iPhone App Development

CM420-09-2016-C

Lesson 4

Lecturer

Bryant Tang

bryant.tang14mo@gmail.com

CPTTMLAB_B
pw: cpttm1234

Git

https://github.com/bryanttang/iOS-Class-2016-9

Summary

- Animation
- UI Constraint
- Delegates (Review)
- Using TextView
- DataSource
- Using Picker

Animation

- Animatable UIView properties
  - Frame
  - bounds
  - center
  - transform
  - alpha
  - backgroundColor
  - contentStretch

Animation

- animateWithDuration:animations:
- animateWithDuration:animations:completion:
- animateWithDuration:delay:options:animations:completion:

Animation

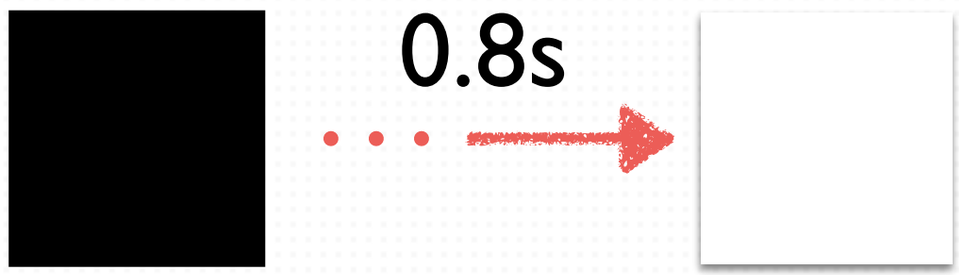
- animateWithDuration:animations:

```
//
// State A
   aview.backgroundColor = [UIColor blackColor];

[UIView animateWithDuration:0.8 animations:^{

   //State B
   aview.backgroundColor = [UIColor whiteColor];

}];
```



Animation

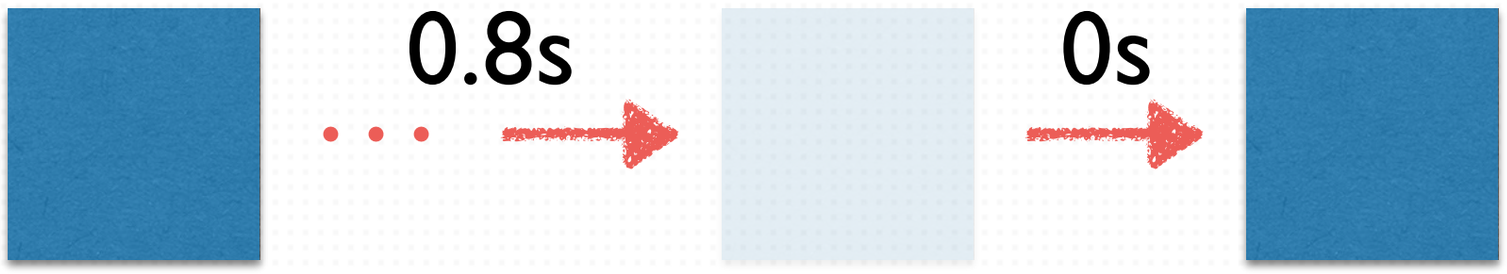
- animateWithDuration:animations:completion:

```
// State A
   aview.alpha = 1.0;

[UIView animateWithDuration:0.8 animations:^{
         // State B
          aview.alpha = 0.3;

    } completion:^(BOOL finished) {
             // State C
             aview.alpha = 1.0;

    }];
```



Animation

- animateWithDuration:delay:options:animations:completion:

```
[UIView animateWithDuration:0.8 animations:^{
        //state 1
    } completion:^(BOOL finished) {
        [UIView animateWithDuration:0.5
                              delay:2
                            options:UIViewAnimationOptionCurveEaseOut

                         animations:^{
                                //state 2
                            }
                         completion:^(BOOL finished) {
                                //state 3
         }];
    }];
```

Animation

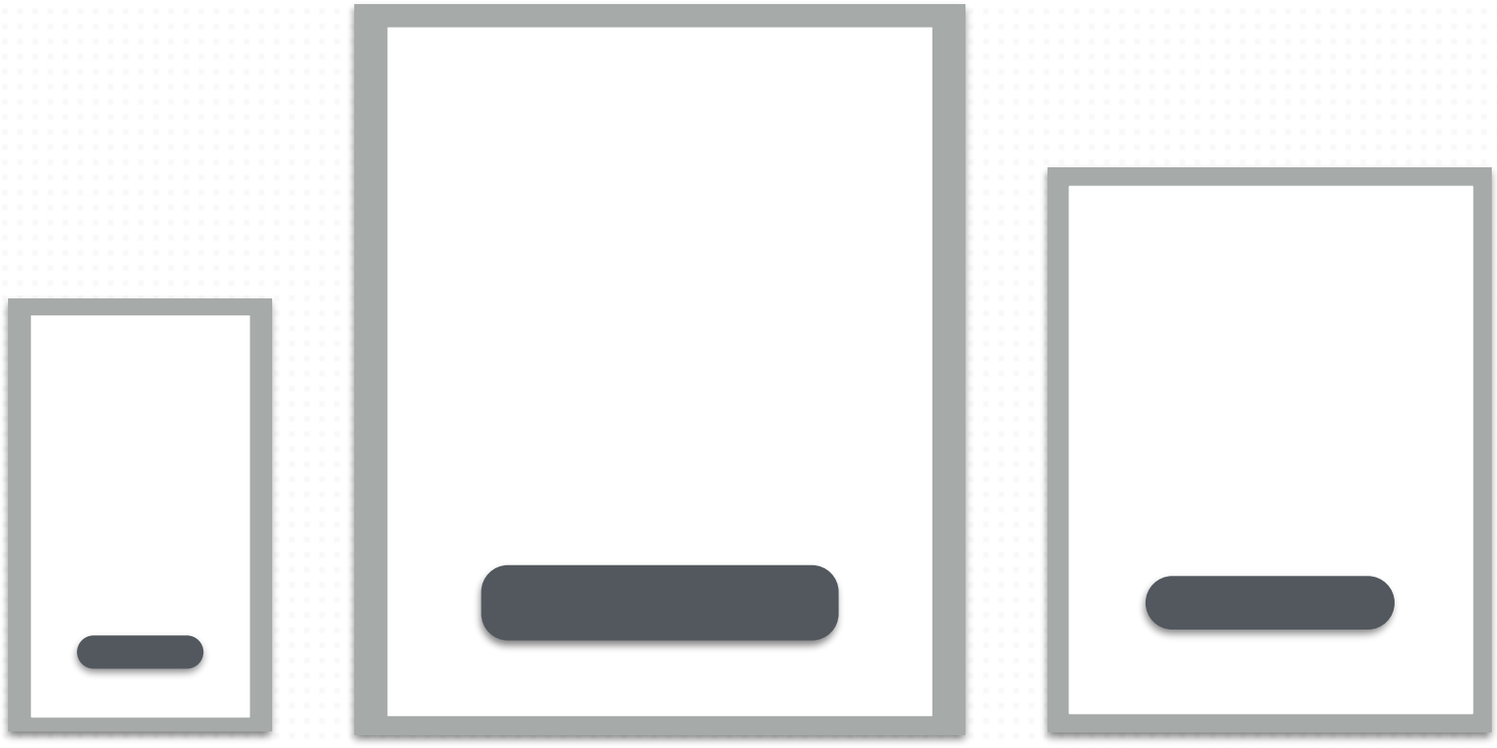
- UIViewAnimationOptions
  - UIViewAnimationOptionRepeat
  - UIViewAnimationOptionAutoreverse
  - UIViewAnimationOptionCurveEaseInOut
  - UIViewAnimationOptionCurveEaseOut
  - UIViewAnimationOptionTransitionFlipFromLeft
  - UIViewAnimationOptionTransitionCurlUp

