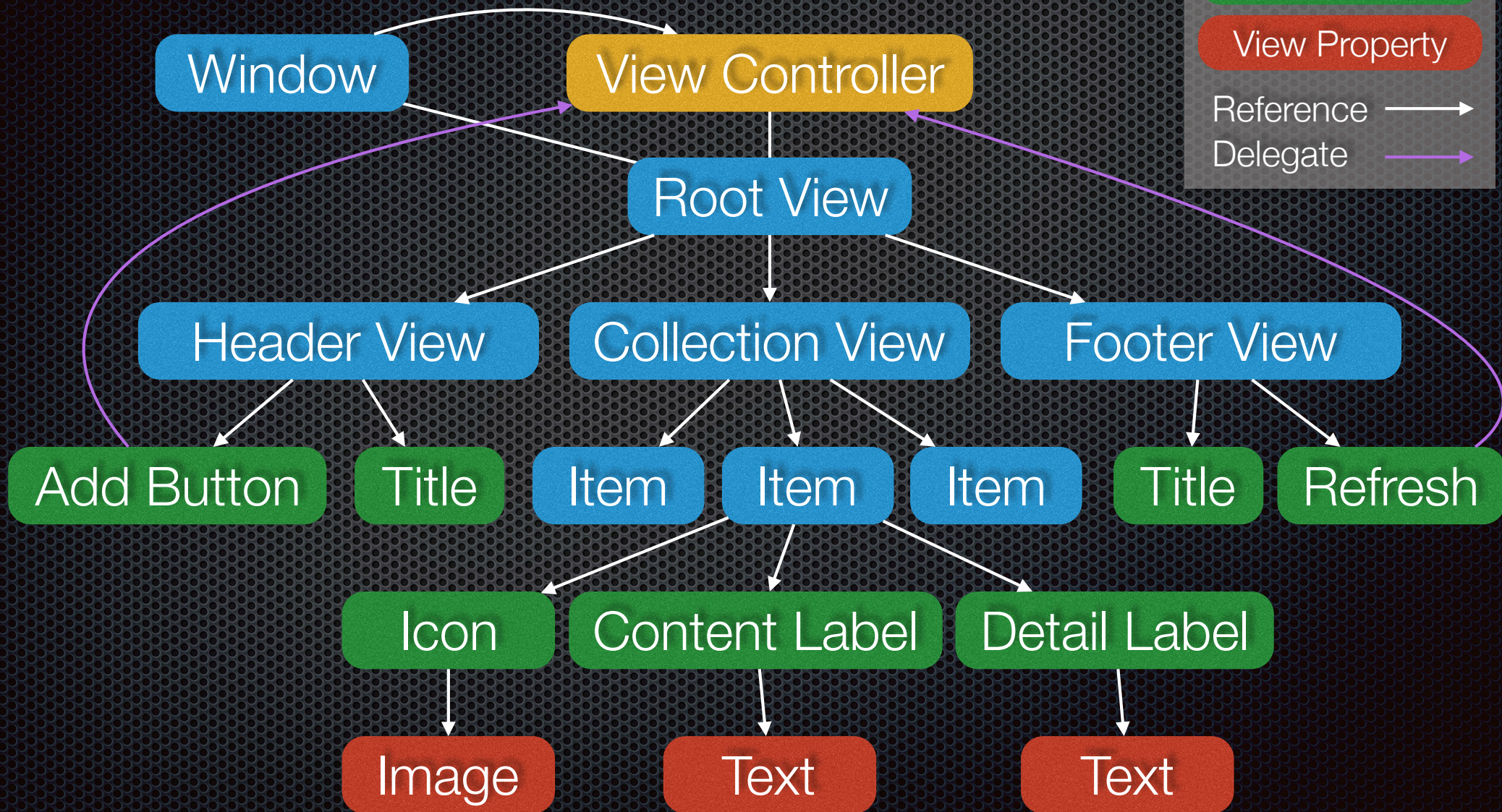
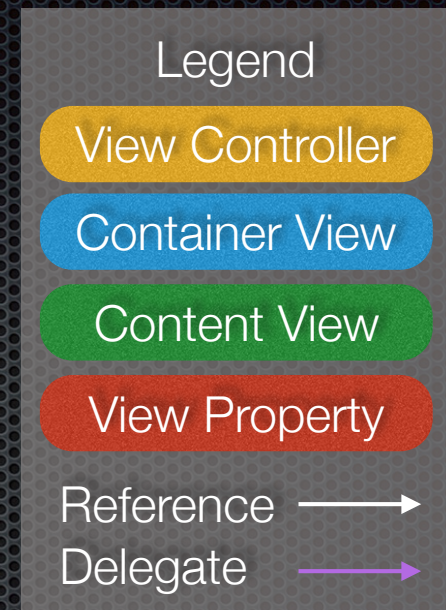


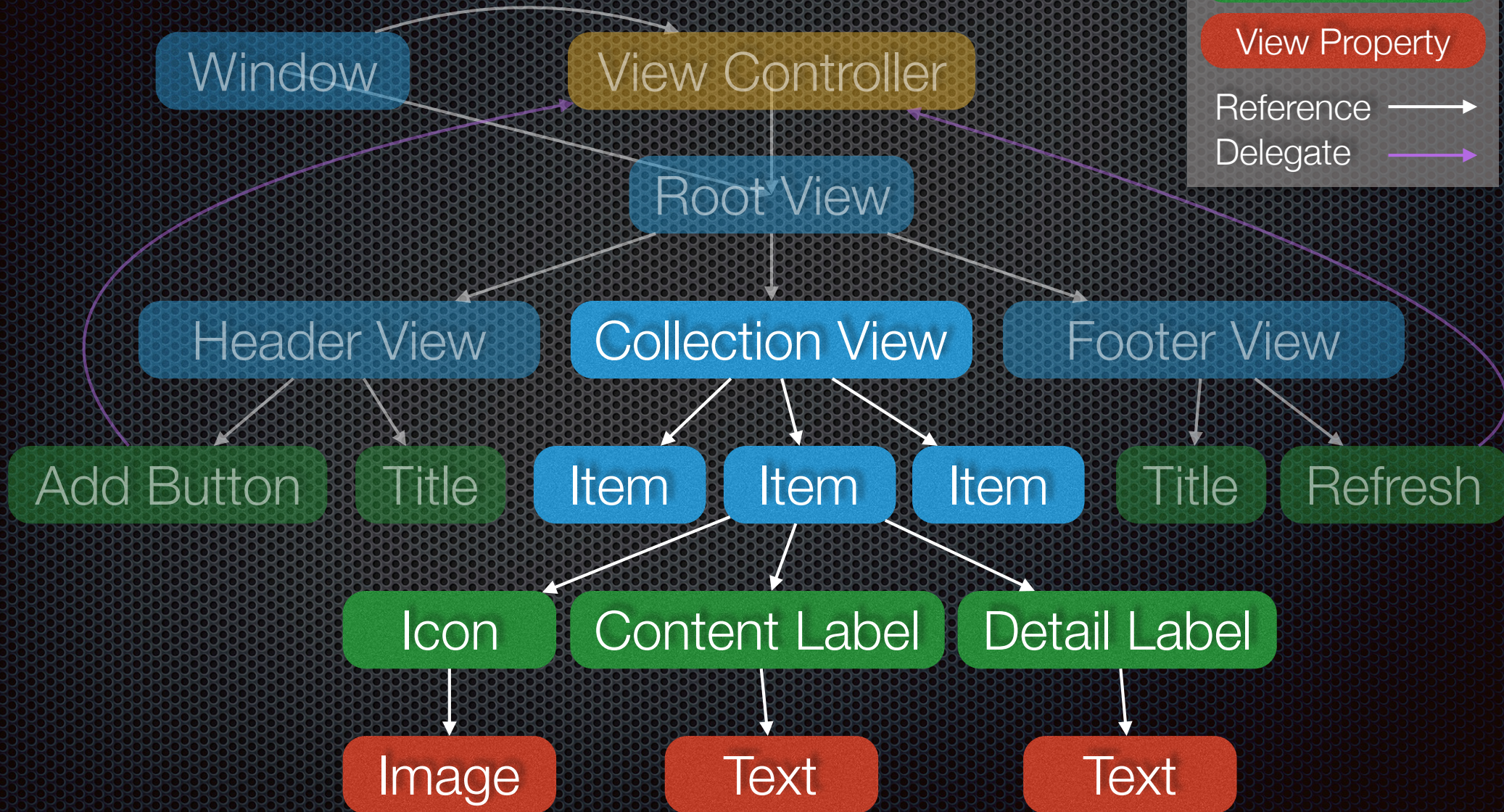
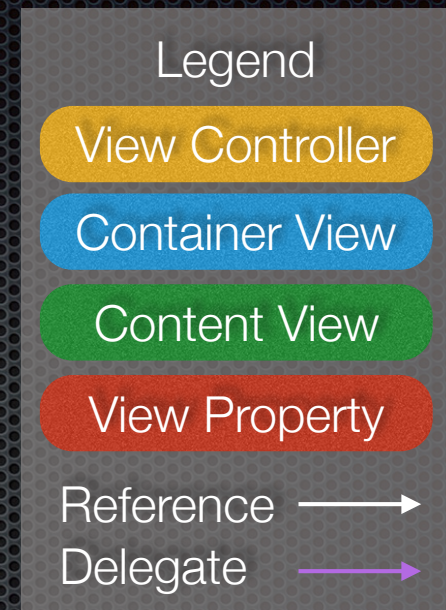
Mobile Application Programming

