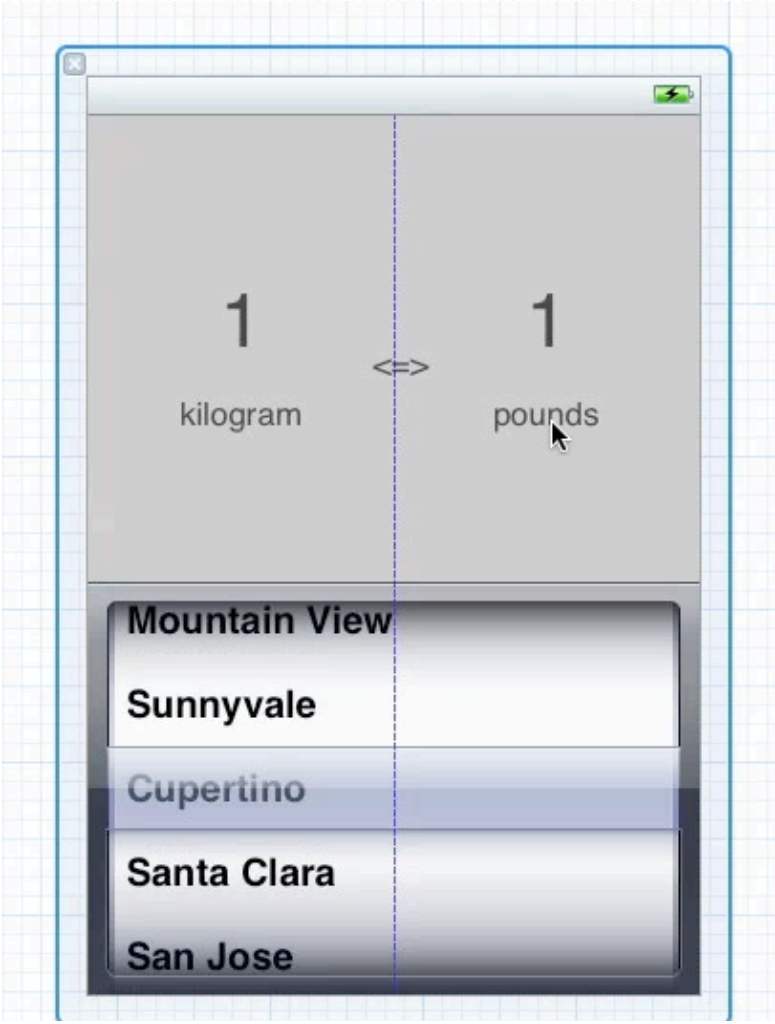
- Unit converter for Kilograms, Pounds, Ounces.

UIPicker Example



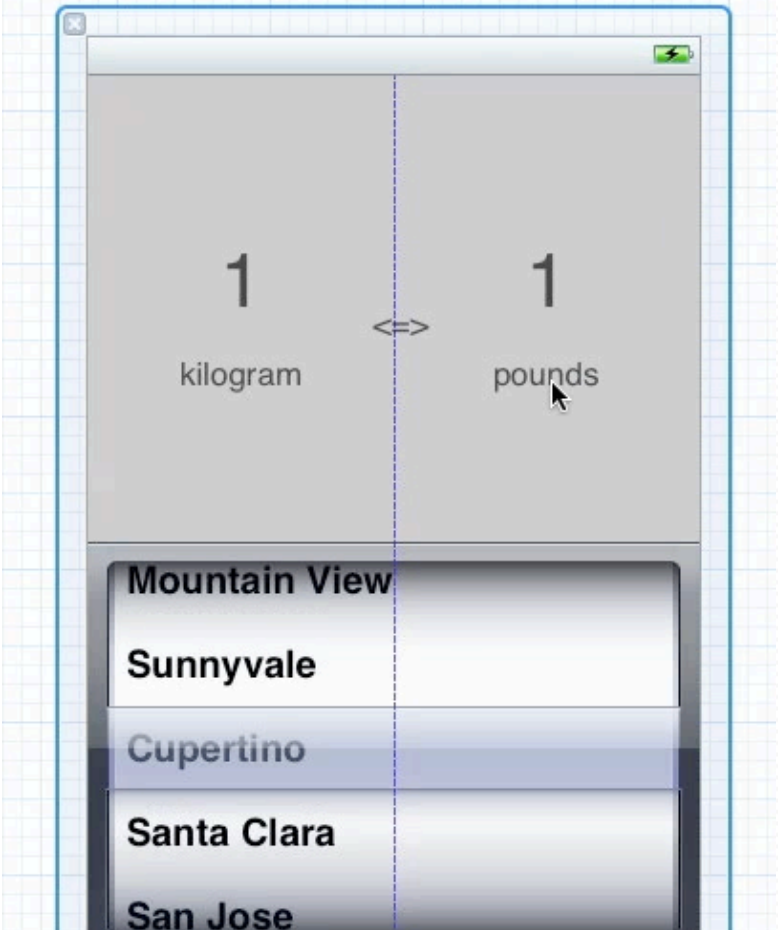
Prepare the UILabels, connect as *leftNumberLabel*, *leftUnitLabel*, *rightNumberLabel*, *rightUnitLabel*.

