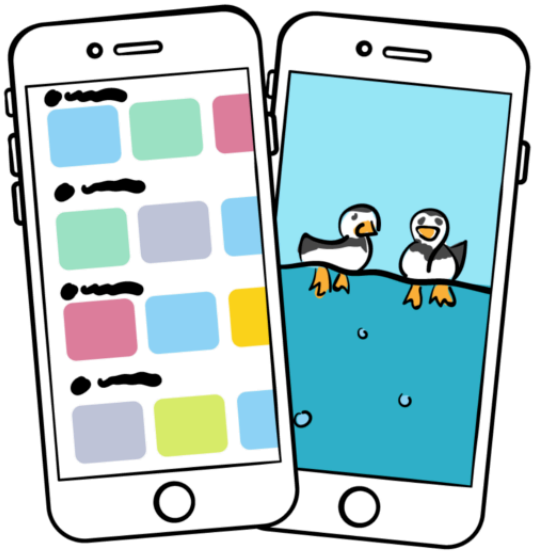
BEGINNING

COLLECTION VIEWS



HANDS-ON CHALLENGES

Beginning Collection Views

Michael Briscoe

Notice of Rights

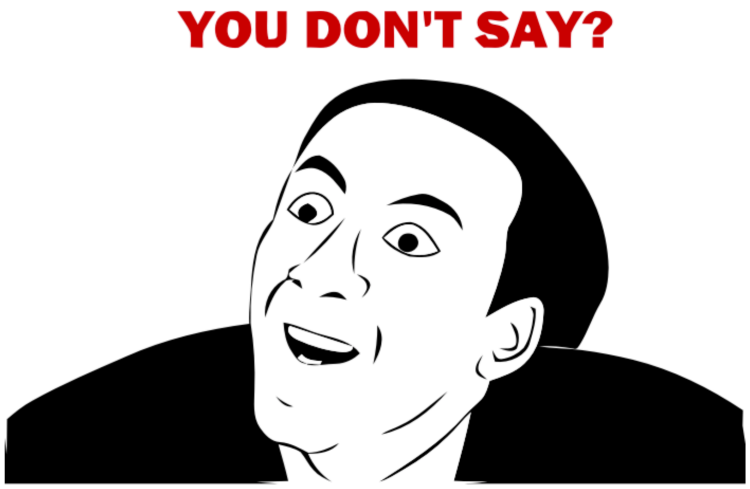
Notice of Liability

Trademarks

Challenge #4: Adding a State Icon

By Michael Briscoe

In the demo you learned how to create a custom section header view that displayed the title of the given section by subclassing `UICollectionReusableView`. It definitely serves its purpose, but is a touch on the bland side.



What you need to do is inject a little color!

If you open **Images.xcassets** you'll notice five images representing the state flags for each of the National Parks. Your challenge this time is to display the appropriate flag image in each section header.

> **Hint:** You'll need to add an image view to your storyboard.
>
> The images also share their names with the section titles supplied by the data source, and you're already retrieving the title for a given section in `collectionView(_:viewForSupplementaryElementOfKind:at:)`. Maybe you could use that?

Solution

Open **SectionHeaderView.swift** from the **Views** group, and add the following `IBOutlet` property declaration just below the existing one:

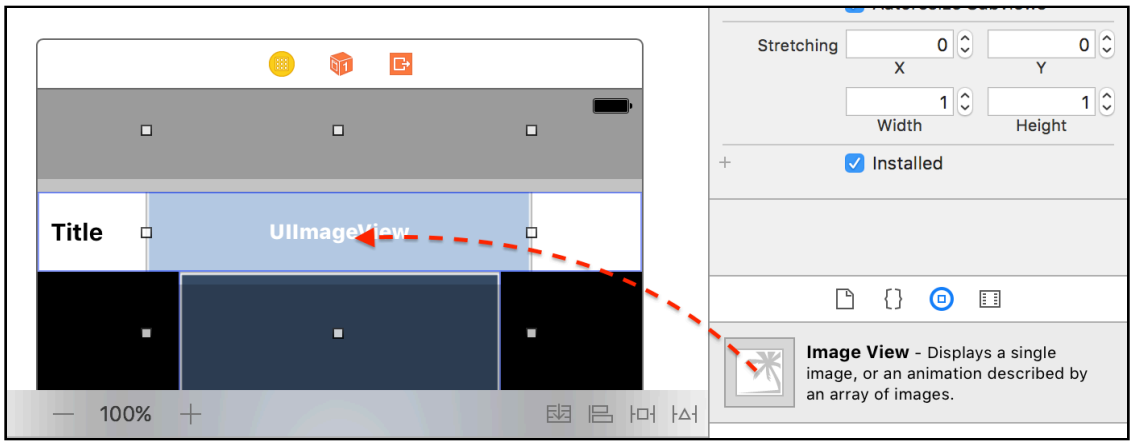
```
@IBOutlet weak var iconImageView: UIImageView!
```