UI Constraint

UI Constraint

Why?

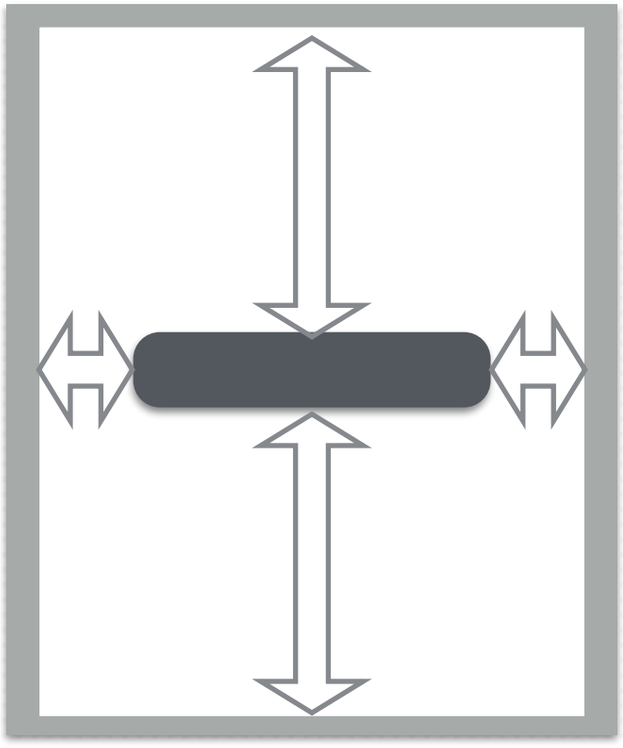
Ans: To be responsive to different size of device



UI Constraint

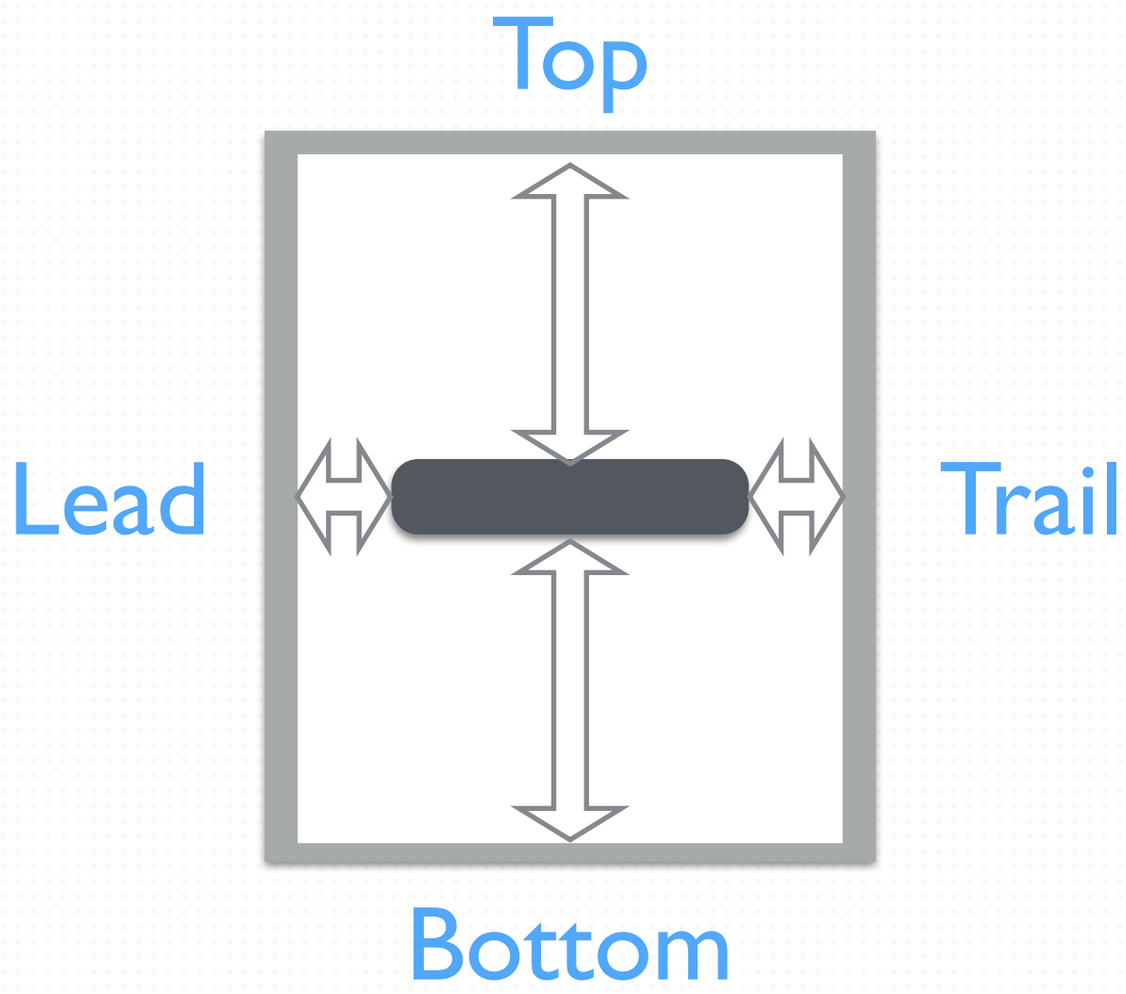
How to implement in StoryBoard?