UIPicker Example



Drag a UIPickerView into the view.

UIPicker Example



Drag a UIPickerView into the view.

Connect the UIPickerView delegate and datasource to File's Owner

UIPicker

- How we can check delegates methods ?

1. Declare the delegates in header.
2. CMD + Click on the delegate name.
3. XCode jumps to the header file of delegate.
4. Check the available delegate methods and related comments.

Unit Converter

```
1 - (NSInteger)numberOfComponentsInPickerView:(UIPickerView *)pickerView
2 {
3     return 2;
4 }
```

Define how many components we split the picker.

Unit Converter

```
1 - (NSInteger)pickerView:(UIPickerView *)pickerView
numberOfRowsInComponent:(NSInteger)component
2 {
3    // both left and right picker component has the same amount of rows
4     return 3;
5 }
```

Define how many row for each picker component.

Unit Converter

```
1 - (NSString *)pickerView:(UIPickerView *)pickerView titleForRow:
(NSInteger)row forComponent:(NSInteger)component
2 {
3     // both left and right components share the same rows and text
4     // otherwise we need to distinguish them.
5     switch (row) {
6         case 0:
7             return @"kilograms";
8             break;
9         case 1:
10            return @"pounds";
11        case 2:
12            return @"ounces";
13        default:
14            break;
15    }
16    return @"";
17 }
```

Tell the picker what text we use for each row.

Unit Converter

```
    1 - (void)pickerView:(UIPickerView *)pickerView didSelectRow:(NSInteger)row inComponent:
(NSInteger)component {
    2     if (component == 0) {
    3         if (row == 0) {
    4             leftUnitLabel.text = @"kilograms";
    5         }
    6         else if (row == 1) {
    7             leftUnitLabel.text = @"pounds";
    8         }
    9         else if (row == 2) {
   10             leftUnitLabel.text = @"ounces";
   11         }
   12     }
...
   24     [self refreshNumbers];
   25 }
```

Change left and right label when we selected a row.

```
1 - (void)pickerView:(UIPickerView *)pickerView didSelectRow:(NSInteger)row
inComponent:(NSInteger)component {
2     if (component == 0) {
3         if (row == 0) {
4             leftUnitLabel.text = @"kilograms";
5         }
6         else if (row == 1) {
7             leftUnitLabel.text = @"pounds";
8         }
9         else if (row == 2) {
10            leftUnitLabel.text = @"ounces";
11        }
12    }
13    else if (component == 1) {
14        if (row == 0) {
15            rightUnitLabel.text = @"kilograms";
16        }
17        else if (row == 1) {
18            rightUnitLabel.text = @"pounds";
19        }
20        else if (row == 2) {
21            rightUnitLabel.text = @"ounces";
22        }
23    }
24    [self refreshNumbers];
25 }
```

The did select delegate, full code.