Then, below the `title` property, declare a second property that'll manage the icon to be displayed:

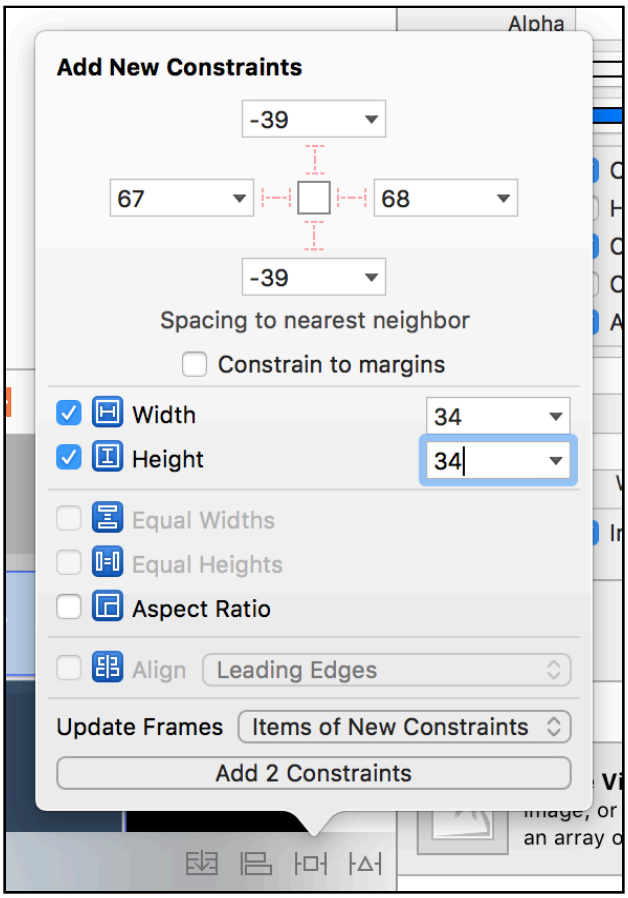
```
var icon: UIImage? {
  didSet {
    iconImageView.image = icon
  }
 }
```

Here you adding a property observer which is responsible for updating the image view whenever the icon is set.

Next, open **Main.storyboard** and drag an **Image View** from the Object Library onto the **Section Header** view in Master View Controller:



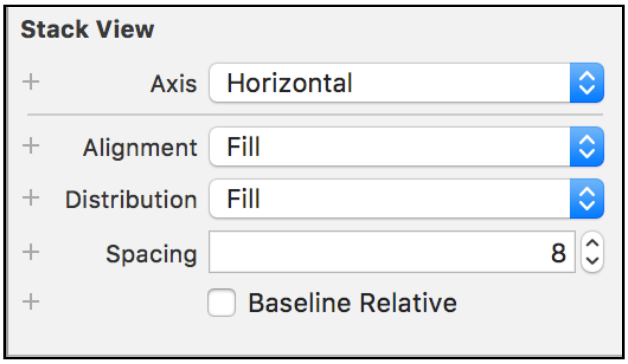
With the image view selected, add the following constraints to set the width and height to **34** using the **Pin** button located at the bottom of the storyboard canvas:



Select **Items of New Constraints** from the Update Frames menu. Click **Add 2 Constraints** to add the constraints.

Select the **Title Label** and the **Image View** in the Document Outline, and then click the **Stack View** button, found to the left of the Pin button (the one that looks like a stack of views with a downward arrow). This will create a stackView containing the label and the imageView.