<< Constraint >>



UI Constraint

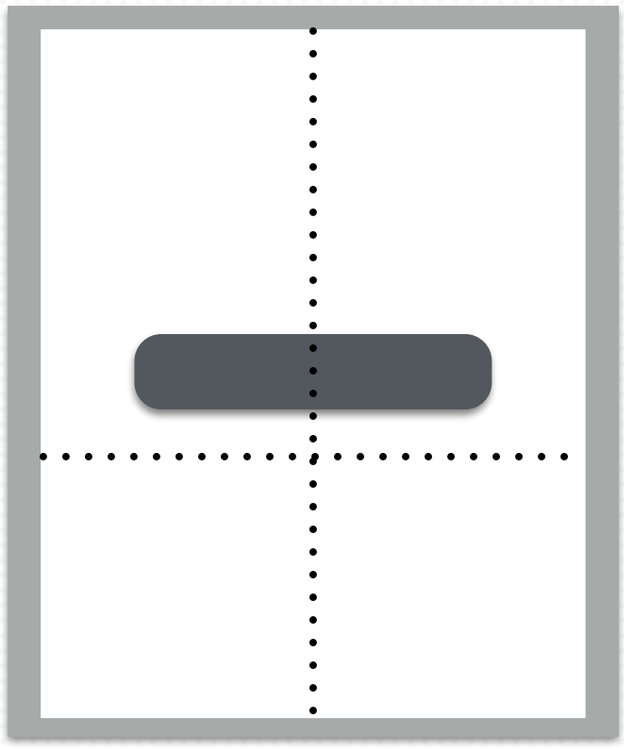
Constraint : Position



UI Constraint

Constraint : Position

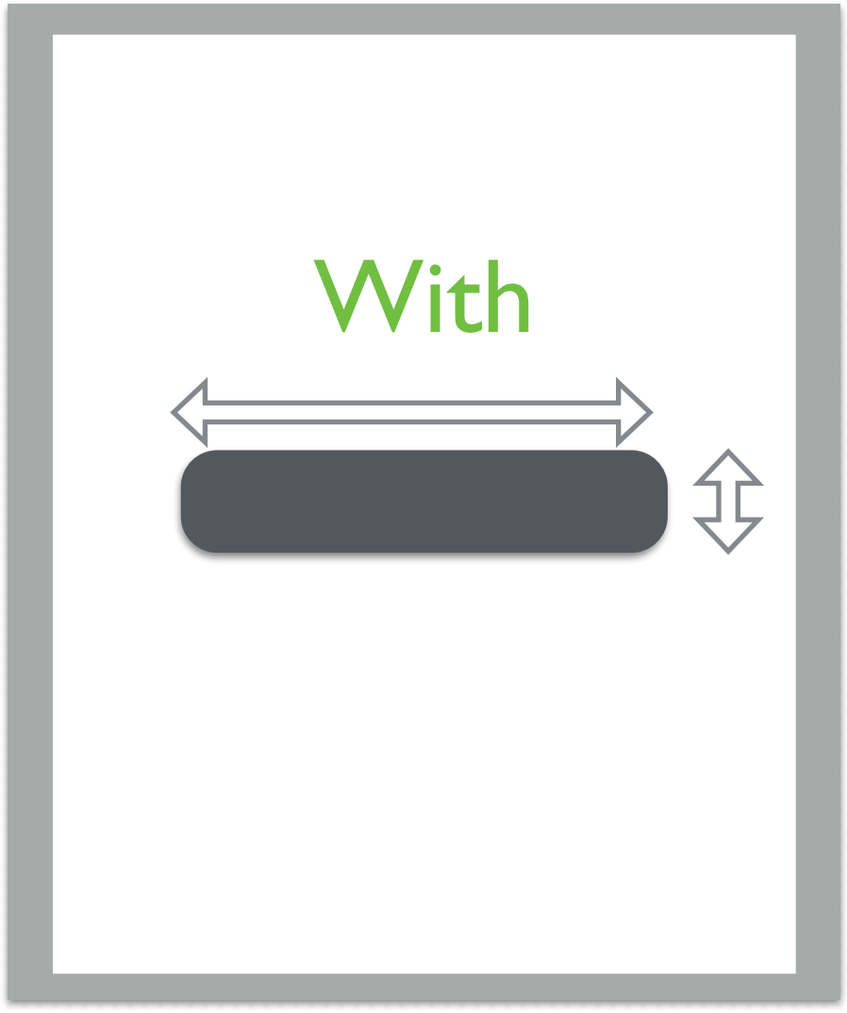
Horizontal



Vertical

UI Constraint

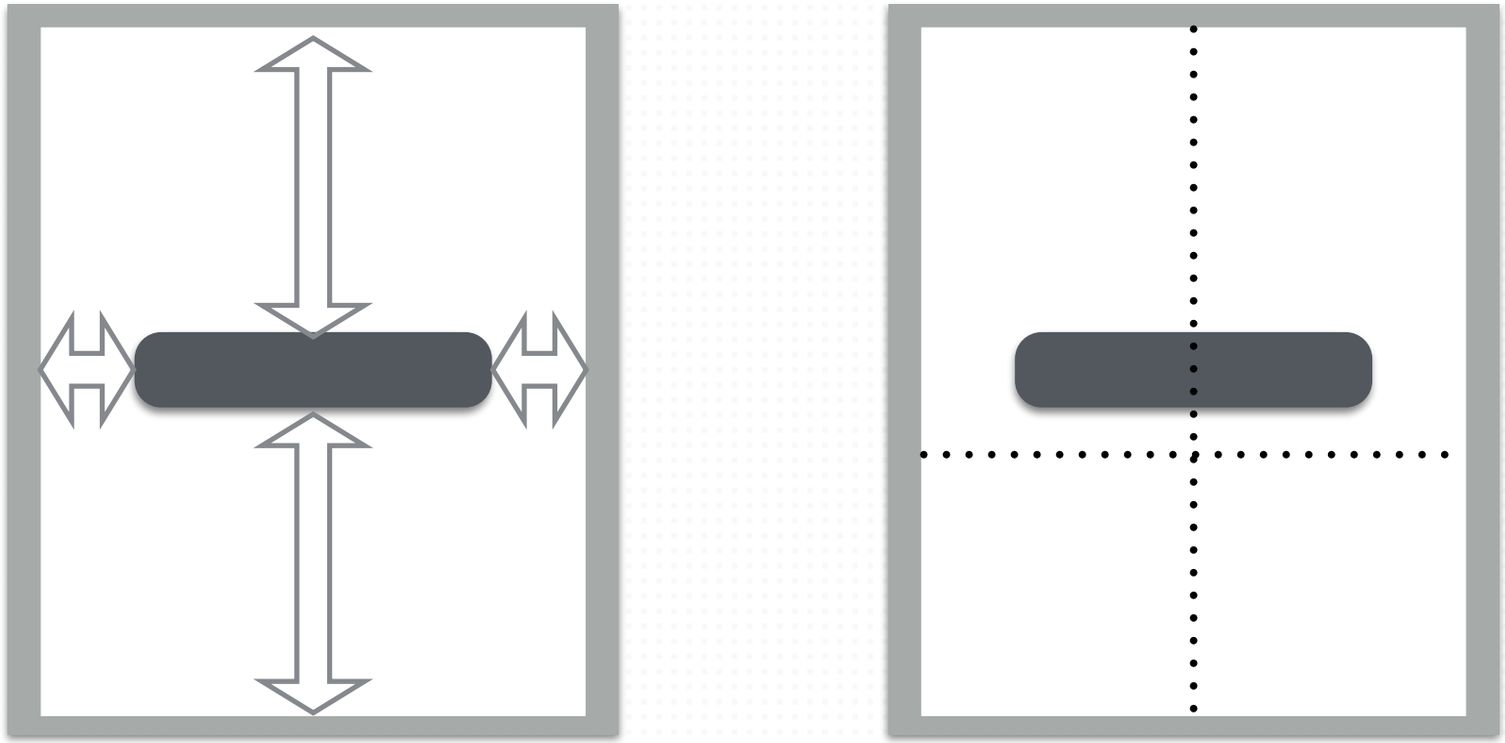
