

Table Views in iOS

Hands-On Challenges

Table Views: Beginning to Advanced Hands-On Challenges

Copyright © 2014 Razeware LLC.

All rights reserved. No part of this book or corresponding materials (such as text, images, or source code) may be reproduced or distributed by any means without prior written permission of the copyright owner.

This book and all corresponding materials (such as source code) are provided on an "as is" basis, without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose, and noninfringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in action of contract, tort or otherwise, arising from, out of or in connection with the software or the use or other dealings in the software.

All trademarks and registered trademarks appearing in this book are the property of their respective owners.



Challenge #7: Static Cells

You can add, delete, and move bugs, but your app is currently lacking one critical feature: editing bugs!

Your challenge this time is to add an editing view into the app. You should create the editing view using a table view controller with static cells, like this:



See if you can do this on your own based on what you learned on the video. If you get stuck, follow along with the full walkthrough below!

Full Walkthrough

Note: You're getting more advanced at this point, so this time I am not listing out each and every step so it's a bit more of a challenge. If you get stuck, refer back to the video or check out the challenge solution. Good luck!

Open the Scary Bugs project where you left it off in the last challenge, or use the starter project provided by the instructor.



Open **Main.storyboard** and drag a new table view controller into the canvas. Select the table view and set the **Content** to **Static Cells**, the **Sections** to **2**, and the **Style** to **Grouped**.

Then lay out your static cells so they look something like this:



Select the two cells under “The Rest” and in the Attributes Inspector (3rd tab) set the **Selection** to **None**.

Next, create a new class named **EditViewController** that derives from **UITableViewController**. Switch back to **Main.storyboard**, and set the class of the view controller to **EditViewController** in the identity inspector.

Using the assistant editor, connect the image view to an outlet in **EditViewController.m** named **bugImageView**, the name text field to an outlet named **bugNameTextField**, and the rating label to an outlet named **bugRatingLabel**.

Finally, set the `EditViewController` as the delegate of the name text field.

Next open **EditViewController.h** and predeclare the `ScaryBug` class before the `@interface`:

```
@class ScaryBug;
```

Also add a new property for the bug to edit between the `@interface` and the `@end`:

```
@property (strong, nonatomic) ScaryBug *bug;
```



Switch to **EditViewController.m** import the ScaryBug header:

```
#import "ScaryBug.h"
```

Mark the interface as implementing these protocols:

```
@interface EditViewController () <UITextFieldDelegate,  
UIImagePickerControllerDelegate, UINavigationControllerDelegate>
```

Then replace everything between the @implementation and the @end with the following:

```
- (void)viewWillAppear:(BOOL)animated {  
    [super viewWillAppear:animated];  
    self.bugImageView.image = self.bug.image;  
    self.bugNameTextField.text = self.bug.name;  
    self.bugRatingLabel.text = [self.bug howScaryString];  
}  
  
- (void)viewWillDisappear:(BOOL)animated {  
    [super viewWillDisappear:animated];  
    self.bug.image = self.bugImageView.image;  
    self.bug.name = self.bugNameTextField.text;  
}  
  
- (void)tableView:(UITableView *)tableView  
didSelectRowAtIndexPath:(NSIndexPath *)indexPath {  
  
    if (indexPath.row == 0 && indexPath.section == 0) {  
  
        [self.tableView deselectRowAtIndexPath:indexPath animated:YES];  
  
        UIImagePickerController *picker =  
            [[UIImagePickerController alloc] init];  
        picker.sourceType = UIImagePickerControllerSourceTypePhotoLibrary;  
        picker.allowsEditing = NO;  
        picker.delegate = self;  
        [self presentViewController:picker animated:YES completion:nil];  
  
    }  
}  
  
- (BOOL)textFieldShouldReturn:(UITextField *)textField {  
    [textField resignFirstResponder];  
    return YES;  
}
```



```

- (void)imagePickerController:(UIImagePickerController *)picker
  didFinishPickingMediaWithInfo:(NSDictionary *)info {
    UIImage *image = info[UIImagePickerControllerOriginalImage];
    self.bug.image = image;
    self.bugImageView.image = image;
    [self dismissViewControllerAnimated:YES completion:nil];
}

```

This code is very similar to the code from the lecture so I will not re-explain it here.

That's it for your new view controller; you just need to configure the bug list view controller to display it. To do this, open **Main.storyboard** and control-drag from your custom bug cell to your new view controller. Choose **push**, and name the segue **GoToEdit**.

Then open **BugTableViewController.m** and add this import to the top of the file:

```
#import "EditTableViewController.h"
```

And add these new methods to the bottom:

```

- (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender {
    if ([segue.identifier isEqualToString:@"GoToEdit"]) {
        EditViewController *edit =
            (EditViewController *)segue.destinationViewController;
        NSIndexPath *indexPath = [self.tableView indexPathForCell:sender];
        BugSection *section = self.bugSections[indexPath.section];
        ScaryBug *bug = section.bugs[indexPath.row];
        edit.bug = bug;
    }
}

- (void)viewWillAppear:(BOOL)animated {
    [self.tableView reloadData];
}

```

Build and run, and you should now be able to edit bugs!



