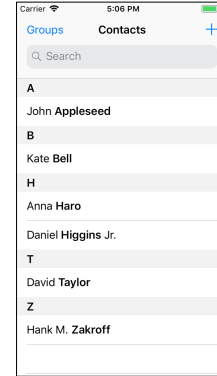


CS 326 Tables

Stephen Freund

UITableView

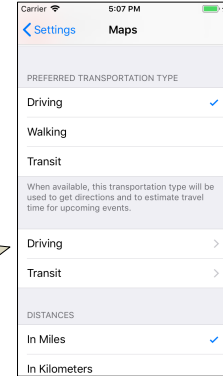
UITableViewStyle.plain



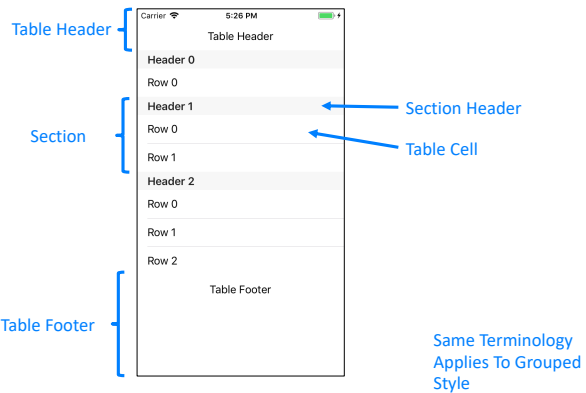
Best For
Dynamic
Content

Best For
Static
Content