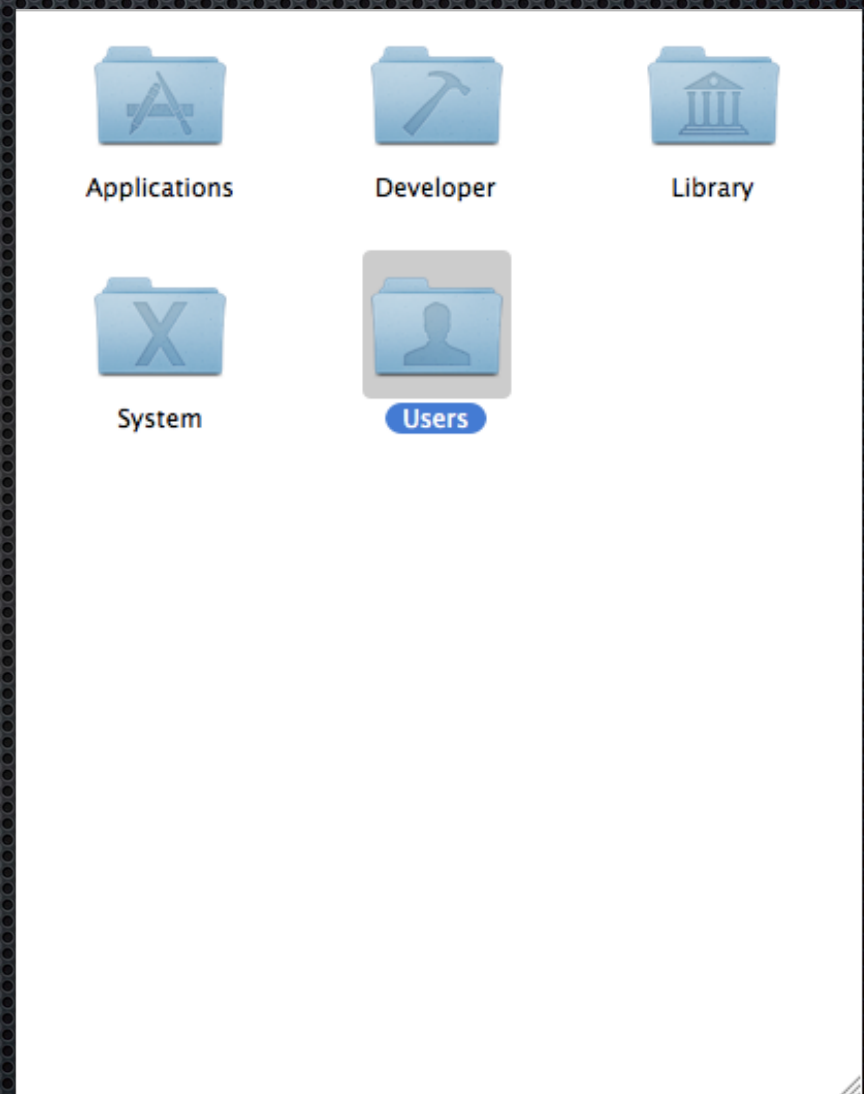
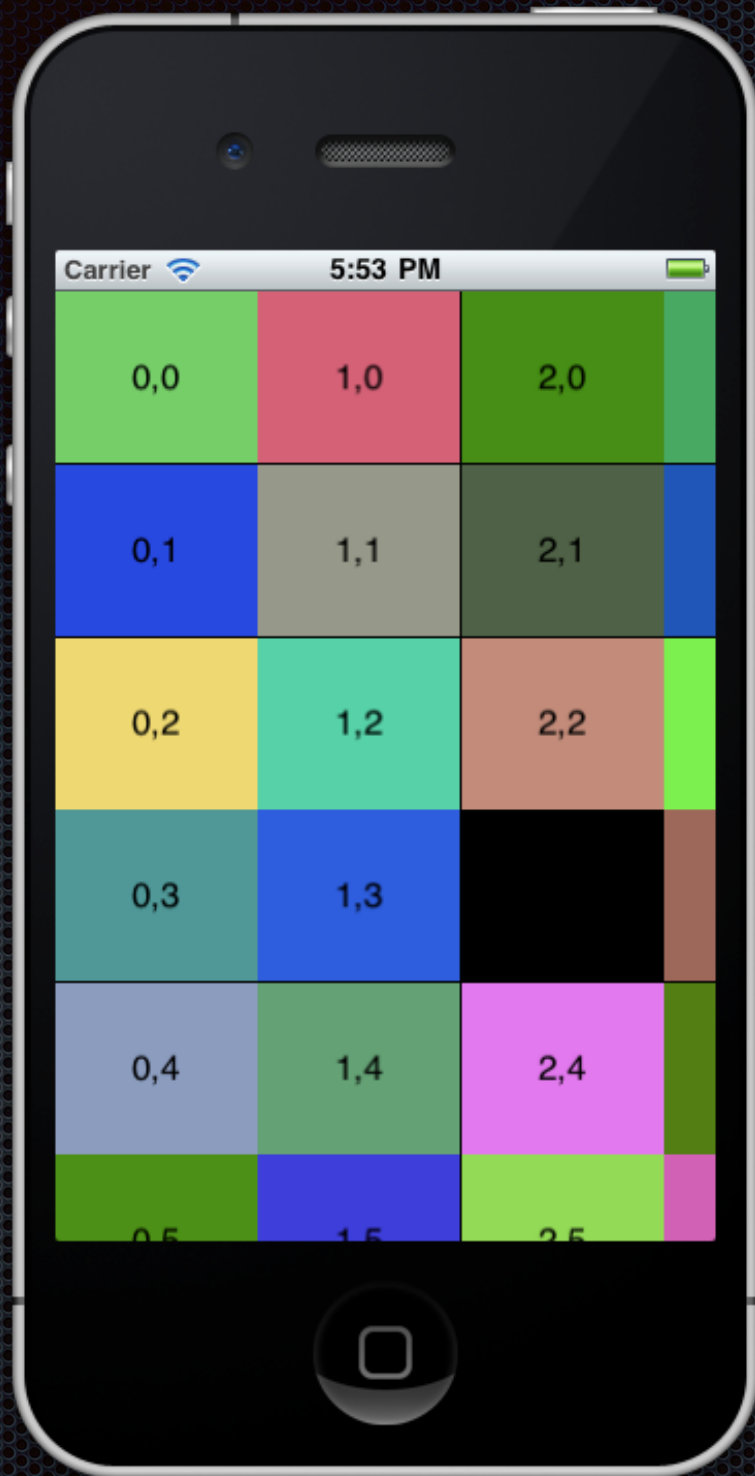
Data Driven UI

Containers & Content



Containers & Content







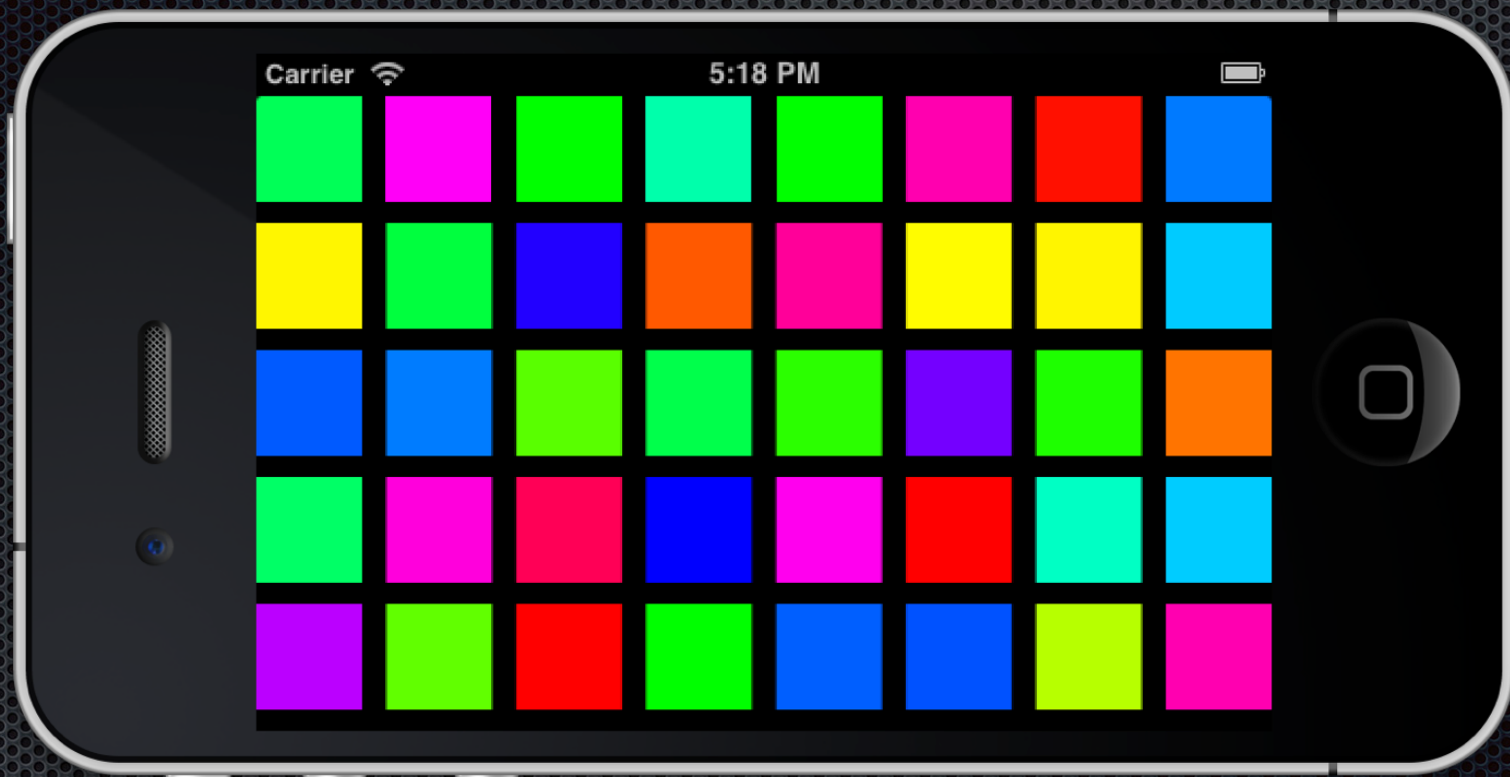
November 30, 2010

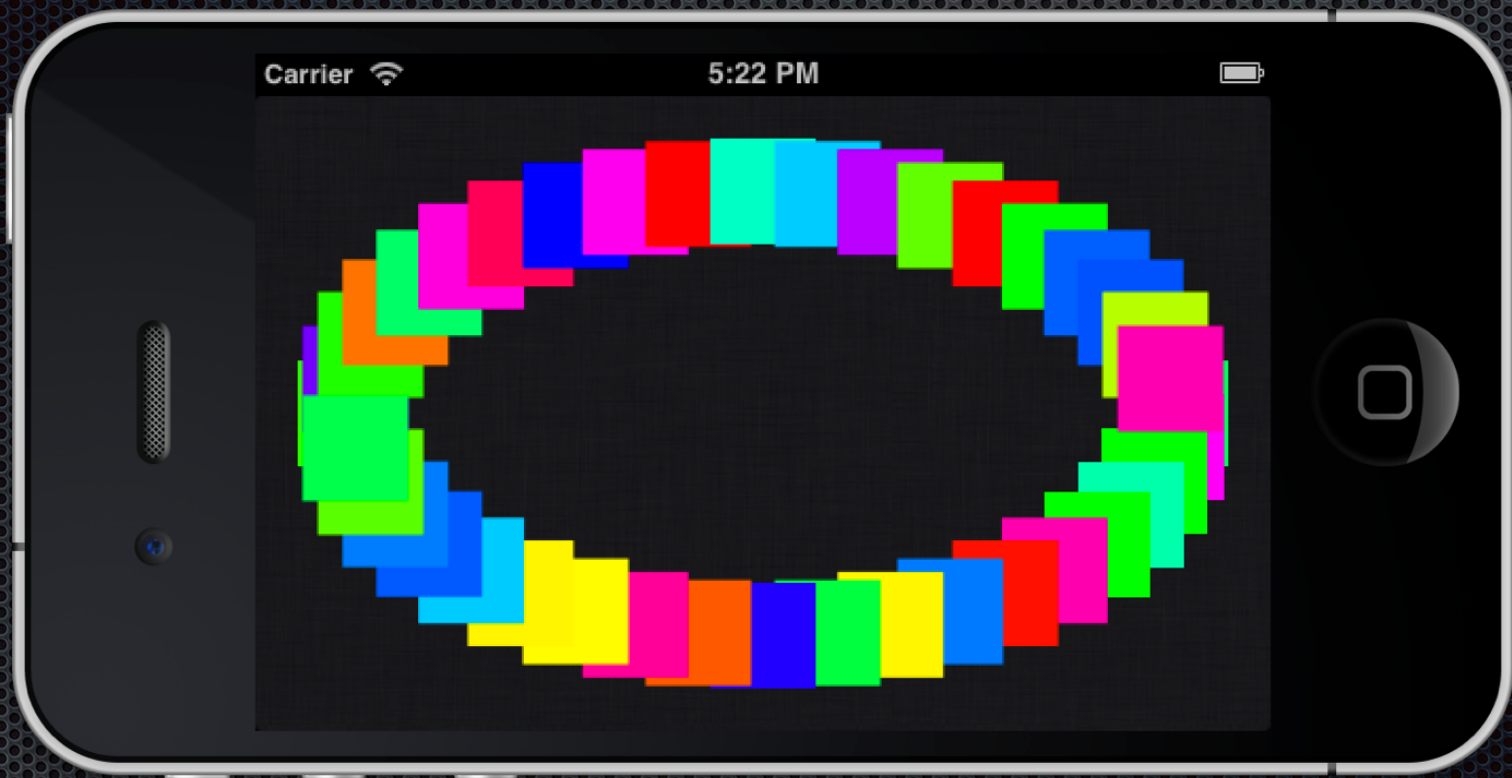


January 17, 2011

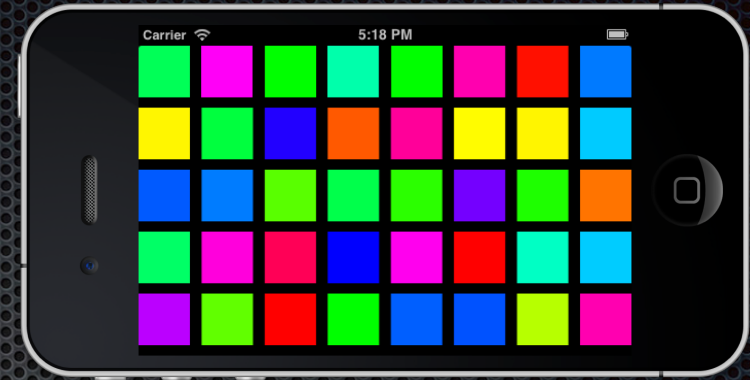


January 16, 2011





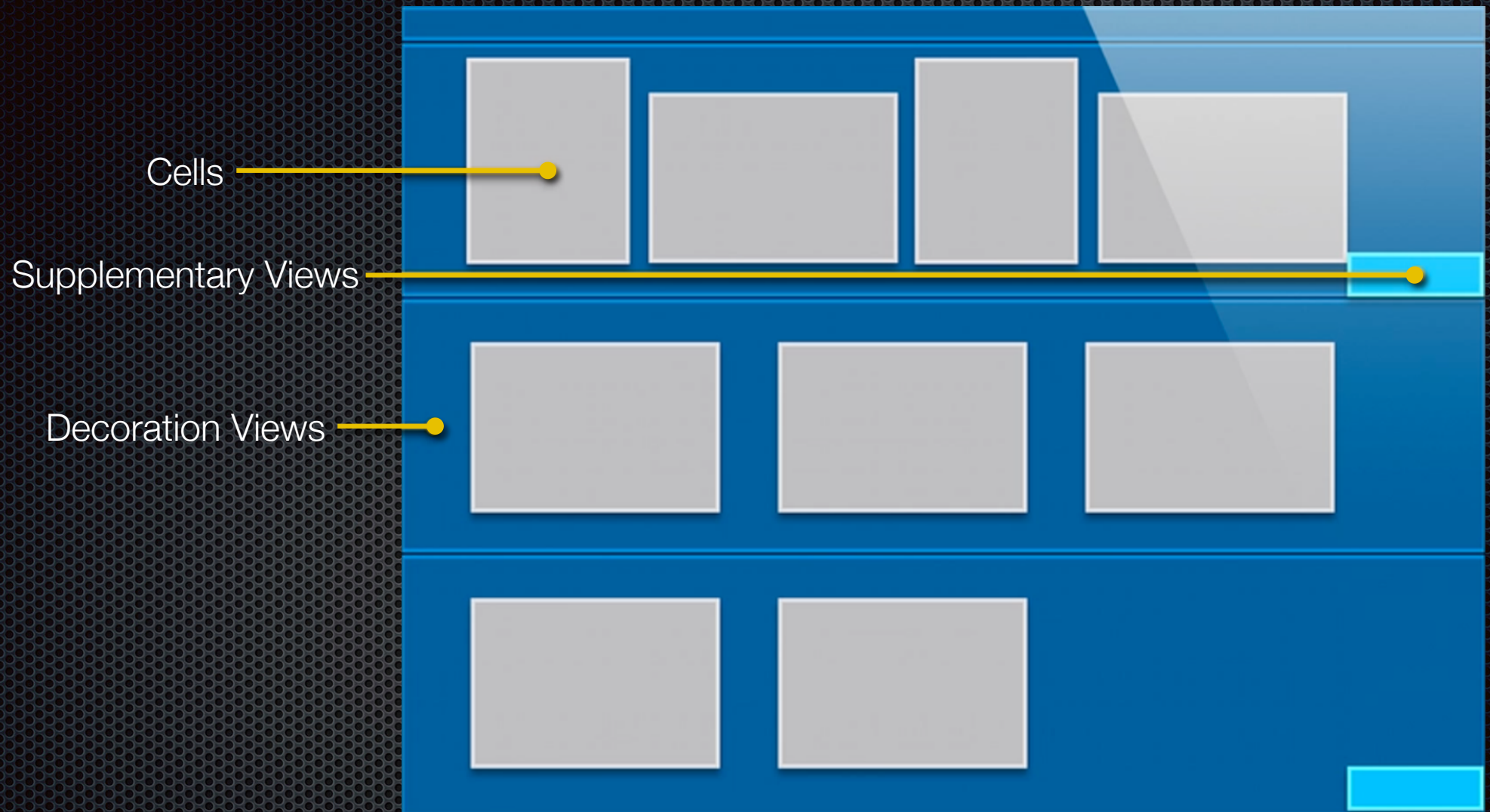
UICollectionView



- ✦ Scrolling view that loads content views as needed
- ✦ Uses a **data source** and **delegate**
- ✦ Views provided by **data source**
 - ✦ **Cells**
 - ✦ **Supplementary Views** (Optional)
- ✦ Also requires a **layout** object (typically UICollectionViewFlowLayout)
 - ✦ **Decoration Views** (Optional)



Apple WWDC 2012 - Introducing Collection Views



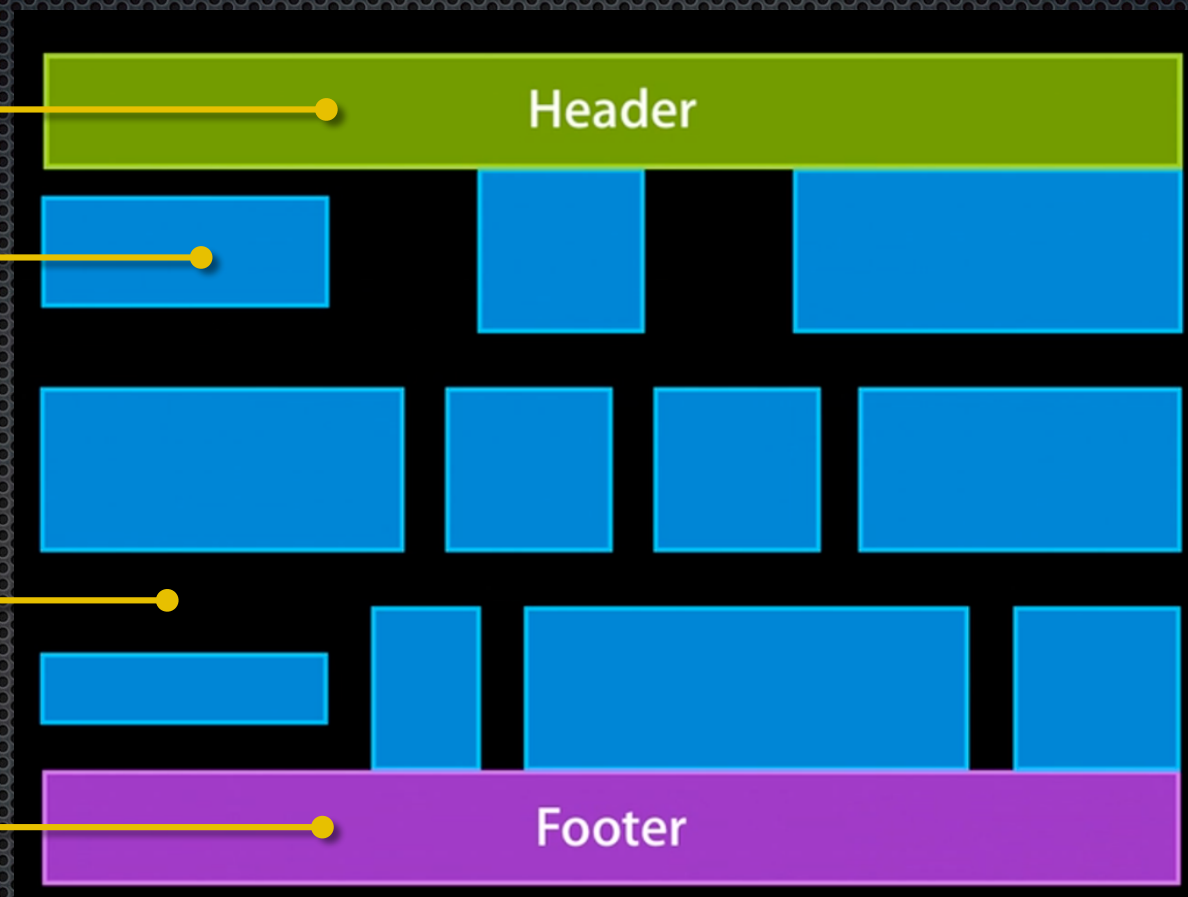
Apple WWDC 2012 - Introducing Collection Views

Supplementary Views

Cells

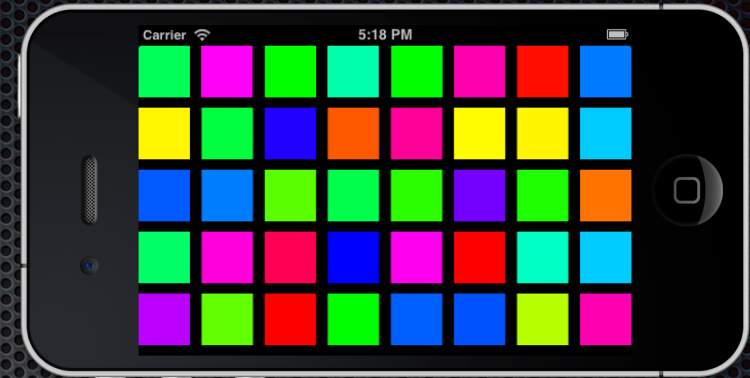
Decoration Views

Supplementary Views



Apple WWDC 2012 - Introducing Collection Views

UICollectionView



Getting Item and Section Metrics

```
func collectionView(UICollectionView, numberOfItemsInSection: Int)
    Required. Asks your data source object for the number of items in the specified section.
```

```
func collectionView(UICollectionView, numberOfSections: Int)
    Asks your data source object for the number of sections in the collection view.
```

Getting Views for Items

```
func collectionView(UICollectionView, cellForItemAt: IndexPath)
    Required. Asks your data source object for the cell that corresponds to the specified item in the collection view.
```

```
func collectionView(UICollectionView, viewForSupplementaryElementOfKind: String, at: IndexPath)
    Asks your data source object to provide a supplementary view to display in the collection view.
```

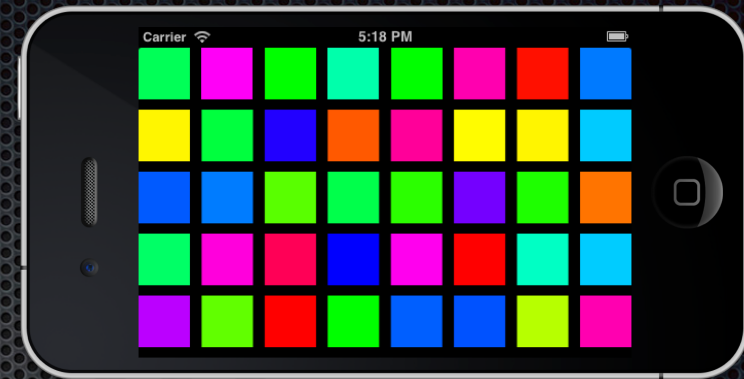
Reordering Items

```
func collectionView(UICollectionView, canMoveItemAt: IndexPath)
    Asks your data source object whether the specified item can be moved to another location in the collection view.
```

```
func collectionView(UICollectionView, moveItemAt: IndexPath, to: IndexPath)
    Tells your data source object to move the specified item to its new location.
```

UICollectionViewDataSource

UICollectionView



Managing the Selected Cells

```
func collectionView(UICollectionView, shouldSelectItemAt: IndexPath)
    Asks the delegate if the specified item should be selected.

func collectionView(UICollectionView, didSelectItemAt: IndexPath)
    Tells the delegate that the item at the specified index path was selected.

func collectionView(UICollectionView, shouldDeselectItemAt: IndexPath)
    Asks the delegate if the specified item should be deselected.

func collectionView(UICollectionView, didDeselectItemAt: IndexPath)
    Tells the delegate that the item at the specified path was deselected.
```

Managing Cell Highlighting

```
func collectionView(UICollectionView, shouldHighlightItemAt: IndexPath)
    Asks the delegate if the item should be highlighted during tracking.

func collectionView(UICollectionView, didHighlightItemAt: IndexPath)
    Tells the delegate that the item at the specified index path was highlighted.

func collectionView(UICollectionView, didUnhighlightItemAt: IndexPath)
    Tells the delegate that the highlight was removed from the item at the specified index path.
```

Tracking the Addition and Removal of Views

```
func collectionView(UICollectionView, willDisplay: UICollectionViewCell,
    forItemAt: IndexPath)
    Tells the delegate that the specified cell is about to be displayed in the collection view.

func collectionView(UICollectionView, willDisplaySupplementaryView:
    UICollectionViewReusableView, forElementKind: String, at: IndexPath)
    Tells the delegate that the specified supplementary view is about to be displayed in the collection view.

func collectionView(UICollectionView, didEndDisplaying: UICollectionViewCell,
    forItemAt: IndexPath)
    Tells the delegate that the specified cell was removed from the collection view.

func collectionView(UICollectionView, didEndDisplayingSupplementaryView:
    UICollectionViewReusableView, forElementKind: String, at: IndexPath)
    Tells the delegate that the specified supplementary view was removed from the collection view.
```

Handling Layout Changes

```
func collectionView(UICollectionView, transitionLayoutForOldLayout:
    UICollectionViewLayout, newLayout: UICollectionViewLayout)
    Asks for the custom transition layout to use when moving between the specified layouts.

func collectionView(UICollectionView,
    targetContentOffsetForProposedContentOffset: CGPoint)
    Gives the delegate an opportunity to customize the content offset for layout changes and animated updates.

func collectionView(UICollectionView, targetIndexPathForMoveFromItemAt:
    IndexPath, toProposedIndexPath: IndexPath)
    Asks the delegate for the index path to use when moving an item.
```

Managing Actions for Cells

```
func collectionView(UICollectionView, shouldShowMenuForItemAt: IndexPath)
    Asks the delegate if an action menu should be displayed for the specified item.

func collectionView(UICollectionView, canPerformAction: Selector, forItemAt:
    IndexPath, withSender: Any?)
    Asks the delegate if it can perform the specified action on an item in the collection view.

func collectionView(UICollectionView, performAction: Selector, forItemAt:
    IndexPath, withSender: Any?)
    Tells the delegate to perform the specified action on an item in the collection view.
```

Managing Focus in a Collection View

```
func collectionView(UICollectionView, canFocusItemAt: IndexPath)
    Asks the delegate whether the item at the specified index path can be focused.

func indexPathForPreferredFocusedView(in: UICollectionView)
    Asks the delegate for the index path of the cell that should be focused.

func collectionView(UICollectionView, shouldUpdateFocusIn:
    UICollectionViewFocusUpdateContext)
    Asks the delegate whether a change in focus should occur.

func collectionView(UICollectionView, didUpdateFocusIn:
    UICollectionViewFocusUpdateContext, with: UIFocusAnimationCoordinator)
    Tells the delegate that a focus update occurred.
```

UICollectionViewDelegate

Table Views

AT&T 2:23 PM 80%

Edit Favorites +

RS	Stoker Roni	mobile	i
LS	Sharp Lani	mobile	i
MJ	Jennings Michelle	mobile	i
JJ	Jennings Judson	mobile	i
SS	Stoker Steve	mobile	i
MS	Stoker Mindy	mobile	i
KB	Bullard Kimberly	mobile	i
SC	Chatelain Shaun	mobile	i
LS	Swank Lorenzo	mobile	i
SP	Porter Seth	mobile	i
BZ	Zeeman Ben	mobile	i
AM	Monroy Andres	mobile	i

★ Recents 👤 Contacts ⌨ Keypad 🗣 Voicemail

AT&T 2:21 PM 81%

All Missed Edit

mobile		
Davis Mark	1/26/15	i
mobile		
Berg Corey	1/26/15	i
mobile		
Zundel Aaron	1/25/15	i
mobile		
Kern Calvin	1/24/15	i
mobile		
Zundel Aaron (2)	1/24/15	i
mobile		
Bullard Kimberly	1/24/15	i
mobile		
Davis Mark	1/24/15	i
mobile		
Stoker Roni	1/24/15	i
mobile		
Berg Corey	1/24/15	i
mobile		
+1 (801) 268-8863	1/24/15	i
Murray, UT		
Stoker Roni	1/24/15	i
mobile		

★ Recents 👤 Contacts ⌨ Keypad 🗣 Voicemail

AT&T 2:20 PM 81%

All Contacts +

Search

H
Howe Harold
Howell Landon
Hunsaker Kellee
I
Itaihow Melissa
Itaihow Sam
Ivie John
Izatt Jason
J
Jacobs Julie
Jennings Alan
Jennings Judson
Jennings Kyle
Jennings Marianne

★ Recents 👤 Contacts ⌨ Keypad 🗣 Voicemail

Table Views

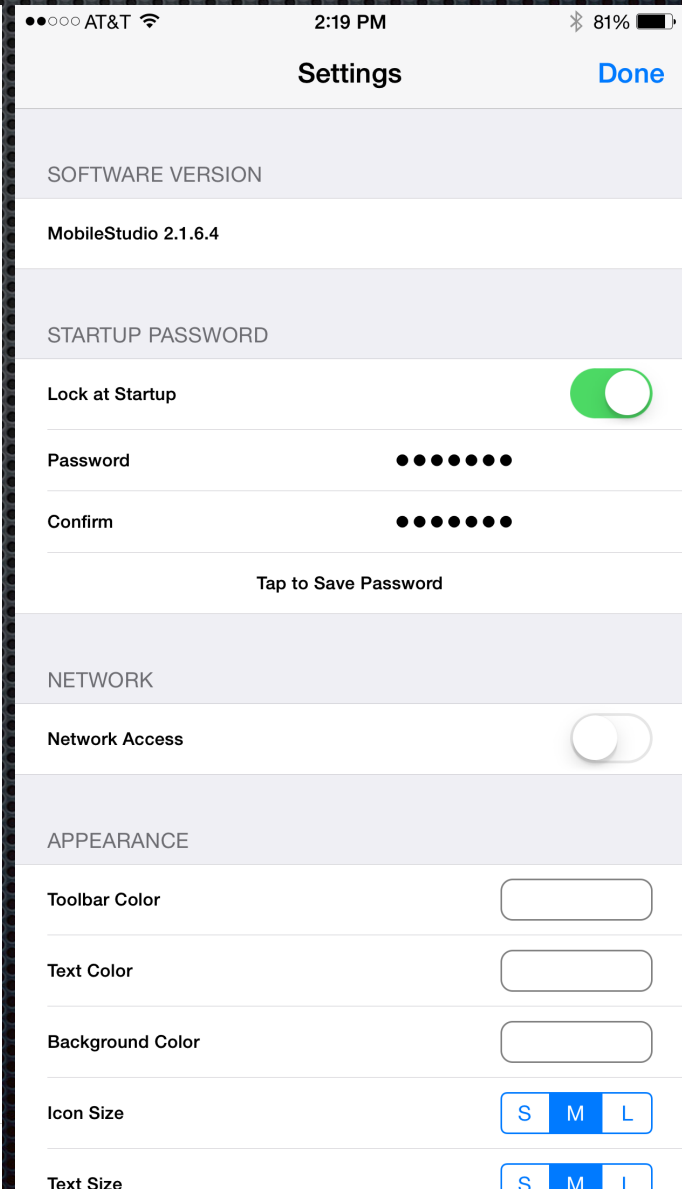
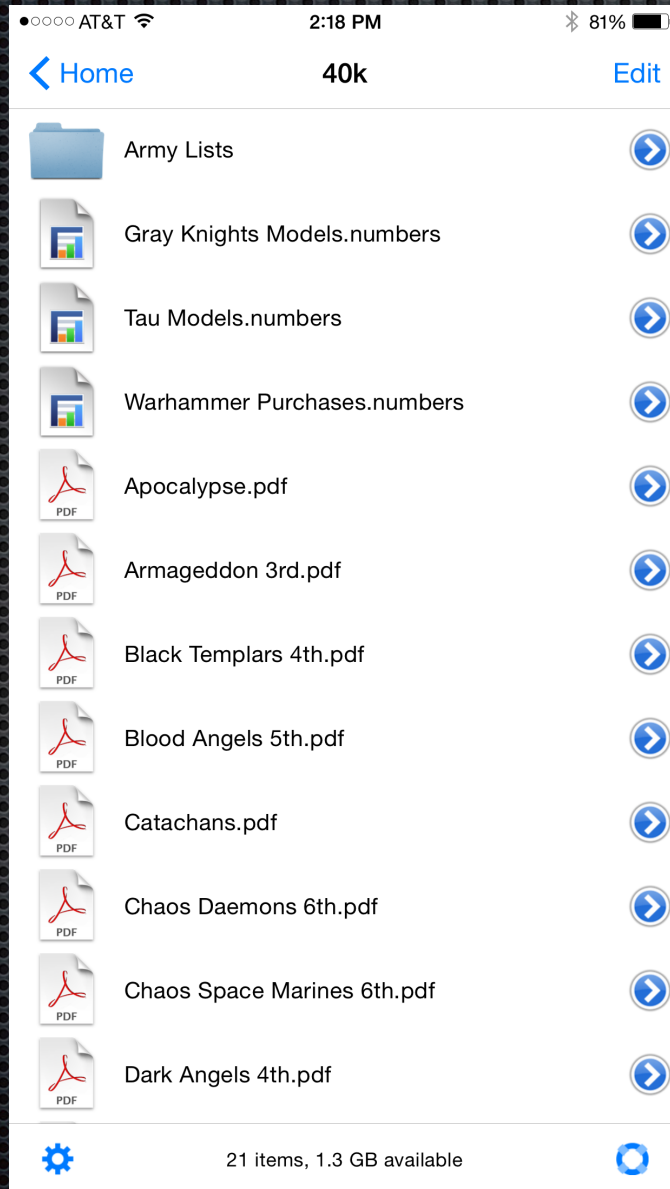
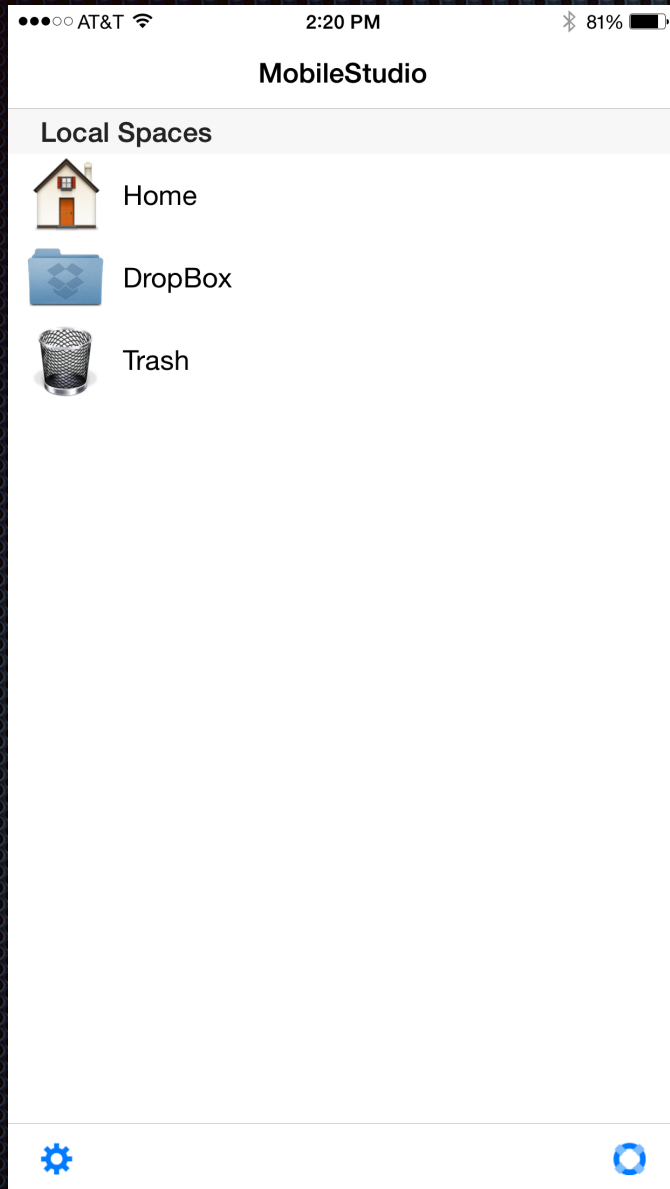


Table Views

- ✦ Great control for organizing **list data** and **hierarchal data** (multiples)
- ✦ Can contain **static data** (eg. a settings list) or **dynamic data** (eg. data pulled from a file or the web)
- ✦ Individual cells are **highly customizable**
 - ✦ The clocks in the Apple Clock app are cells, for example

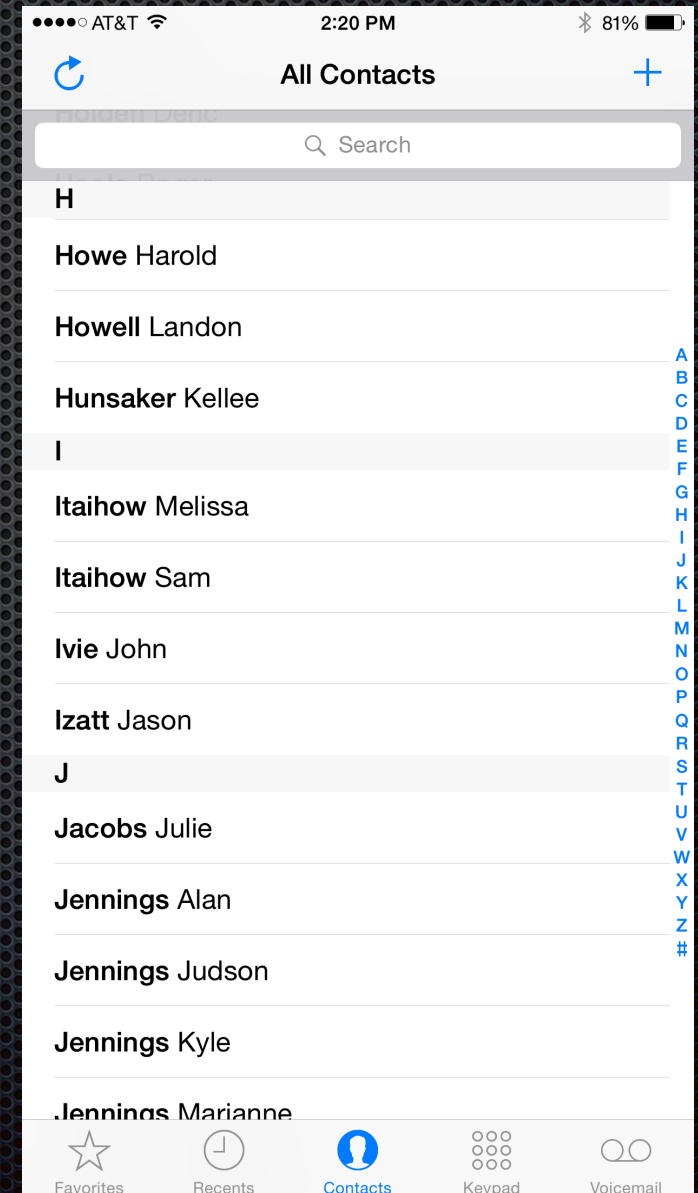


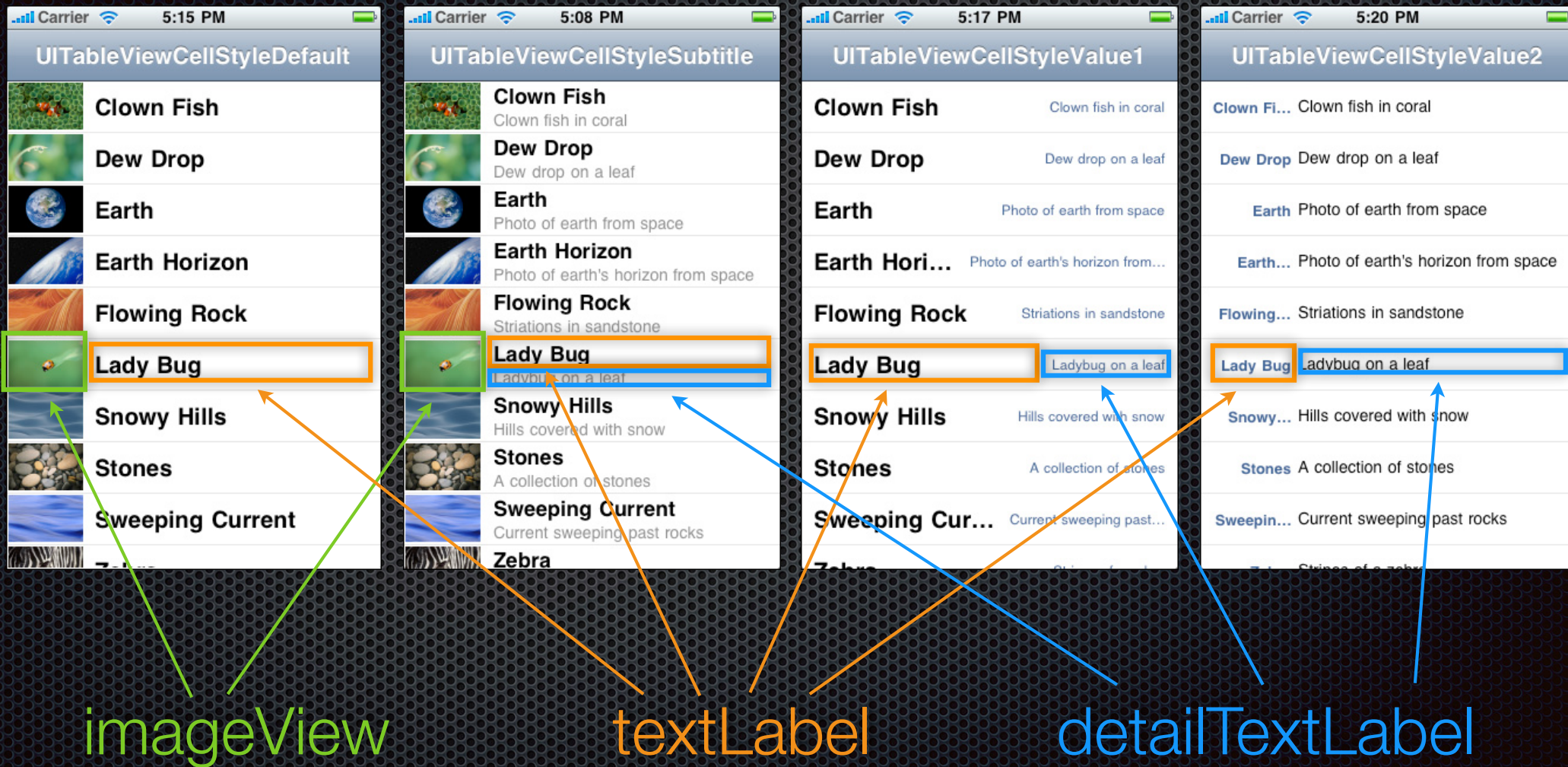
Table Views

- ✦ Set the UITableView's **dataSource** property to your **UIViewController** (more on this later) and override:

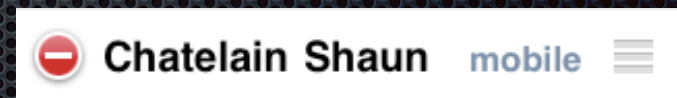
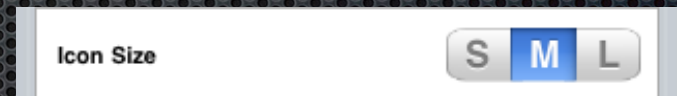
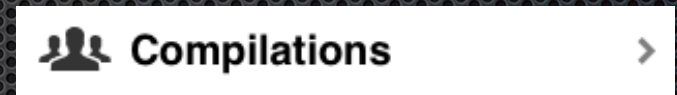
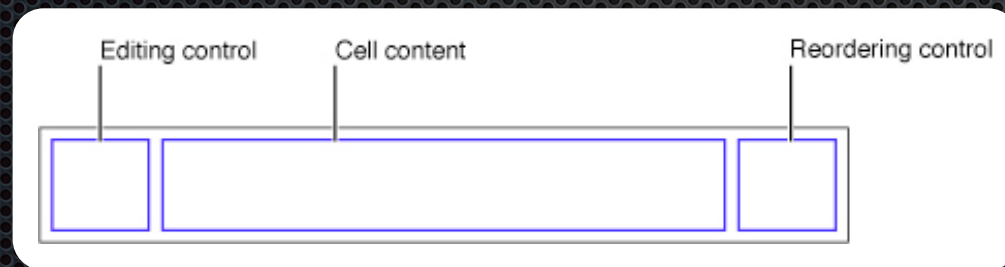
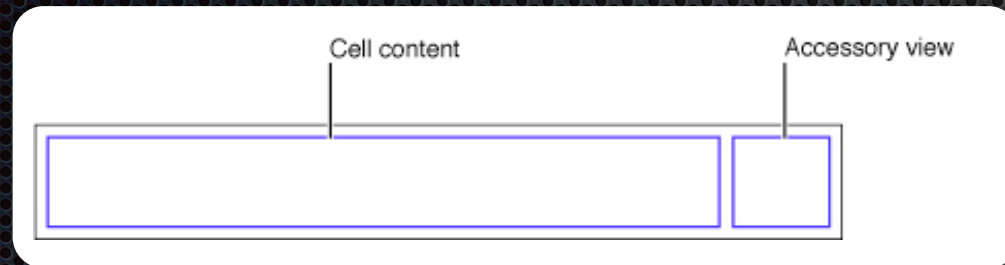
```
optional public func numberOfSections(in tableView: UITableView) -> Int // Default is 1 if not implemented
public func tableView(_ tableView: UITableView, numberOfRowsInSectionSection section: Int) -> Int
public func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell
```

- ✦ The first two methods return integers for the **number of sections** in the table, and the **number of rows** in each section
- ✦ The last returns a **UITableViewCell** object for the designated **row** and **column** in the **index path**

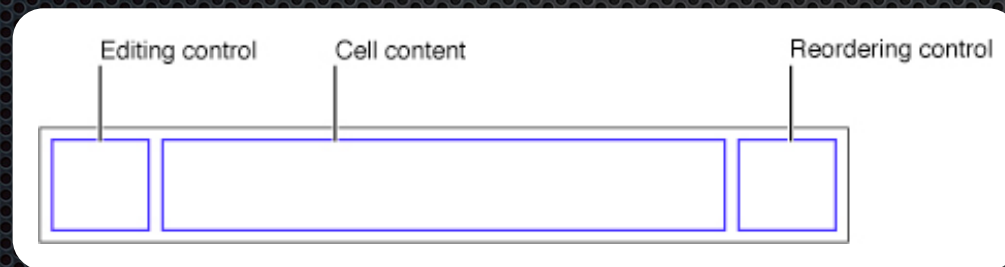
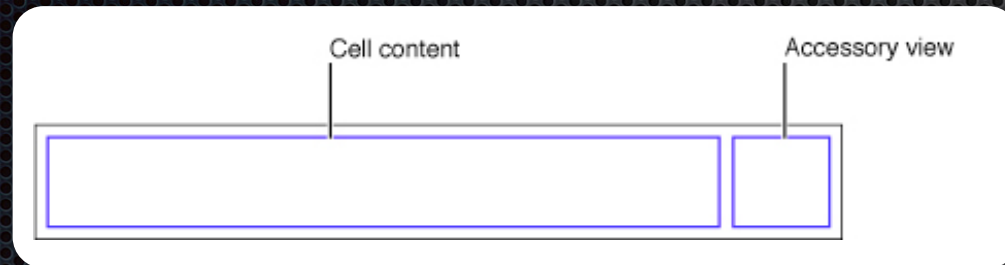
Standard Table View Cells