Constraint : Size



UI Constraint

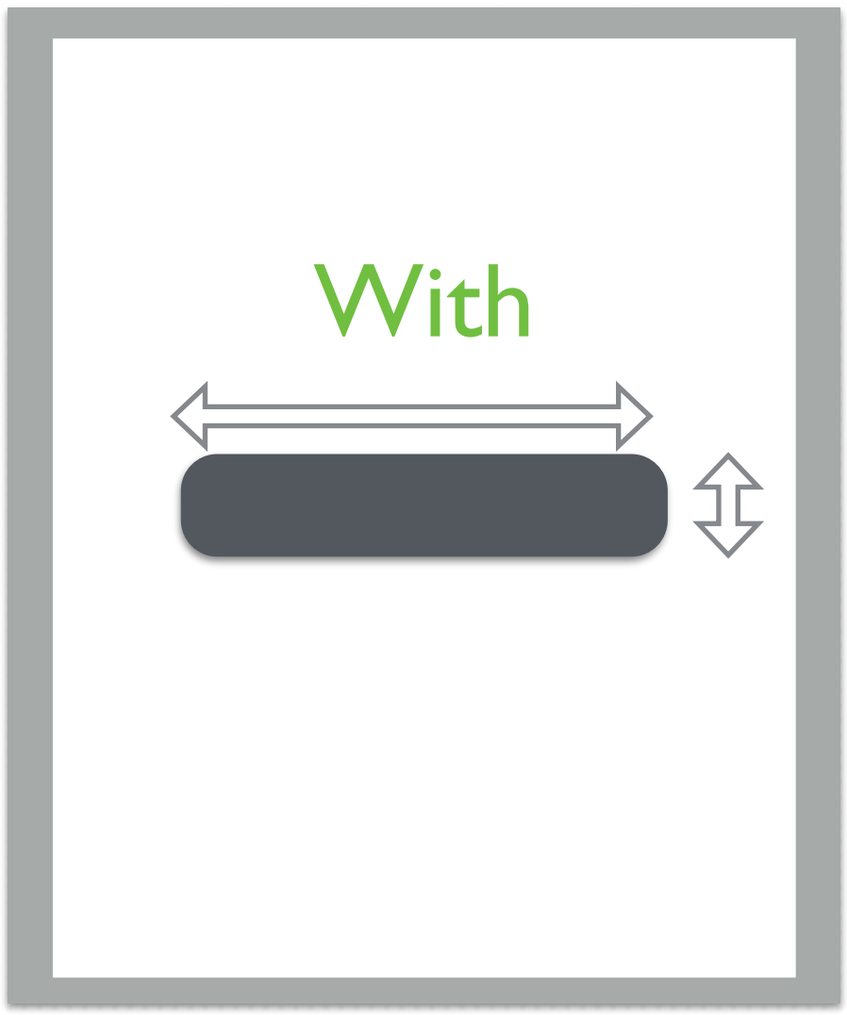
Two Objects involve

View Button



UI Constraint

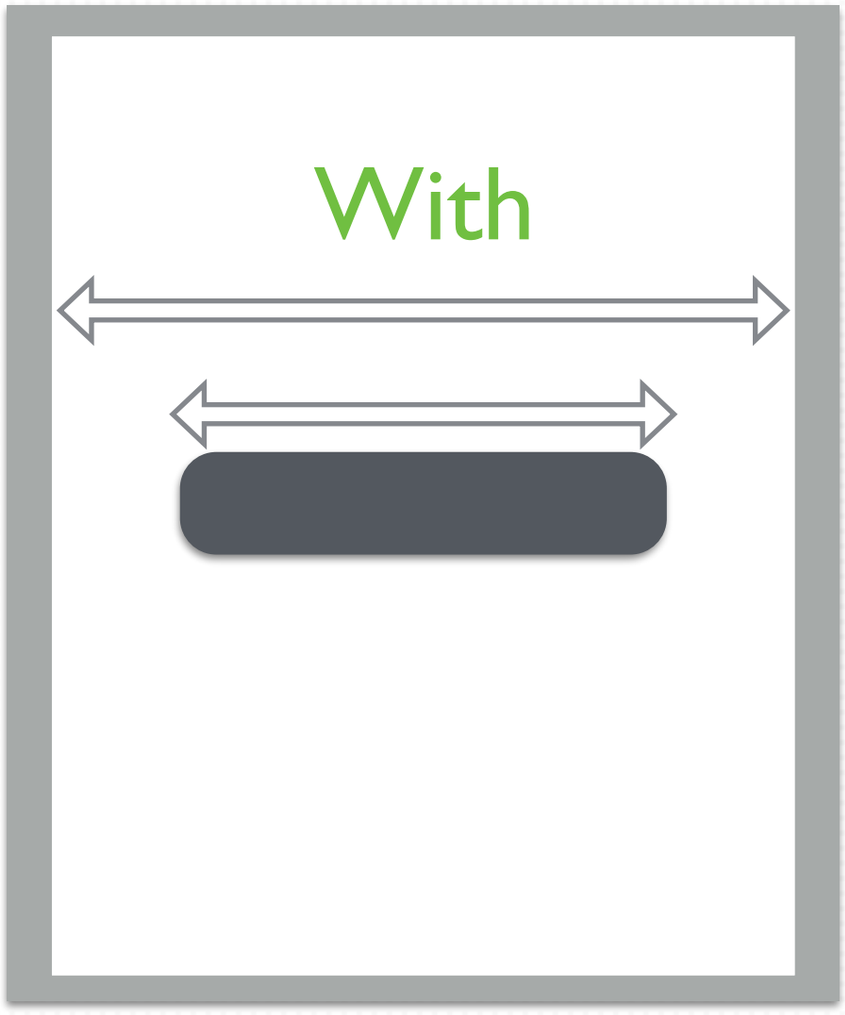
One Object involve?



UI Constraint

Two Objects involve? **Yes**

View Button