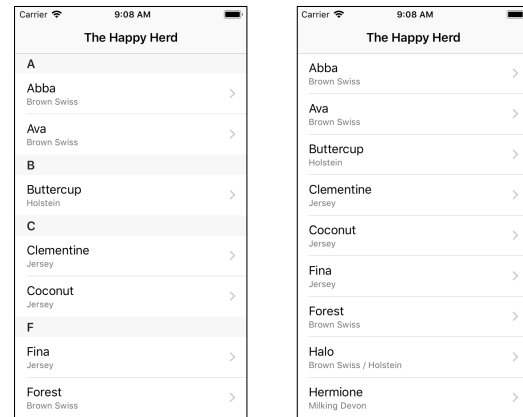
UITableViewStyle.grouped



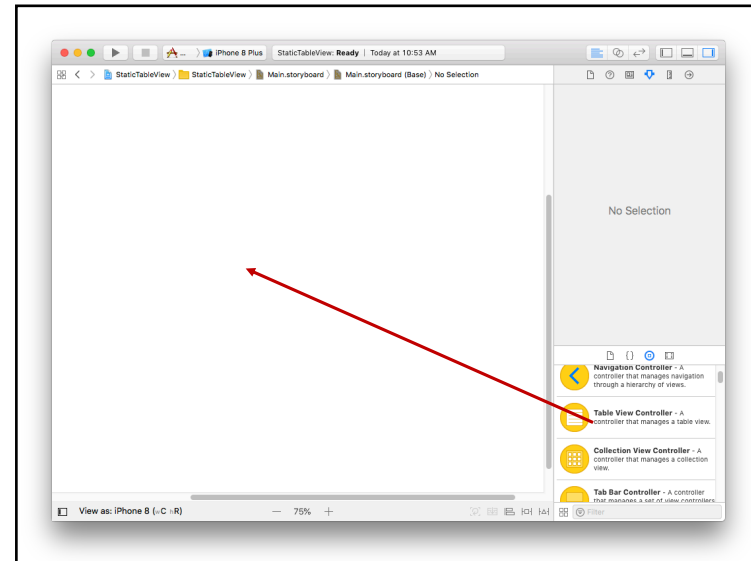
UITableView Plain Style



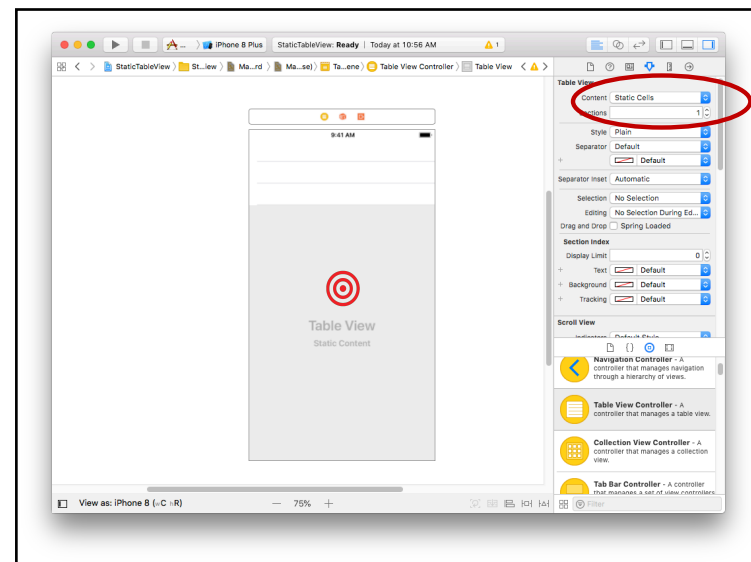
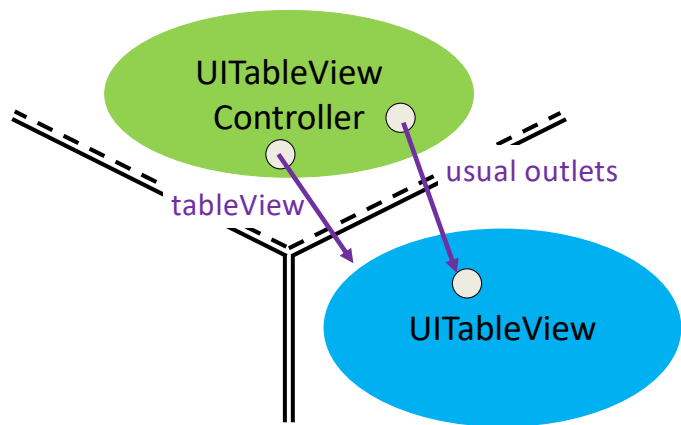
Section Headings vs No Headings