Unit Converter

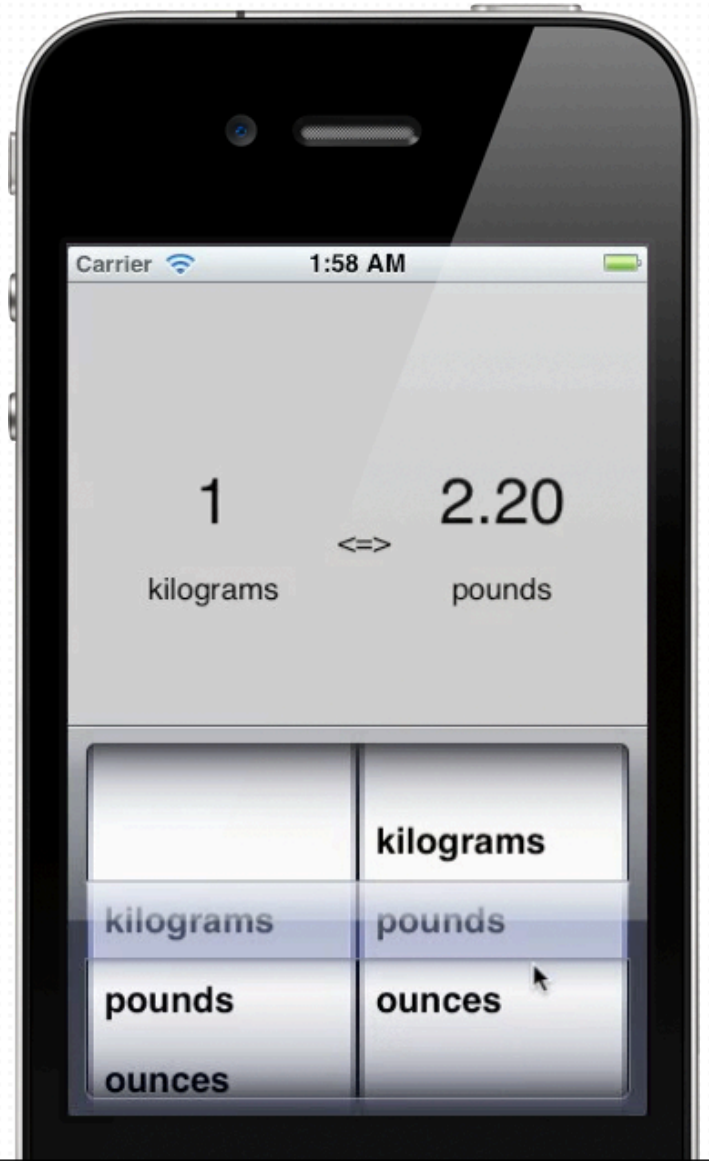
```
1 - (void)refreshNumbers {
2     if ([leftUnitLabel.text isEqualToString:@"kilograms"] &&
[rightUnitLabel.text isEqualToString:@"kilograms"]) {
3         rightNumberLabel.text = @"1";
4     }
5     else if ([leftUnitLabel.text isEqualToString:@"kilograms"] &&
[rightUnitLabel.text isEqualToString:@"pounds"]) {
6         rightNumberLabel.text = @"2.20";
7     }
8     else if ([leftUnitLabel.text isEqualToString:@"kilograms"] &&
[rightUnitLabel.text isEqualToString:@"ounces"]) {
9         rightNumberLabel.text = @"35.27";
10    }
11    ...
12 }
```

The conversion part, show correct number according to the selected left and right picker component.

