

# Some Nice Things About iOS7

That aren't particularly exciting, but are very useful

# Tinting

- Introduced in iOS 5 on certain UIControls
- Allows you to change the colours of those elements with little code

# Tinting

```
@interface UIView(UIViewRendering)
/*
    -tintColor always returns a color. The color returned is the first non-default
    value in the receiver's superview chain (starting with itself).
    If no non-default value is found, a system-defined color is returned.
    If this view's -tintColorAdjustmentMode returns Dimmed, then the color that is returned
    for -tintColor will automatically be dimmed.
    If your view subclass uses tintColor in its rendering, override -tintColorDidChange
    in order to refresh the rendering if the color changes.
    */
@property(nonatomic,retain) UIColor *tintColor NS_AVAILABLE_IOS(7_0);
```

# What's New in Tinting

- Tinting is now hierarchical
- Tinting a view tints all of its subviews
- Like CSS, one can set tint subviews differently to the parent
- Allows you to make significant changes to the design of an app, with very little effort
- Great when your client keeps changing the designs!

# Demo

```
- (IBAction)tintColorPressed:(id)sender {  
    self.view.tintColor = [UIColor randomColor];  
}
```

# Examples

# Train Times

The screenshot shows a mobile application interface for a 'Journey Planner'. At the top, the status bar displays 'O2-UK', signal strength, time '11:24', location services, Bluetooth, and a 93% battery level. The app header features a back arrow and the title 'Journey Planner'. Below this are two input fields, 'From' and 'To', with a swap button (two arrows) to their right. A tab bar below the inputs has three options: 'Departing' (highlighted in orange), 'Arriving', and 'Last'. Under the 'Departing' tab, the text 'Departing now' is shown next to a clock icon. Below this is a button labeled 'Add return journey'. The bottom of the screen has a white bar with a star icon on the left and the text 'Find Trains' on the right. A dark grey reflection of the bottom bar is visible at the very bottom of the image.

O2-UK 11:24 93%

< Back Journey Planner

From

To

Departing Arriving Last

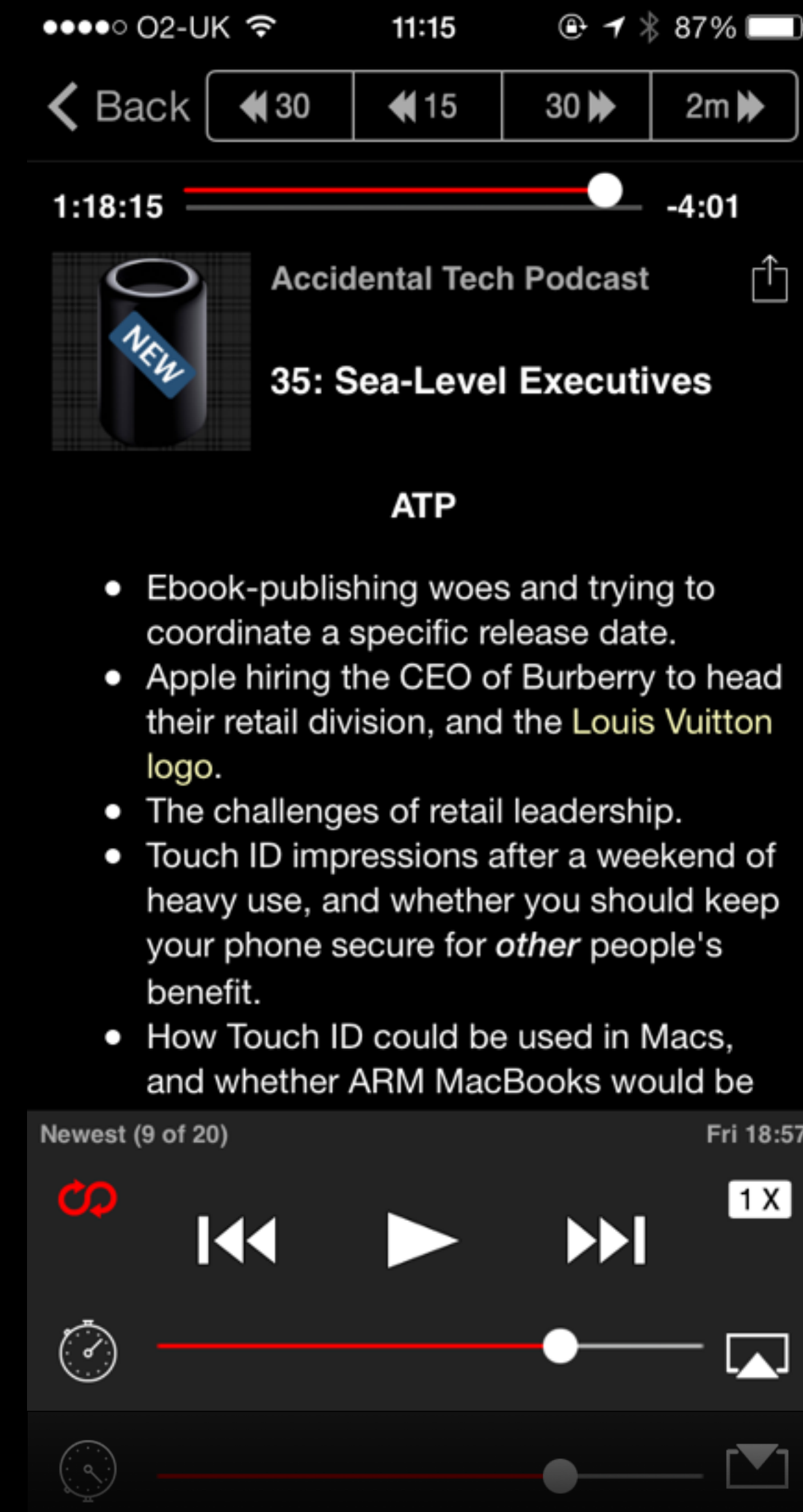
Departing now

Add return journey

Find Trains

# Downcast

```
self.navigationController.tintColor = [UIColor greyColor];  
self.view.tintColor = [UIColor redColor];
```





# Music

```
self.view.tintColor = [UIColor hotPinkColor];  
self.controlContainerView.tintColor = [UIColor blackColor];
```



# It Supports UIAppearance! (sort of)

```
- (void)configureInitialAppearance {  
    // UINavigationController  
    // Background colour  
    [[UINavigationController appearance] setBarTintColor:[UIColor navigationBarColour]];  
    // For the buttons:  
    [[UINavigationController appearance] setTintColor:[UIColor ios7ButtonTintColor]];  
  
    // UIToolbar  
    // Background colour  
    [[UIToolbar appearance] setBarTintColor:[UIColor toolbarColour]];  
    // Buttons  
    [[UIToolbar appearance] setTintColor:[UIColor ios7ButtonTintColor]];  
}
```

# It Supports UIAppearance! (sort of)

```
/* Note for iOS7: On iOS7 the tintColor property has moved to UIView, and now  
has special inherited behavior described in UIView.h.  
    This inherited behavior can conflict with the appearance proxy, and  
therefore tintColor is now disallowed with the appearance proxy.  
*/
```

# It's Easy to Implement in Custom Views

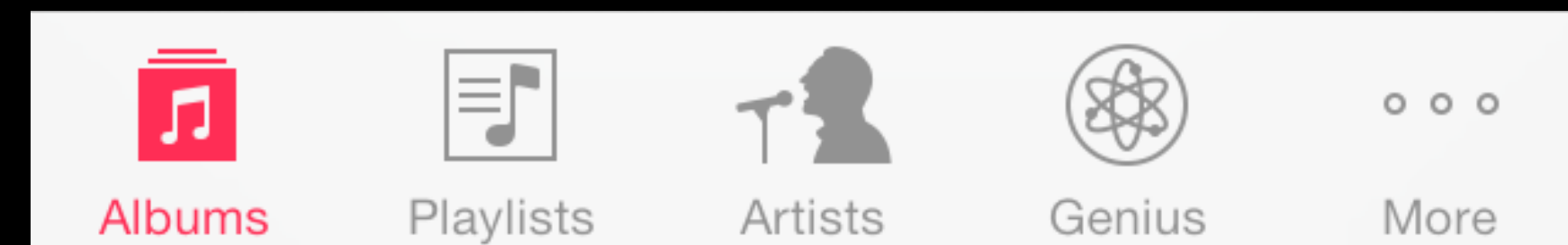
- If your view only contains subviews that already support `tintColor`, you get it for free!
- If your view does custom drawing, override `tintColorDidChange`

# Template Images

Because iOS 7 is All About Flat!

# Template Images

- Takes just the image's alpha mask
- Makes the rest a solid colour
- UITabBar is a good example of this



# Problems

- I have no idea how to use Photoshop
- When I ask a designer for an alpha mask they say “Why do you need that? You have it in the PSD and it’s already the right colour in there. Surely you can just copy that? It won’t look correct if you do it your way and besides I don’t have time to do that right now. If we change the colours in the future, I’ll just send you new assets.”

# Convert Any Image to a Template Image

```
[self.image imageWithRenderingMode:UIImageRenderingModeAlwaysTemplate];
```

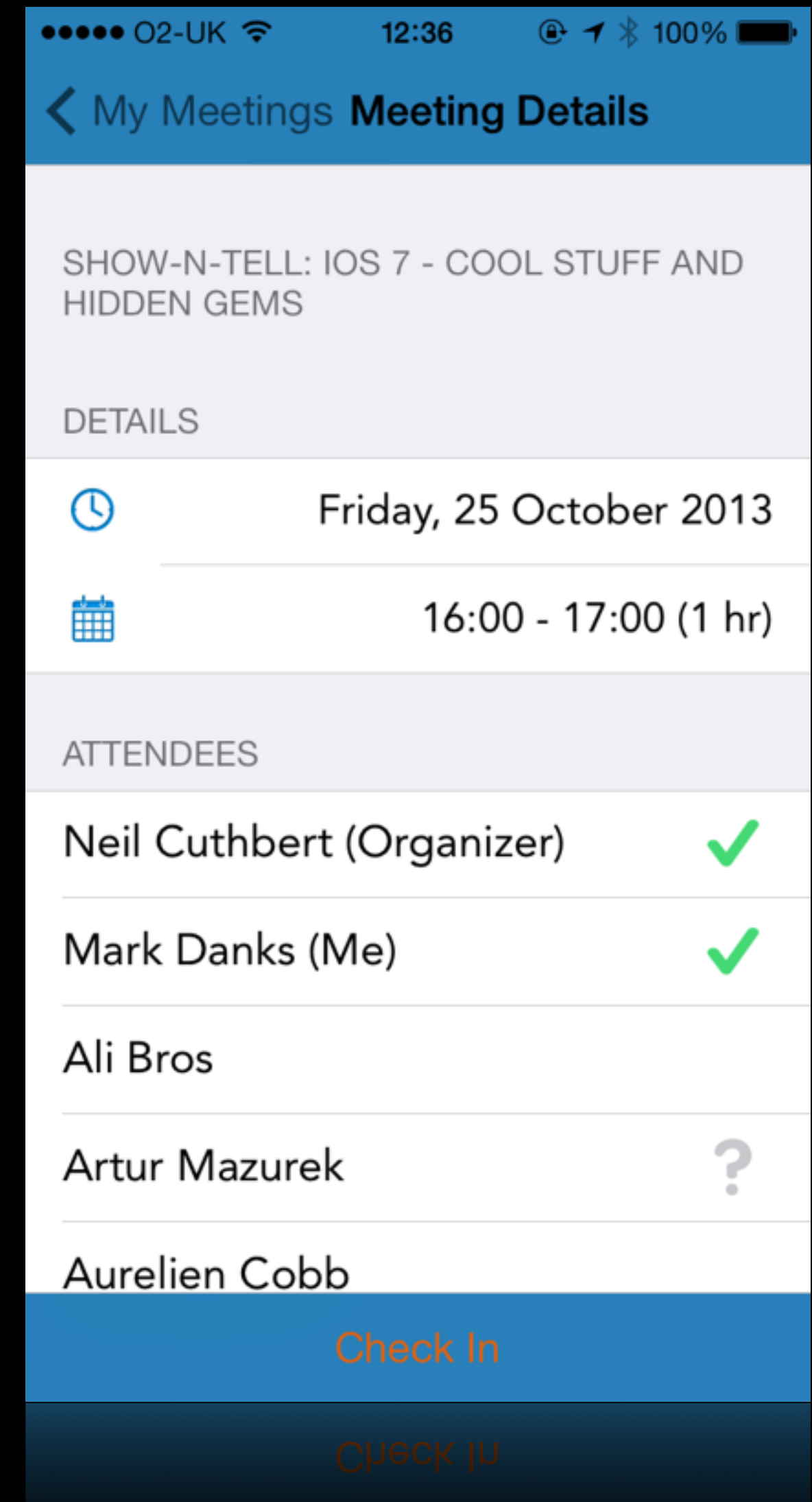
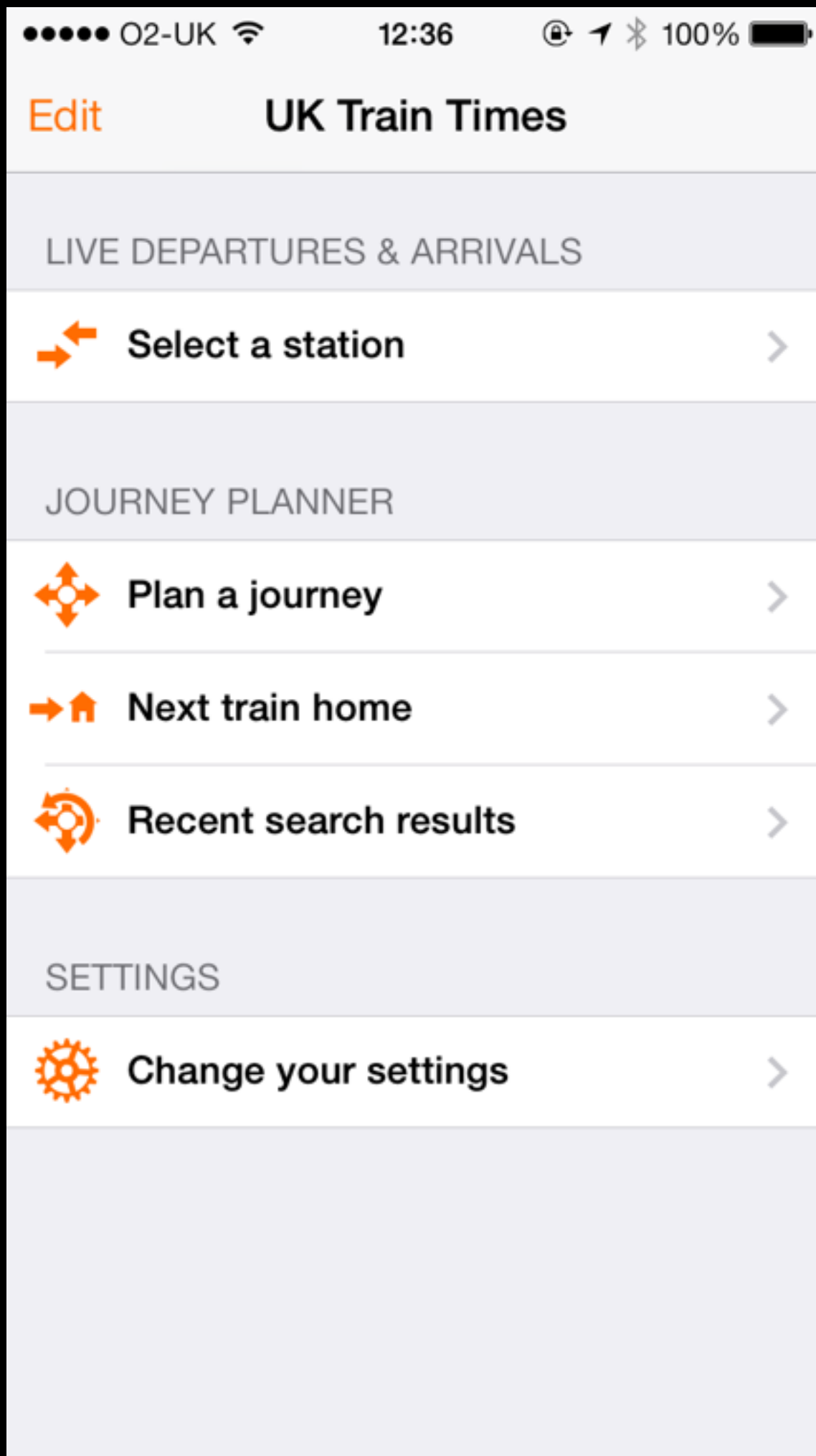


Why would you do  
this?

Because you can tint them!

Demo

# Good for Icons



# UITableView Improvements

# Registering Cells

- Guarantees you'll be given a cell when dequeuing
- The cell you're given will be the correct size for the row it'll be placed into
- Makes working with cells defined in nibs easier
- Also works with headers and footers

# Registering Cells

- (void)registerNib:(UINib \*)nib forCellReuseIdentifier:(NSString \*)identifier;
- (void)registerClass:(Class)cellClass forCellReuseIdentifier:(NSString \*)identifier;

```
[self.tableView registerNib:[UINib nibWithNibName:@"MTInviteCell"  
                                         bundle:nil]
```

# Dequeuing Cells

```
- (id)dequeueReusableCellWithIdentifier:(NSString *)identifier
    forIndexPath:(NSIndexPath *)indexPath

- (UITableViewCell *)tableView:(UITableView *)tableView
    cellForRowAtIndexPath:(NSIndexPath *)indexPath {
    GTLCalendarEvent *event = [self.mutableEvents objectAtIndex:indexPath.row];

    MTInviteCell *cell = [tableView dequeueReusableCellWithIdentifier:kInviteReuseIdentifier
        forIndexPath:indexPath];

    cell.delegate = self;
    cell.event = event;
    return cell;
}
```

# Estimating Row Heights

- When implementing cells with varying height, performance suffers if the calculation for the height is slow.
- As `tableView:heightForRowAtIndexPath:` has to be called for every cell, before the tableView can even be displayed
- New methods let you defer that calculation until the cell is about to be displayed



Demo

# Implementation

```
-(CGFloat)tableView:(UITableView *)tableView  
estimatedHeightForRowAtIndexPath:(NSIndexPath *)indexPath;
```

Text Styles

# Text Styles

- iOS 7 introduces user customisable text size

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

## **Lorem ipsum**

Lorem ipsum

# Lorem ipsum

# Lorem ipsum

# Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

**Lorem ipsum**

**Lorem ipsum**

# Lorem ipsum

# Lorem ipsum

# Lorem ipsum

# Lorem ipsum

# Lorem ipsum

# Lorem ipsum

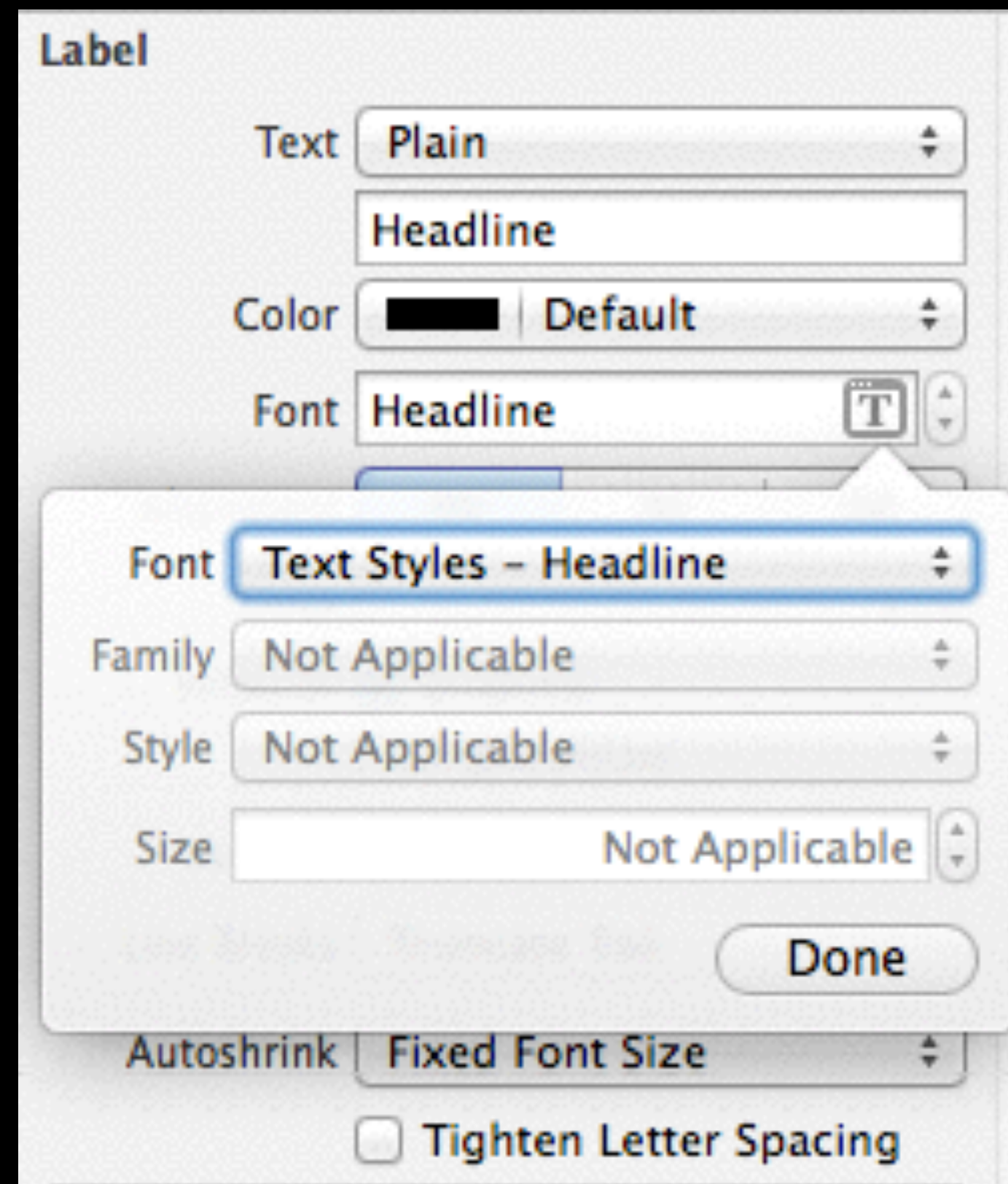
# Lorem ipsum

# Lorem ipsum

# Lorem ipsum

# Applying Text Styles

```
+ (UIFont *)preferredFontForTextStyle:(NSString *)style
    UIFontTextStyleHeadline
    UIFontTextStyleBody
    UIFontTextStyleSubheadline
    UIFontTextStyleFootnote
    UIFontTextStyleCaption1
    UIFontTextStyleCaption2
```



Demo

# Keeping in Sync

- The user can change their preferences at any time, and you need to react to that.

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view.

    [[NSNotificationCenter defaultCenter] addObserver:self
                                             selector:@selector(fontSizeChangedNotification:)
                                             name:UIContentSizeCategoryDidChangeNotification
                                             object:nil];
}

-(void)fontSizeChangedNotification:(NSNotification*)notification{
    // Annoyingly, font styles in nibs don't refresh automatically
    self.headline.font = [UIFont preferredFontForTextStyle:UIFontTextStyleHeadline];
    self.subheadline.font = [UIFont preferredFontForTextStyle:UIFontTextStyleSubheadline];
    self.body.font = [UIFont preferredFontForTextStyle:UIFontTextStyleBody];
    self.caption1.font = [UIFont preferredFontForTextStyle:UIFontTextStyleCaption1];
    self.caption2.font = [UIFont preferredFontForTextStyle:UIFontTextStyleCaption2];
    self.footnote.font = [UIFont preferredFontForTextStyle:UIFontTextStyleFootnote];

    [self.labels makeObjectsPerformSelector:@selector(invalidateIntrinsicContentSize)];
}
```

# Other Resources

- <http://nshipster.com/ios7/>
- <http://www.objc.io/issue-5/>
- The HIG