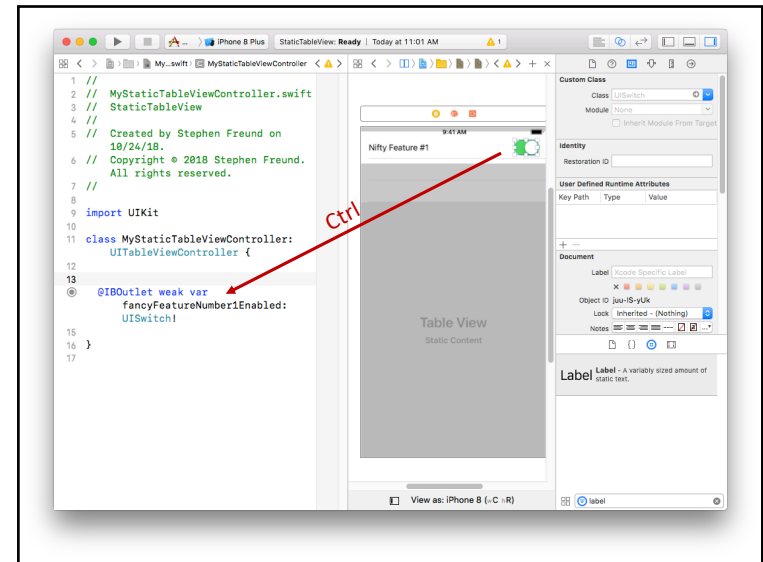
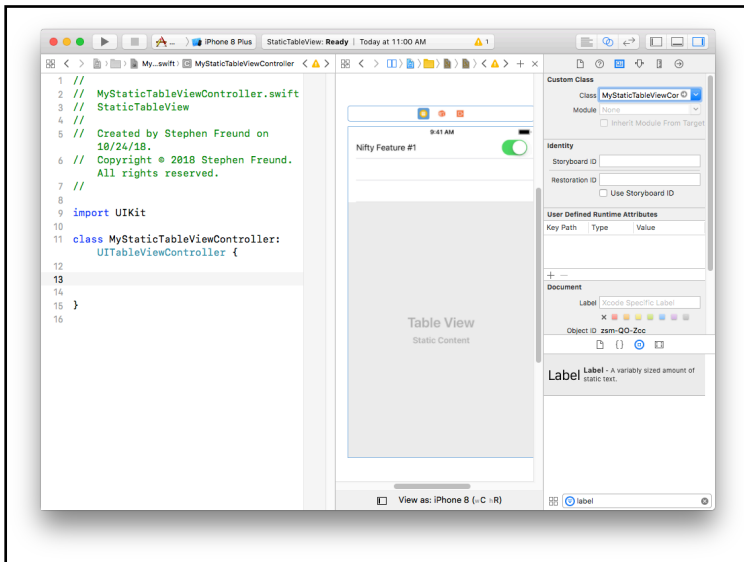
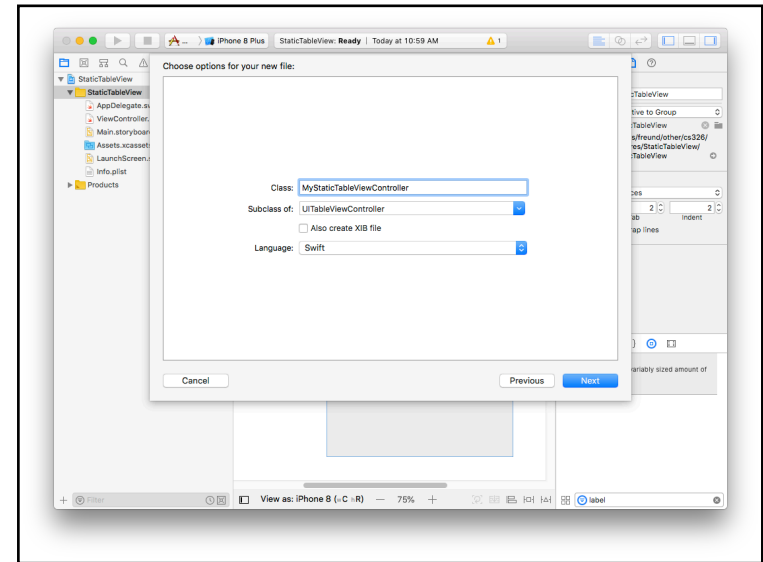
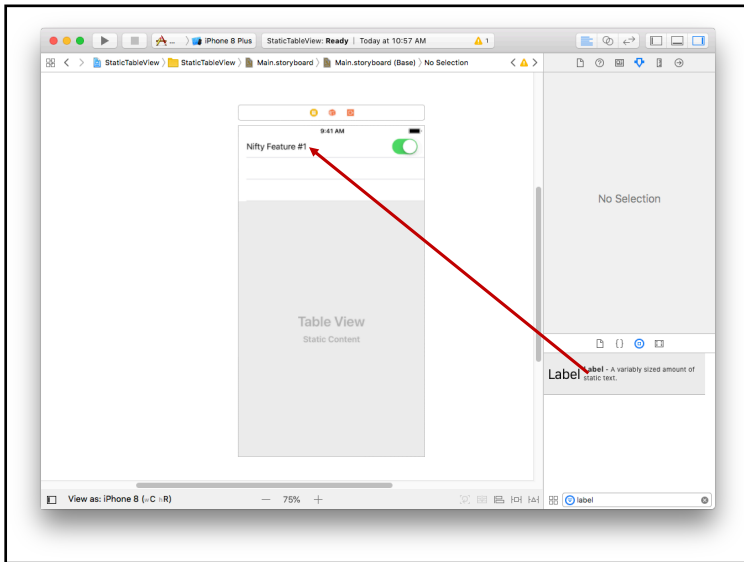