Unit Converter

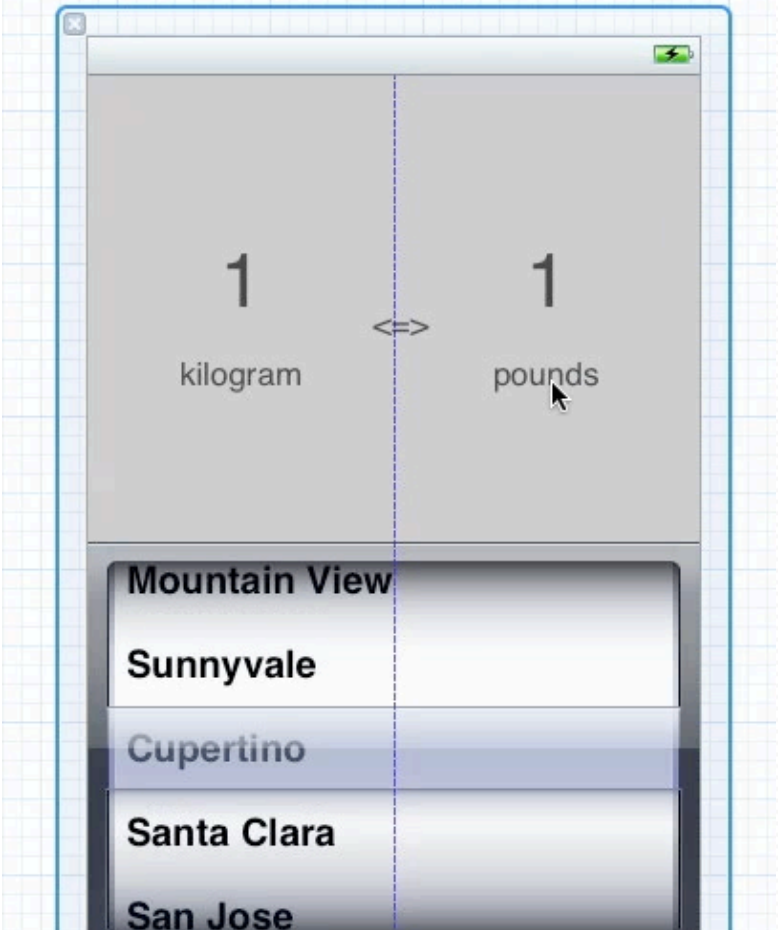
Result:

a basic unit converter



Using Image View

UIPicker Example

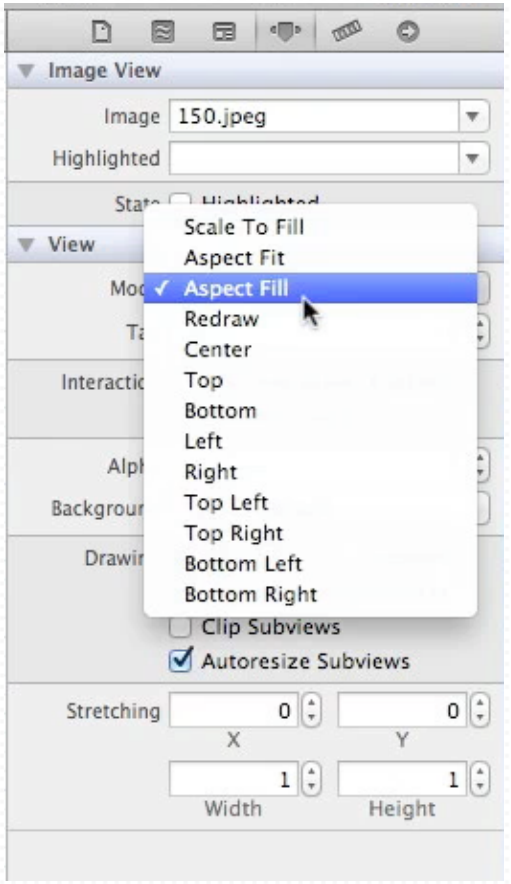


Drag a UIPickerView into the view.

Connect the UIPickerView delegate and datasource to File's Owner

Using UIImageView

Scale mode to define how the image fits into the bounds.



Using UIImageView

- [UIImage imageNamed:]
- (Will talk about how to load network image in next lesson)

Exercise

- ✓ Can you further develop the color / unit convertor to fit your usage?
- ✓ Present it to the class in next lesson.