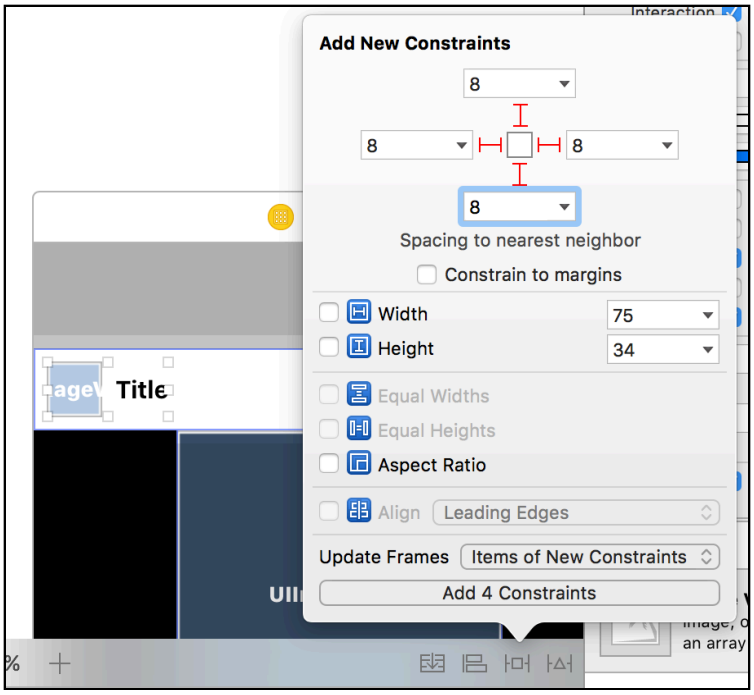
With the **Stack View** still selected, choose the Attributes Inspector and make sure that the Stack View **Axis** is **Horizontal**, **Alignment** is **Fill**, and **Spacing** is **8**.



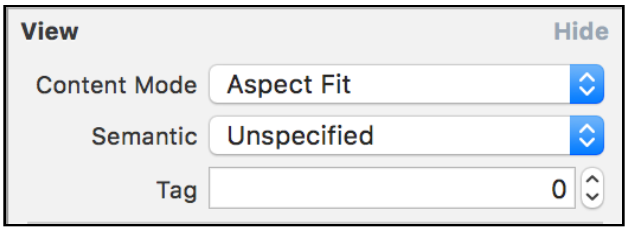
You may have to drag the title below the image in the Document Outline, to change

the order in the stackView. This is a horizontal stackView so the topmost item will be on the left.

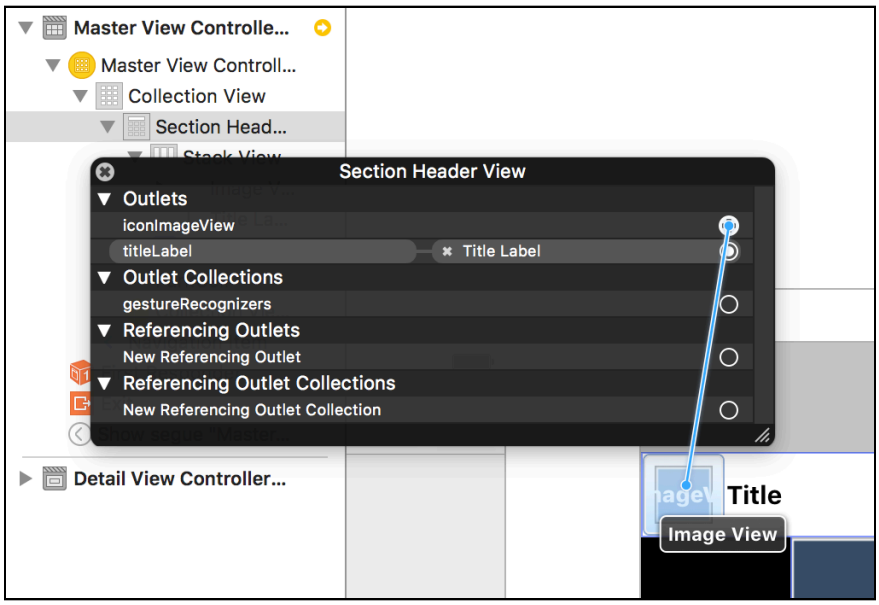
Now, select the **Stack View** in the Document Outline, click the **Pin** button and add the following new constraints:



Then select the **Image View** and set its View **Content Mode** to **Aspect Fit** from the Attributes inspector:



The last thing you need to do in Interface Builder is to connect the image view outlet you added to `SectionHeaderView` to the image view you just added to the section header. **Right-click** on Section Header View in the Document Outline and drag from the **iconImageView** outlet in the popup to the **Image View** object in the section header to connect the two:



Finally, open **MasterViewController.swift** and locate `collectionView(_:viewForSupplementaryElementOfKind:at:)`. Add the following statement just below where you set the title on the header:

```
sectionHeaderView.icon = UIImage(named: title)
```

This loads an image from the asset catalog that has the same name as the section title, and sets it as the icon on the section header view.

Build and run. The section headers now display the relevant state flag icon: