

UITableView Changes, Tips and Tricks

Session 125

Jason Beaver

iOS Frameworks

These are confidential sessions—please refrain from streaming, blogging, or taking pictures

Introduction

- What's New
 - Automatic Dimensions
 - Automatic Animation Style
 - New Update Methods
 - Menu Support
 - Multiple Selection
 - Automatic Cell Loading
 - Static Table Views
- Tips and Tricks

What's New

Automatic Dimensions

Automatic Dimensions



Automatic Dimensions



Automatic Dimensions



```
- (CGFloat)tableView:(UITableView *)tableView  
heightForHeaderInSection:(NSInteger)section
```

Automatic Dimensions



```
- (CGFloat)tableView:(UITableView *)tableView  
heightForHeaderInSection:(NSInteger)section
```

Automatic Dimensions

5



CGFloat UITableViewAutomaticDimension

What's New

Automatic Animation Style

Automatic Animation Style



```
insertRowAtIndexPath:[NSIndexPath indexPathForRow:0 inSection:0]  
withRowAnimation:UITableViewRowAnimationBottom
```

Automatic Animation Style



```
deleteRowAtIndexPath:[NSIndexPath indexPathForRow:0 inSection:0]  
withRowAnimation:UITableViewRowAnimationBottom
```

Automatic Animation Style



```
insertRowAtIndexPath:[NSIndexPath indexPathForRow:3 inSection:0]  
withRowAnimation:UITableViewRowAnimationTop
```

Automatic Animation Style



```
deleteRowAtIndexPath:[NSIndexPath indexPathForRow:3 inSection:0]  
withRowAnimation:UITableViewRowAnimationTop
```

Automatic Animation Style



```
insertRowAtIndexPath:[NSIndexPath indexPathForRow:0 inSection:0]  
withRowAnimation:UITableViewRowAnimationTop
```

Automatic Animation Style



```
deleteRowAtIndexPath:[NSIndexPath indexPathForRow:0 inSection:0]  
withRowAnimation:UITableViewRowAnimationTop
```

Automatic Animation Style



```
insertRowAtIndexPath:[NSIndexPath indexPathForRow:3 inSection:0]  
withRowAnimation:UITableViewRowAnimationBottom
```


Automatic Animation Style



```
deleteRowAtIndexPath:[NSIndexPath indexPathForRow:3 inSection:0]  
withRowAnimation:UITableViewRowAnimationBottom
```

Automatic Animation Style

5



`UITableViewRowAnimationAutomatic`

What's New

New Update Methods

New Update Methods



New Update Methods



New Update Methods



New Update Methods

5



```
- (void)moveSection:(NSInteger)section  
    toSection:(NSInteger)newSection
```

New Update Methods



```
- (void)moveSection:(NSInteger)section  
    toSection:(NSInteger)newSection
```


New Update Methods

- (void)moveSection:(NSInteger)section
toSection:(NSInteger)newSection
- (void)insertSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation
- (void)deleteSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation
- (void)reloadSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation

New Update Methods

- (void)moveSection:(NSInteger)section
toSection:(NSInteger)newSection
- (void)insertSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation
- (void)deleteSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation
- (void)reloadSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation

New Update Methods

- (void)moveSection:(NSInteger)section
toSection:(NSInteger)newSection
- (void)insertSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation
- (void)deleteSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation
- (void)reloadSections:(NSIndexSet *)sections
withRowAnimation:(UITableViewRowAnimation)animation

New Update Methods

5



- `(void)moveRowAtIndexPath:(NSIndexPath *)indexPath
toIndexPath:(NSIndexPath *)newIndexPath`

New Update Methods



```
- (void)moveRowAtIndexPath:(NSIndexPath *)indexPath  
    toIndexPath:(NSIndexPath *)newIndexPath
```

New Update Methods

- (void)moveRowAtIndexPath:(NSIndexPath *)indexPath
toIndexPath:(NSIndexPath *)newIndexPath
- (void)insertRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation
- (void)deleteRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation
- (void)reloadRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation

New Update Methods

- (void)moveRowAtIndexPath:(NSIndexPath *)indexPath
toIndexPath:(NSIndexPath *)newIndexPath
- (void)insertRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation
- (void)deleteRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation
- (void)reloadRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation

New Update Methods

- (void)moveRowAtIndexPath:(NSIndexPath *)indexPath
toIndexPath:(NSIndexPath *)newIndexPath
- (void)insertRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation
- (void)deleteRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation
- (void)reloadRowsAtIndexPaths:(NSArray *)indexPaths
withRowAnimation:(UITableViewRowAnimation)animation

Demo

Luke Hiesterman
iOS Frameworks Engineer

What's New

Menu Support

Menu Support



Menu Support

5



```
- (BOOL)tableView:(UITableView *)tableView  
    shouldShowMenuForRowAtIndexPath:(NSIndexPath *)indexPath
```

Menu Support

5



```
- (BOOL)tableView:(UITableView *)tableView  
  canPerformAction:(SEL)action  
  forRowAtIndexPath:(NSIndexPath *)indexPath  
  withSender:(id)sender
```

Menu Support



Menu Support

5



```
- (void)tableView:(UITableView *)tableView  
    performAction:(SEL)action  
    forRowAtIndexPath:(NSIndexPath *)indexPath  
    withSender:(id)sender
```

What's New

Multiple Selection

Multiple Selection



Multiple Selection



Table View Properties

`BOOL allowsMultipleSelection`

`BOOL allowsMultipleSelectionDuringEditing`

Multiple Selection

Table View Properties

BOOL allowsMultipleSelection

BOOL allowsMultipleSelectionDuringEditing

BOOL allowsSelection

BOOL allowsSelectionDuringEditing

Multiple Selection



Table View Properties

`BOOL allowsMultipleSelection`

`BOOL allowsMultipleSelectionDuringEditing`

Table View Method

– `(NSArray *)indexPathsForSelectedRows`

Multiple Selection

Table View Properties

`BOOL allowsMultipleSelection`

`BOOL allowsMultipleSelectionDuringEditing`

Table View Method

– `(NSArray *)indexPathsForSelectedRows`

– `(NSIndexPath *)indexPathForSelectedRow`

Multiple Selection



Table View Cell Property

`UIView *multipleSelectionBackgroundView`

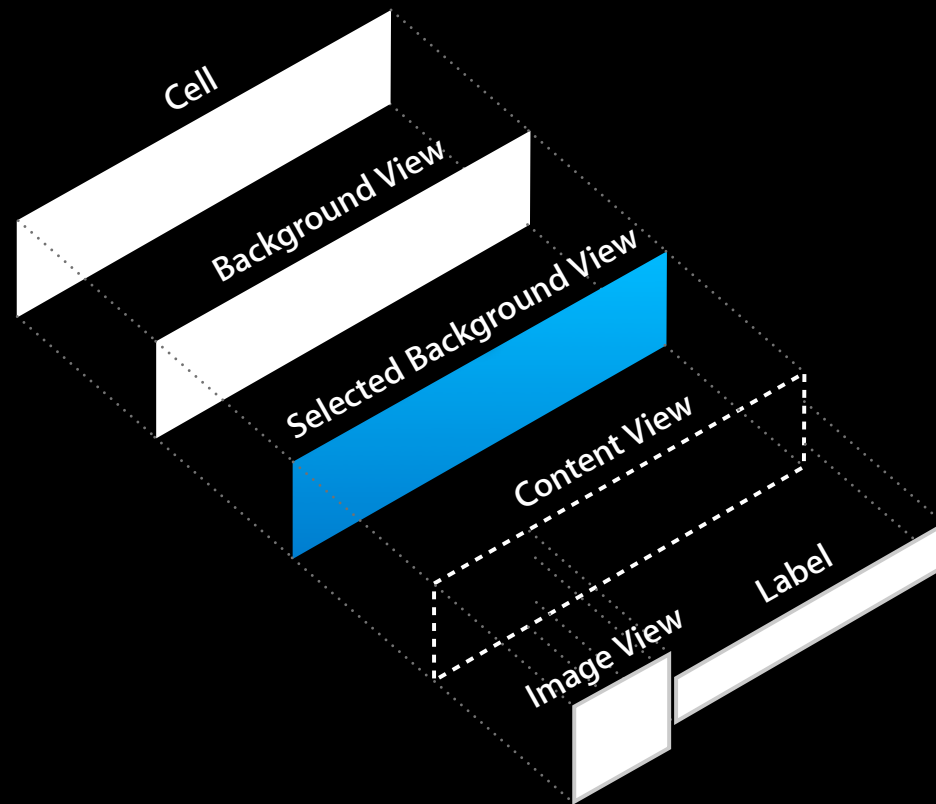
Multiple Selection

Table View Cell Property

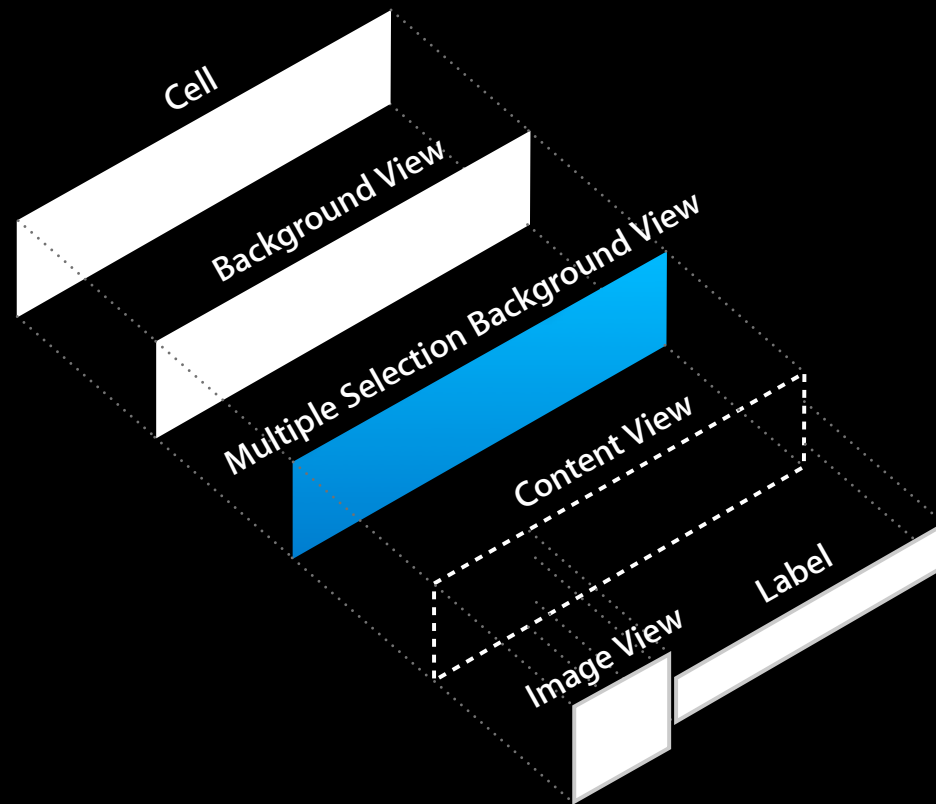
`UIView *multipleSelectionBackgroundView`

`UIView *selectedBackgroundView`

Single Selection



Multiple Selection



What's New

Automatic Cell Loading

Automatic Cell Loading

```
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    NSString *reuseID = [self reuseIdentifierForRow:indexPath];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseID];

    if (cell == nil) {
        if ([reuseID isEqualToString:@"Reuse Identifier 1"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
                reuseIdentifier:reuseID];

            // Setup cell
        } else if ([reuseID isEqualToString:@"Reuse Identifier 2"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle
                reuseIdentifier:reuseID];

            // Setup cell
        } ...
    }

    // Configure cell

    return cell;
}
```

Automatic Cell Loading

```
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    NSString *reuseID = [self reuseIdentifierForRow:indexPath];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseID];

    if (cell == nil) {
        if ([reuseID isEqualToString:@"Reuse Identifier 1"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
                reuseIdentifier:reuseID];

            // Setup cell
        } else if ([reuseID isEqualToString:@"Reuse Identifier 2"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle
                reuseIdentifier:reuseID];

            // Setup cell
        } ...
    }

    // Configure cell

    return cell;
}
```

Automatic Cell Loading

```
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    NSString *reuseID = [self reuseIdentifierForRow:indexPath];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseID];

    if (cell == nil) {
        if ([reuseID isEqualToString:@"Reuse Identifier 1"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
                reuseIdentifier:reuseID];

            // Setup cell
        } else if ([reuseID isEqualToString:@"Reuse Identifier 2"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle
                reuseIdentifier:reuseID];

            // Setup cell
        } ...
    }

    // Configure cell

    return cell;
}
```

Automatic Cell Loading

```
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    NSString *reuseID = [self reuseIdentifierForRow:indexPath];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseID];

    if (cell == nil) {
        if ([reuseID isEqualToString:@"Reuse Identifier 1"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
                reuseIdentifier:reuseID];

            // Setup cell
        } else if ([reuseID isEqualToString:@"Reuse Identifier 2"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle
                reuseIdentifier:reuseID];

            // Setup cell
        } ...
    }

    // Configure cell

    return cell;
}
```

Automatic Cell Loading

```
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    NSString *reuseID = [self reuseIdentifierForRow:indexPath];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseID];

    if (cell == nil) {
        if ([reuseID isEqualToString:@"Reuse Identifier 1"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
                reuseIdentifier:reuseID];

            // Setup cell
        } else if ([reuseID isEqualToString:@"Reuse Identifier 2"]) {
            cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle
                reuseIdentifier:reuseID];

            // Setup cell
        } ...
    }

    // Configure cell

    return cell;
}
```

Automatic Cell Loading

```
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    NSString *reuseID = [self reuseIdentifierForRow:indexPath];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseID];

    // Configure cell

    return cell;
}
```


Automatic Cell Loading

```
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndex:(NSIndexPath *)indexPath
{
    NSString *reuseID = [self reuseIdentifierForRow:indexPath];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseID];

    // Configure cell

    return cell;
}
```

Automatic Cell Loading



```
- (void)registerNib:(UINib *)nib  
    forCellReuseIdentifier:(NSString *)identifier
```

Automatic Cell Loading

```
- (void)registerNib:(UINib *)nib  
    forCellReuseIdentifier:(NSString *)identifier
```

Nib Constraints

- Nib must contain a single table view cell

Automatic Cell Loading

```
- (void)registerNib:(UINib *)nib  
    forCellReuseIdentifier:(NSString *)identifier
```

Nib Constraints

- Nib must contain a single table view cell
- Cell's reuse identifier must match the one specified above (or be nil)

Automatic Cell Loading

```
- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    NSString *reuseID = [self reuseIdentifierForRow:indexPath];
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:reuseID];

    // Configure cell

    return cell;
}
```

What's New

Static Table Views

Static Table Views



Static Table Views



Introducing Interface Builder Storyboarding

Presidio
Wednesday 11:30AM

Tips & Tricks

Action Row

Action Row



Demo

Luke Hiesterman
iOS Frameworks Engineer

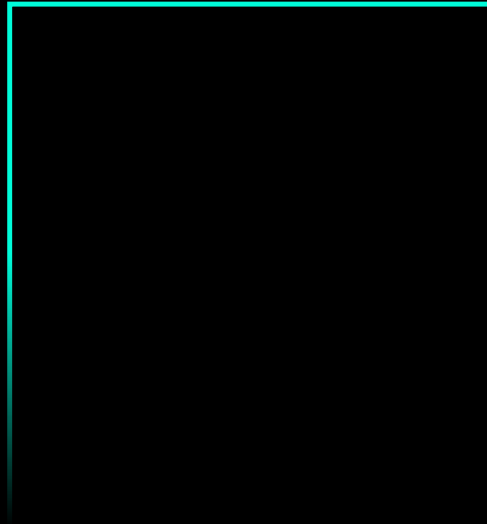
Tips & Tricks

Floating Views

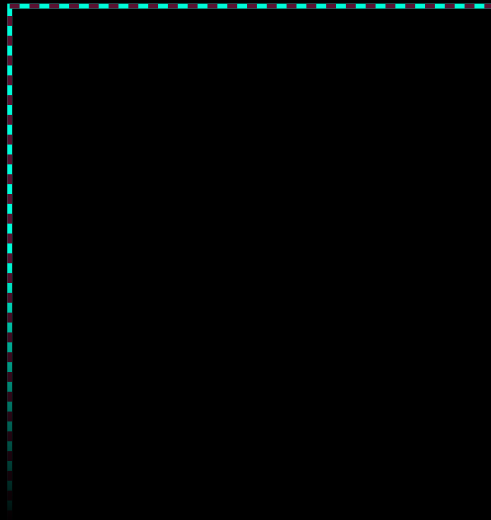
Floating Views



Floating Views



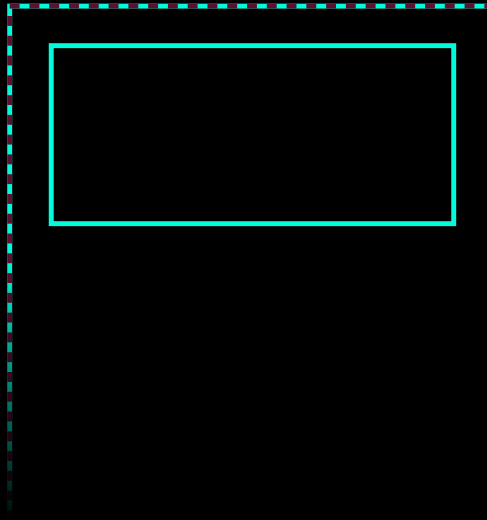
Floating Views



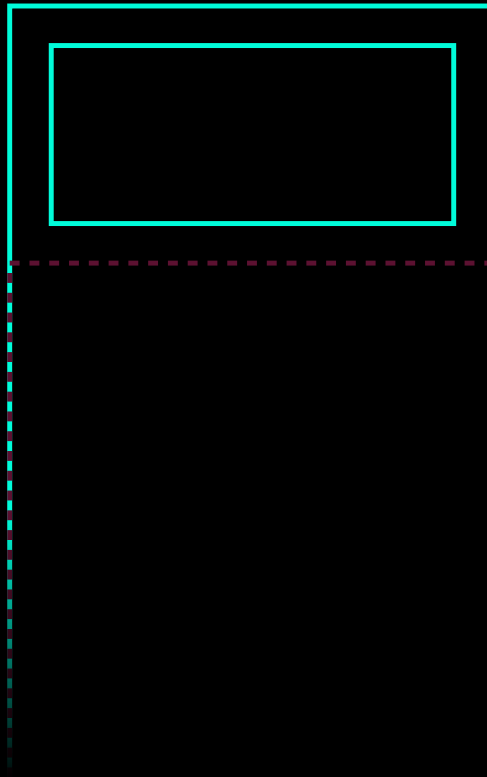
Floating Views



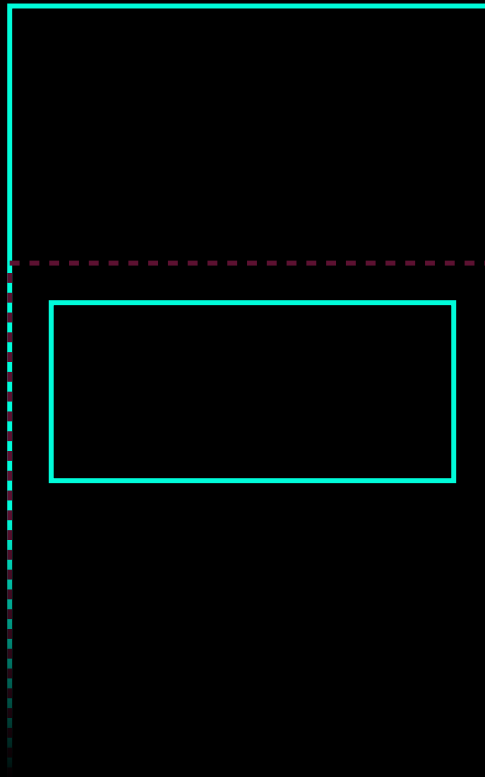
Floating Views



Floating Views



Floating Views

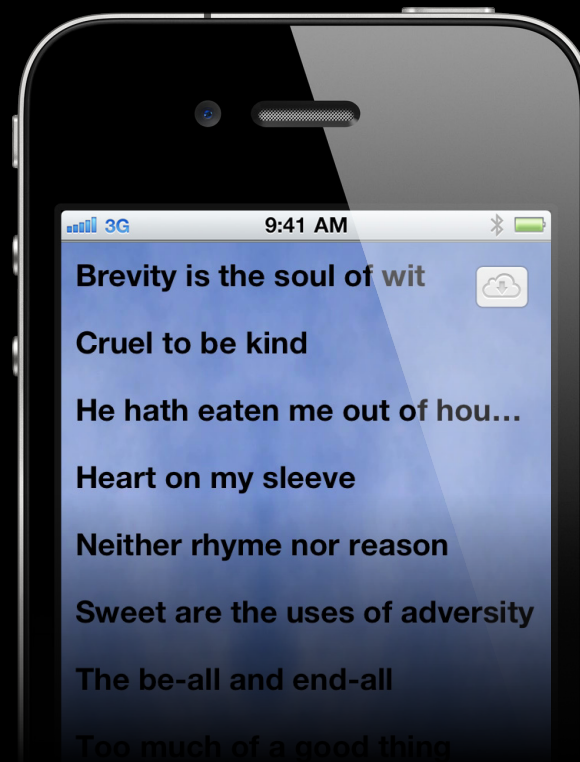


- (void)scrollViewDidScroll:(UIScrollView *)scrollView

Tips & Tricks

Dynamic Background View

Dynamic Background View



Demo

Luke Hiesterman
iOS Frameworks Engineer

Summary

- New Features
 - Automatic Dimensions
 - Automatic Animation Style
 - New Update Methods
 - Menu Support
 - Multiple Selection
 - Automatic Cell Loading
 - Static Table Views
- Tips and Tricks

More Information

Bill Dudney

Application Frameworks Evangelist
dudney@apple.com

Documentation

UITableView Class Reference

http://developer.apple.com/library/ios/#documentation/uikit/reference/UITableView_Class

Apple Developer Forums

<http://devforums.apple.com>

Related Sessions

Simplifying Touch Event Handling with Gesture Recognizers

Pacific Heights
Wednesday 3:15PM

Advanced Gesture Recognition

Pacific Heights
Wednesday 4:00PM