View's width : Button's width
2 : 1

Delegates

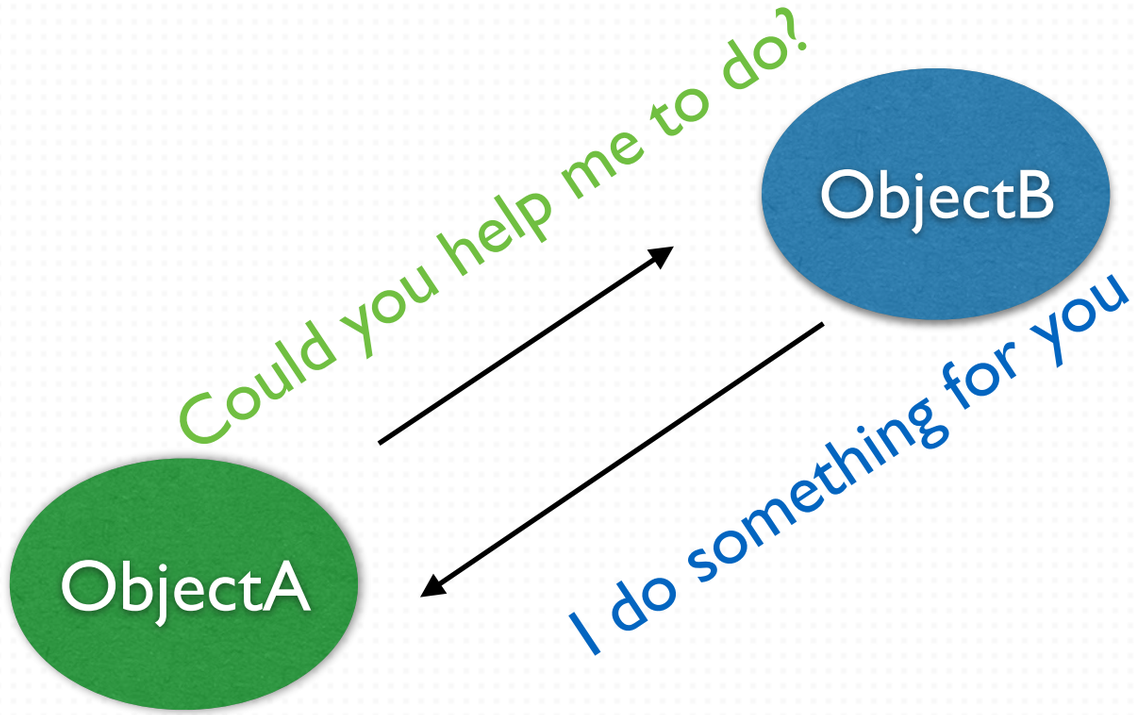
Delegates

Concept:

To help an object complete certain task

Example:

- UITextField
- UITextFieldDelegates
- UIPickerViewDelegates
- UITableViewDelegates