Static Tables

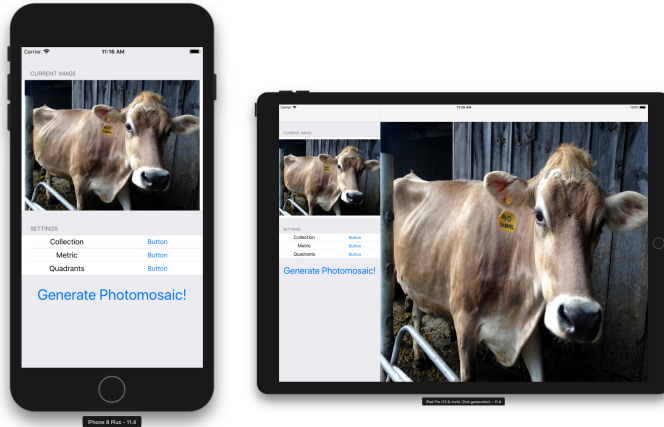


UITableViewController





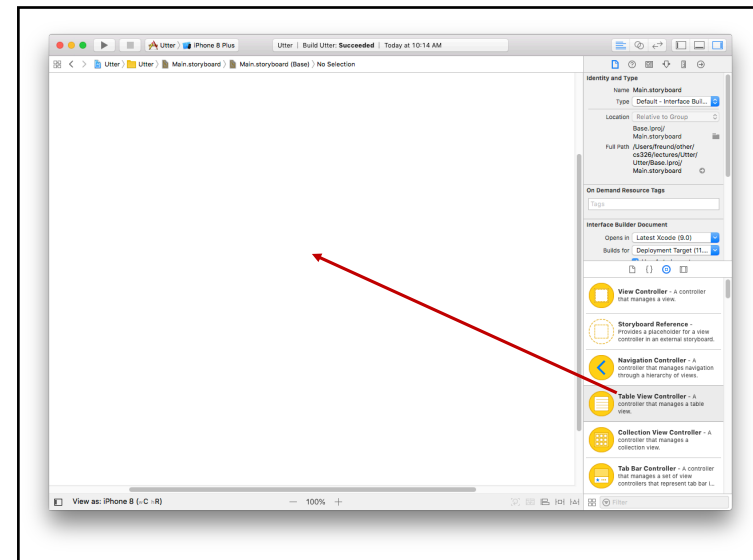
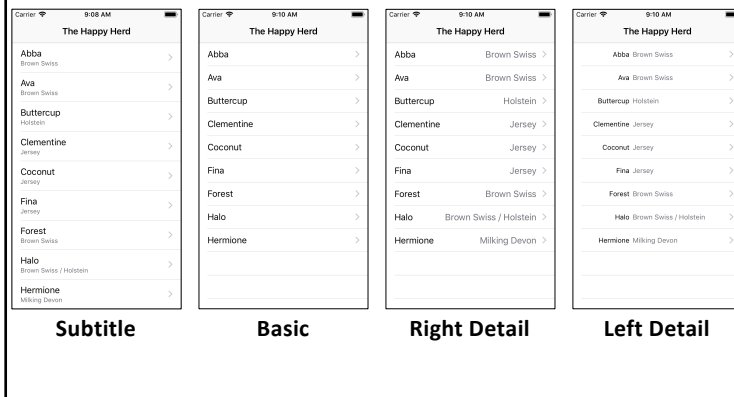
Feel Free to Use Static Table For Photomosaics!