Standard Table View Cells



Custom Table View Cells



Subclass UITableViewCell

Table View Delegates

UITableViewDataSource

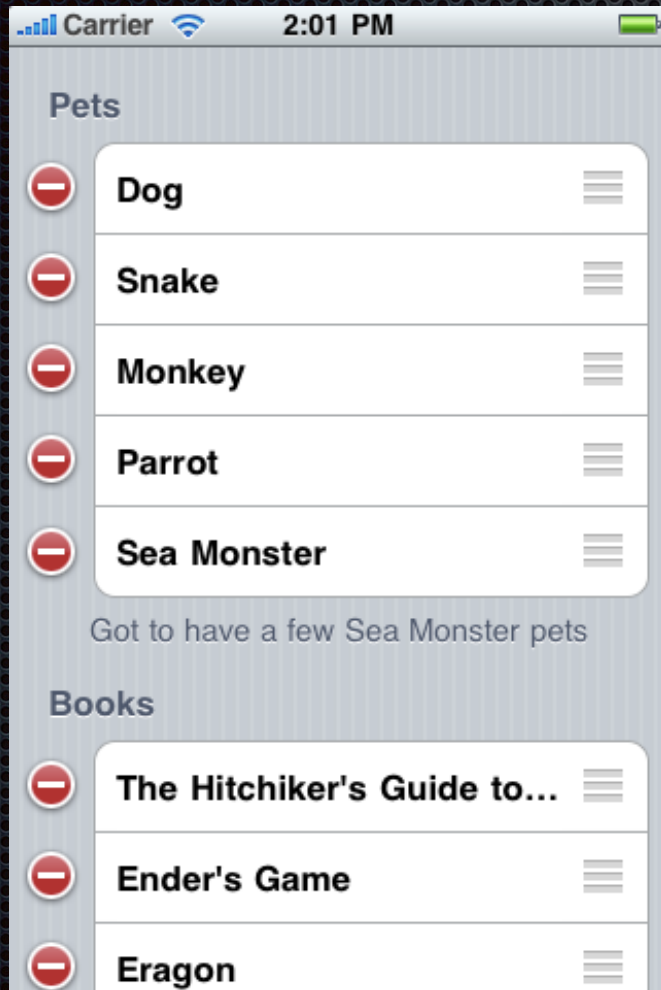


Table View Delegates

Configuring a Table View

- `tableView:cellForRowAtIndexPath:`
Required. Asks the data source for a cell to insert in a particular location of the table view.
- `numberOfSectionsInTableView:`
Asks the data source to return the number of sections in the table view.
- `tableView:numberOfRowsInSection:`
Required. Tells the data source to return the number of rows in a given section of a table view.
- `sectionIndexTitlesForTableView:`
Asks the data source to return the titles for the sections for a table view.
- `tableView:sectionForSectionIndexTitle:atIndex:`
Asks the data source to return the index of the section having the given title and section title index.
- `tableView:titleForHeaderInSection:`
Asks the data source for the title of the header of the specified section of the table view.
- `tableView:titleForFooterInSection:`
Asks the data source for the title of the footer of the specified section of the table view.

Inserting or Deleting Table Rows

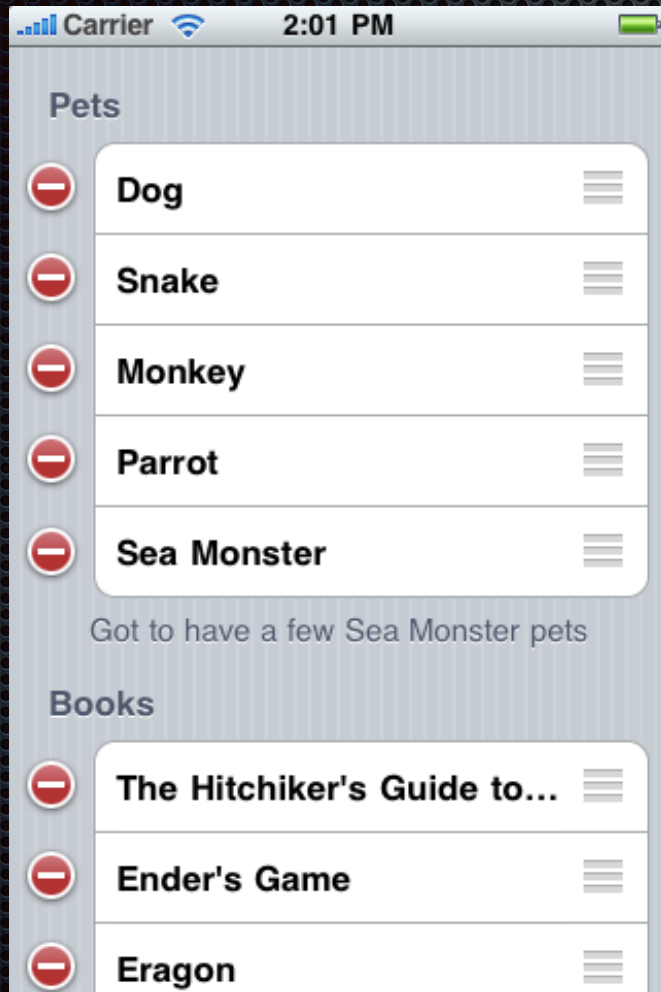
- `tableView:commitEditingStyle:forRowAtIndexPath:`
Asks the data source to commit the insertion or deletion of a specified row in the receiver.
- `tableView:canEditRowAtIndexPath:`
Asks the data source to verify that the given row is editable.

Reordering Table Rows

- `tableView:canMoveRowAtIndexPath:`
Asks the data source whether a given row can be moved to another location in the table view.
- `tableView:moveRowAtIndexPath:toIndexPath:`
Tells the data source to move a row at a specific location in the table view to another location.

UITableViewDataSource

Table View Delegates



UITableViewDataSource



UITableViewDelegate

Table View Delegates

Configuring Rows for the Table View

```
tableView(_:heightForRowAtIndexPath:)
tableView(_:estimatedHeightForRowAtIndexPath:)
tableView(_:indentationLevelForRowAtIndexPath:)
tableView(_:willDisplayCell:forRowAtIndexPath:)
```

Managing Accessory Views

```
tableView(_:editActionsForRowAtIndexPath:) Required
tableView(_:accessoryButtonTappedForRowWithIndexPath:)
```

1 Objective-C symbol hidden

Managing Selections

```
tableView(_:willSelectRowAtIndexPath:)
tableView(_:didSelectRowAtIndexPath:)
tableView(_:willDeselectRowAtIndexPath:)
tableView(_:didDeselectRowAtIndexPath:)
```

Modifying the Header and Footer of Sections

```
tableView(_:viewForHeaderInSection:)
tableView(_:viewForFooterInSection:)
tableView(_:heightForHeaderInSection:)
tableView(_:estimatedHeightForHeaderInSection:)
tableView(_:heightForFooterInSection:)
tableView(_:estimatedHeightForFooterInSection:)
tableView(_:willDisplayHeaderView:forSection:)
tableView(_:willDisplayFooterView:forSection:)
```

Editing Table Rows

```
tableView(_:willBeginEditingRowAtIndexPath:)
tableView(_:didEndEditingRowAtIndexPath:)
tableView(_:editingStyleForRowAtIndexPath:)
tableView(_:titleForDeleteConfirmationButtonForRowAtIndexPath:)
tableView(_:shouldIndentWhileEditingRowAtIndexPath:)
```

Reordering Table Rows

```
tableView(_:targetIndexPathForMoveFromRowAtIndexPath:toProposedIndexPath:)
```

Tracking the Removal of Views

```
tableView(_:didEndDisplayingCell:forRowAtIndexPath:)
tableView(_:didEndDisplayingHeaderView:forSection:)
tableView(_:didEndDisplayingFooterView:forSection:)
```

Copying and Pasting Row Content

```
tableView(_:shouldShowMenuForRowAtIndexPath:)
tableView(_:canPerformAction:forRowAtIndexPath:withSender:)
tableView(_:performAction:forRowAtIndexPath:withSender:)
```

Managing Table View Highlighting

```
tableView(_:shouldHighlightRowAtIndexPath:)
tableView(_:didHighlightRowAtIndexPath:)
tableView(_:didUnhighlightRowAtIndexPath:)
```

UITableViewDelegate