ObjectA.delegate = ObjectB

Using Text View

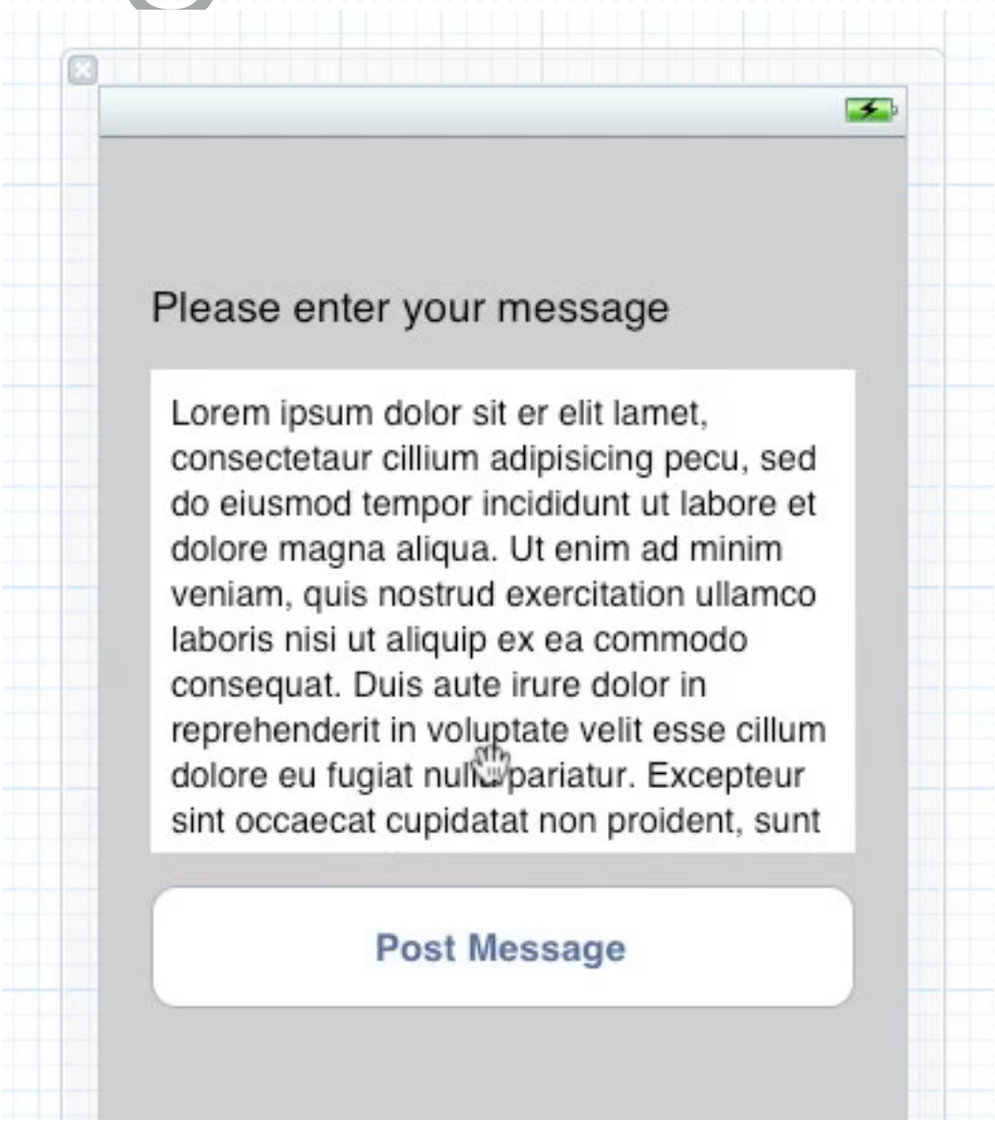
Using UITextView

- 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];
```

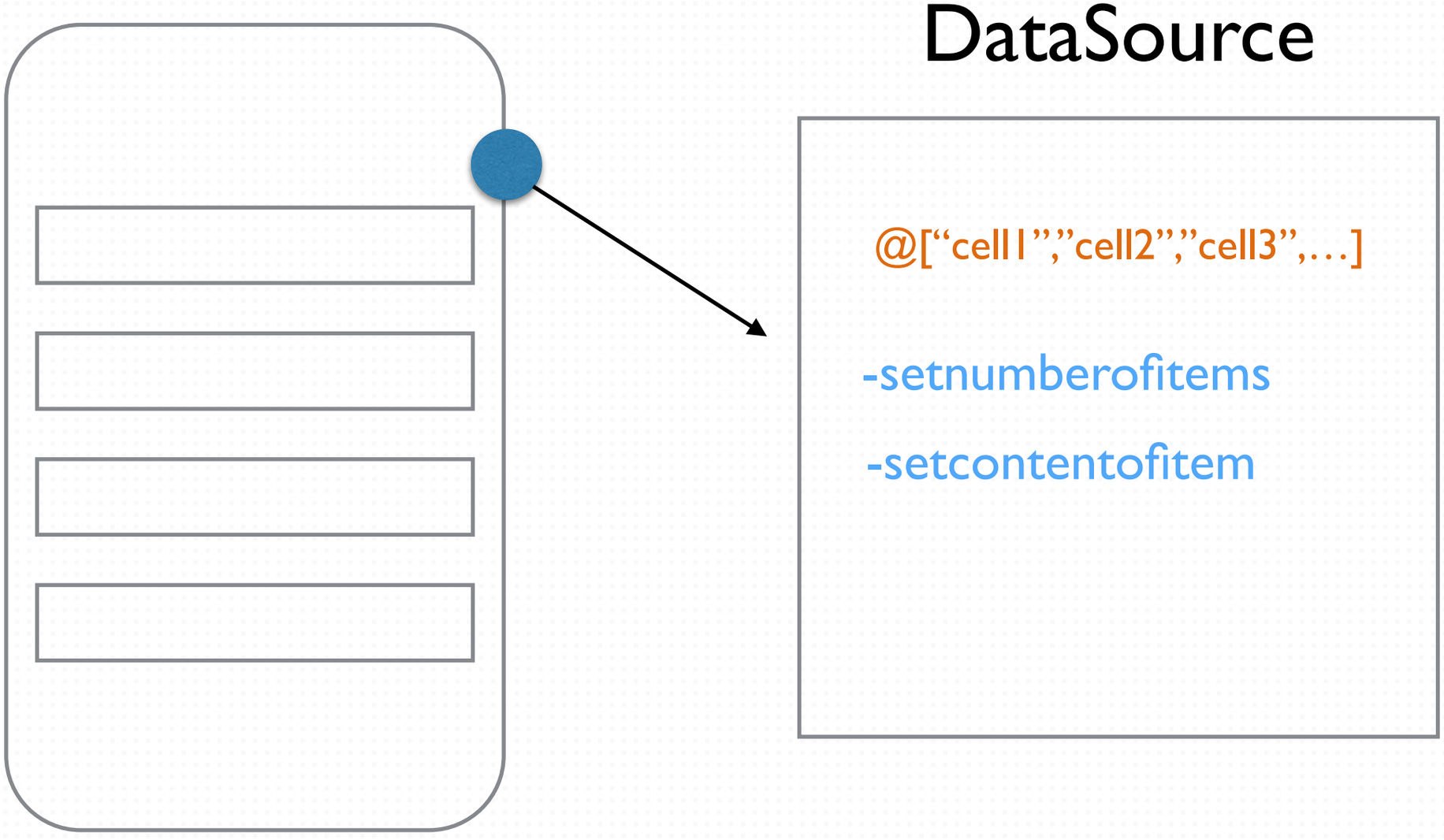
(Optional) Focus on the textview by code.

DataSource

DataSource

- Provide dataset for Collection UI
- It must be a Controller
- Keep maintain the data of Collection UI

DataSource

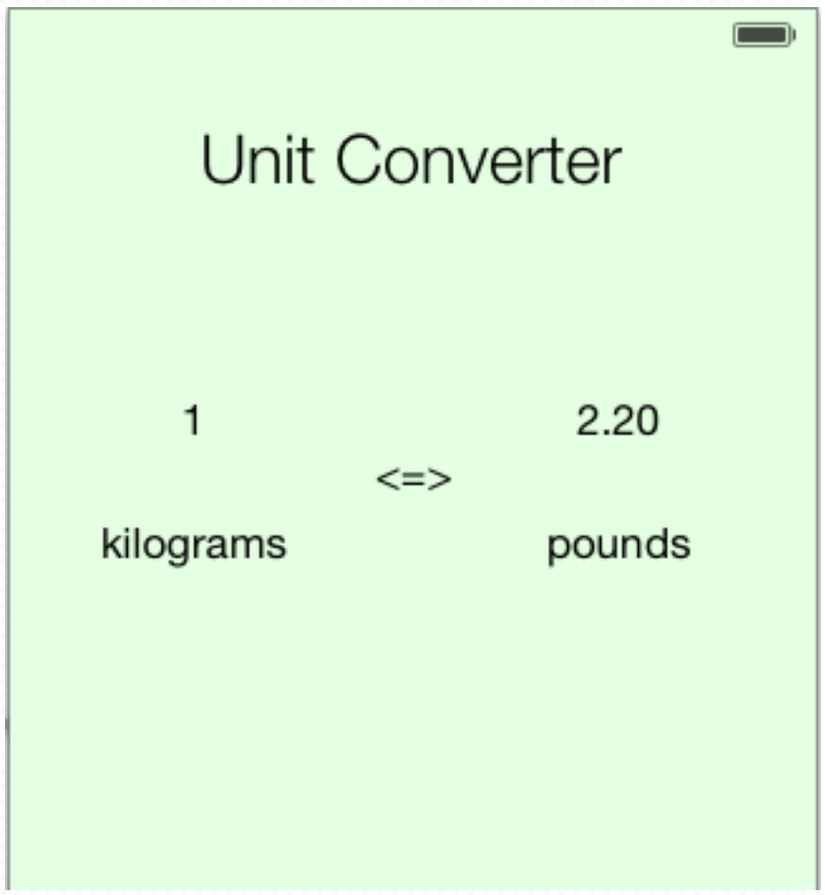

Collection UI
DataSource
@[“cell1”,”cell2”,”cell3”,…]
-setnumberofitems
-setcontentofitem

Using Picker

UIPicker Example

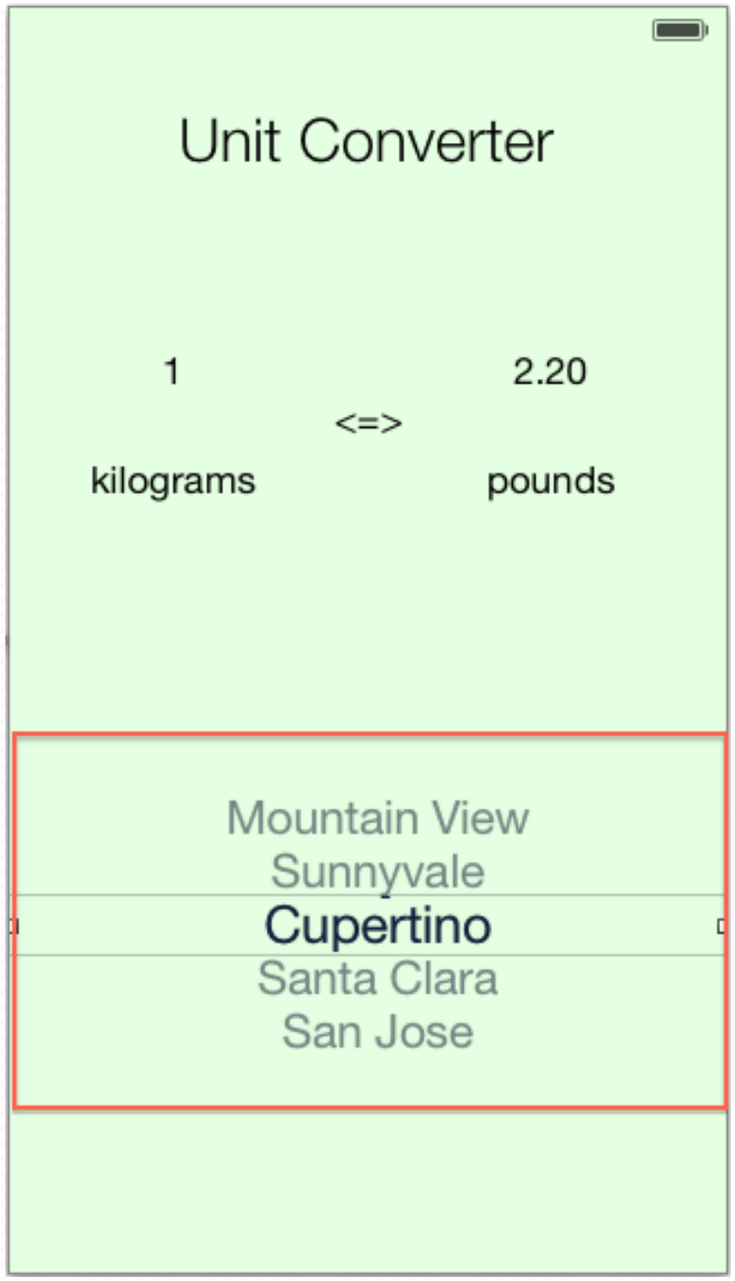
- Unit converter for Kilograms, Pounds, Ounces.

UIPicker Example



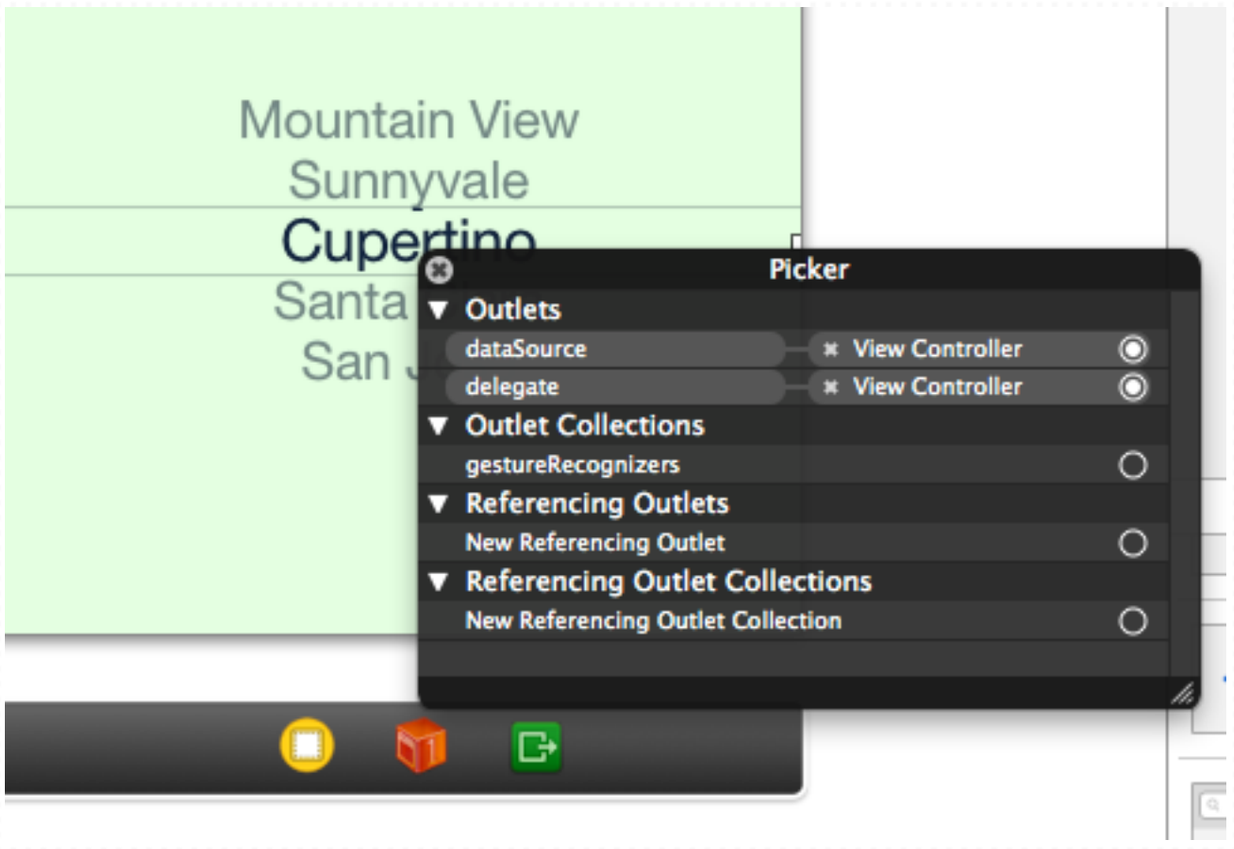
Prepare the UILabels, connect as *leftNumberLabel*, *leftUnitLabel*, *rightNumberLabel*, *rightUnitLabel*.

