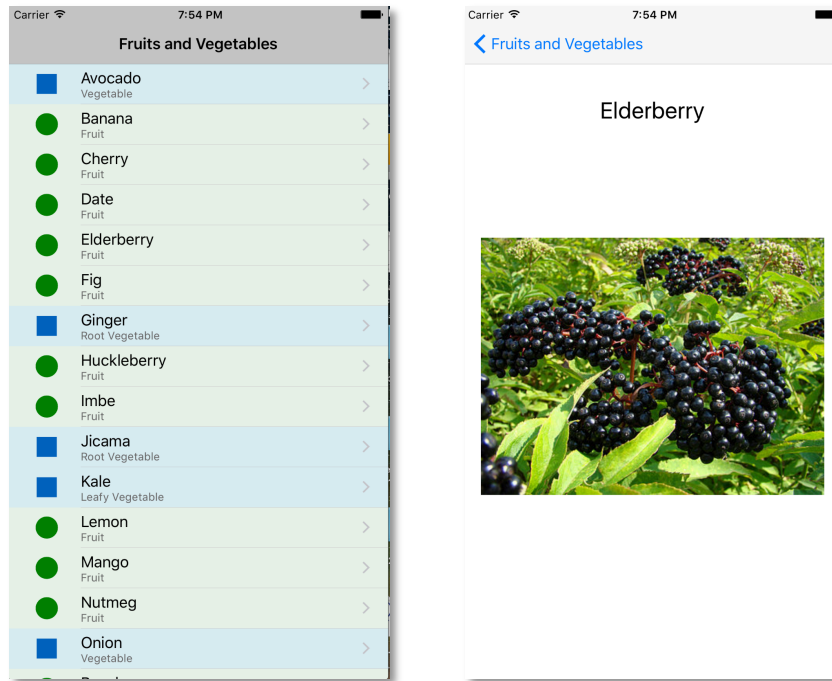


Fruits and Vegetables

An easy introduction to Table View and Navigation Controller Part 1: Table View and Data Model



Introduction

The table view presents a scrolling list of data. You can embed the table view into a navigation controller to segue to another view without having to write a lot of code. The table view has several options, including a disclosure mark (the > you see in the image above), a subtitle that can be placed to the left, to the right, or underneath; an image; and you can color the table cells and the text.

Materials

- See the zip file with many fruit and vegetable images (fruit_images.zip)

Instructions

1. Create a new Xcode project: iOS, Single View application, etc.
 - a. Turn OFF the check boxes for Core Data, Unit Tests, and UI Test. They aren't necessary in this project.
2. Using the Object Library, drag a new Table View Controller into the canvas.
 - a. You may see a warning indicating that although view controllers exist in the storyboard, the table view controller scene is unreachable because no initial view controller has been specified.

3. Using Interface Builder, select the table view controller, open the Attributes Inspector, ensure that the **Is Initial View Controller** attribute is checked, and observe how Interface Builder displays an arrow to the left of the table view controller scene.
4. Run the app, observe the empty table cells, and click and drag to interact with the table.
5. Add a new Cocoa Touch Class to the project called FruitTableViewController and specify a Subclass of UITableViewController.
6. Using Interface Builder, select the table view controller, and use the Identity Inspector to change the Class attribute to FruitTableViewController.
7. Run the app, observe the empty table rows, and note the warning raised by Xcode.
 - a. Note the "prototype cells must have reuse identifier" warning raised by Xcode.
8. Using Interface Builder, select the prototype cell in the table view controller, open the Attributes Inspector, set the Identifier attribute to **FruitCell**, and observe the Xcode warning disappear.
9. The value **FruitCell** will serve as the cell reuse identifier when the table view displays the cell for a table row.
10. Download the fruit_images.zip file. Drag all the images in all the folders into the Assets section of your app.
11. At the top of the FruitTableViewController class, type this code:

```
class FruitTableViewController: UITableViewController {
    var fruits = [
        "Avocado", "Banana", "Cherry", "Date",
        "Elderberry", "Fig", "Ginger", "Huckleberry",
        "Imbe", "Jicama", "Kale", "Lemon",
        "Mango", "Nutmeg", "Onion", "Peach",
        "Quandong", "Raspberry", "Spinach", "Tomato",
        "Ugni", "Vanilla", "Watermelon", "Yam", "Zucchini"
    ];

    var vegetables = [ 0, 6, 9, 10, 14, 18, 19, 23, 24 ];
    var roots = [ 6, 9, 23 ];
    var leafy = [ 10, 18 ];

    override func viewDidLoad() {
```

12. The **numberOfSections** should return the number of sections in the Table View. In the future, you might write other apps that will use more than one section. In the Fruits and Vegetables app, there will be only one section.

```
    override func numberOfSections(in tableView: UITableView) -> Int {
        // #warning Incomplete implementation, return the number of sections
        return 1
    }
```

13. The data model should determine the **numberOfRowsInSection**. In this case, the data model is the fruits array we declared above.

```
override func tableView(_ tableView: UITableView,
    numberOfRowsInSection section: Int) -> Int {
    // #warning Incomplete implementation, return the number of rows
    return fruits.count
}
```

14. Uncomment and update the boilerplate tableView:cellForRowAt:IndexPath: method implementation.

```
override func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {
    let cell = tableView.dequeueReusableCell(
        withIdentifier: "FruitCell", for: indexPath)

    // Configure the cell...
    cell.textLabel?.text = fruits[indexPath.row]
    return cell
}
```

15. Run the app, observe the table cells with a list of fruits and vegetables.

- a. Observe that Xcode does not report any warnings.
- b. The list should scroll.
- c. When you click or drag, nothing will happen, because we did not set up any navigation or segues yet.