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. Command + 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             self.leftUnitLabel.text = @"kilograms";
5         }
6         else if (row == 1) {
7             self.leftUnitLabel.text = @"pounds";
8         }
9         else if (row == 2) {
10            self.leftUnitLabel.text = @"ounces";
11        }
12    }
...
24    [self refreshNumbers];
25 }
```

Change left and right label when we selected a row.

(Update: add '`self.`' before `leftUnitLabel`)

```
- (void)pickerView:(UIPickerView *)pickerView didSelectRow:(NSInteger)row
inComponent:(NSInteger)component {
    if (component == 0) {
        if (row == 0) {
            self.leftUnitLabel.text = @"kilograms";
        }
        else if (row == 1) {
            self.leftUnitLabel.text = @"pounds";
        }
        else if (row == 2) {
            self.leftUnitLabel.text = @"ounces";
        }
    }
    else if (component == 1) {
        if (row == 0) {
            self.rightUnitLabel.text = @"kilograms";
        }
        else if (row == 1) {
            self.rightUnitLabel.text = @"pounds";
        }
        else if (row == 2) {
            self.rightUnitLabel.text = @"ounces";
        }
    }
    [self refreshNumbers];
}
```

The did select delegate, full code.

Unit Converter

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

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

Exercise

- ✓ Can you further develop the unit convertor to fit your usage?