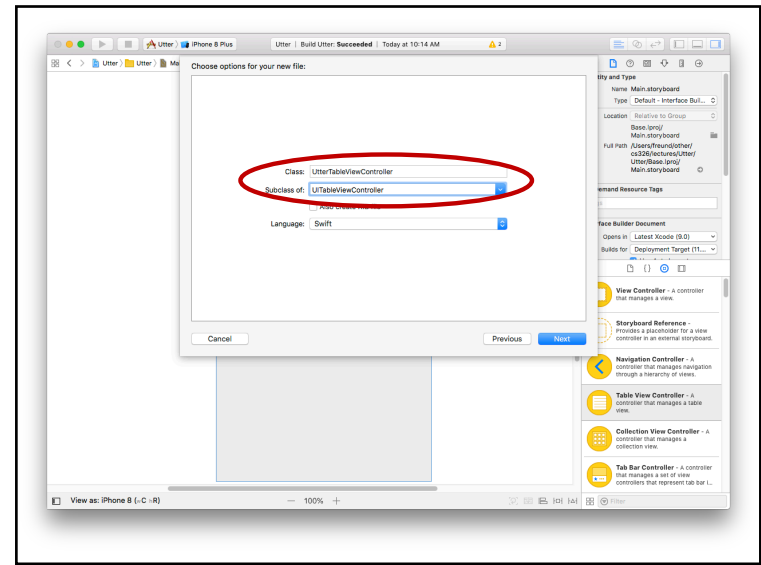
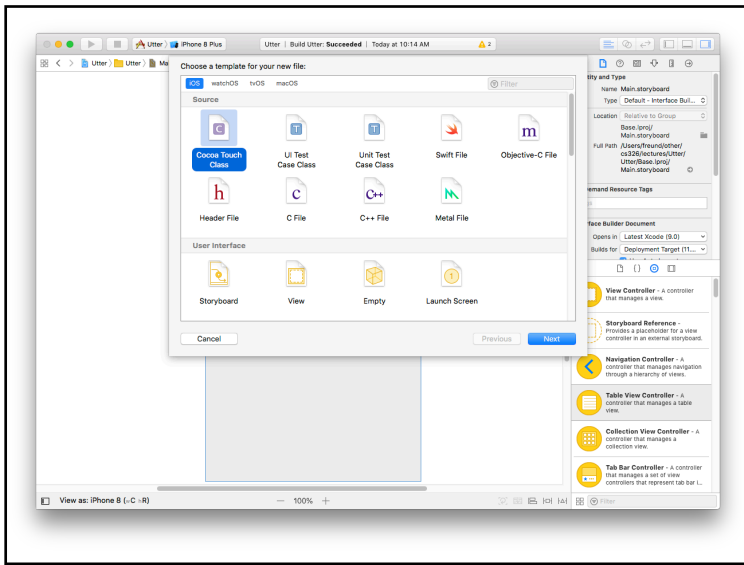
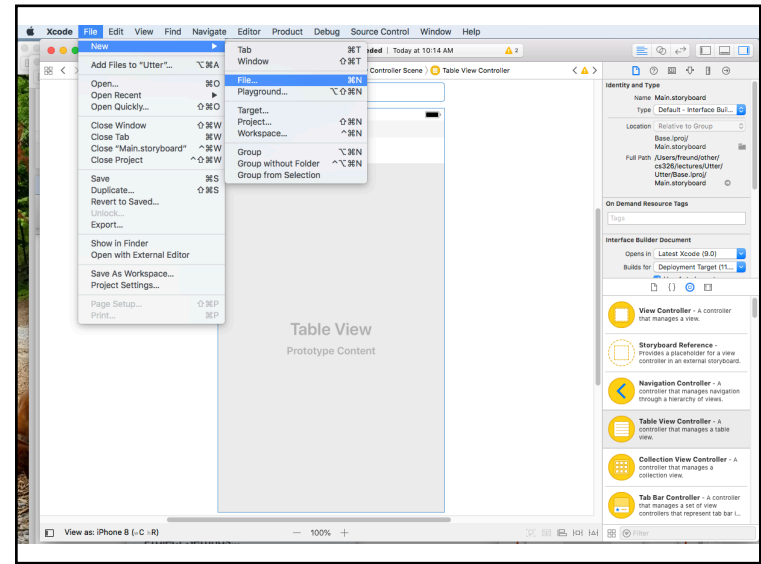
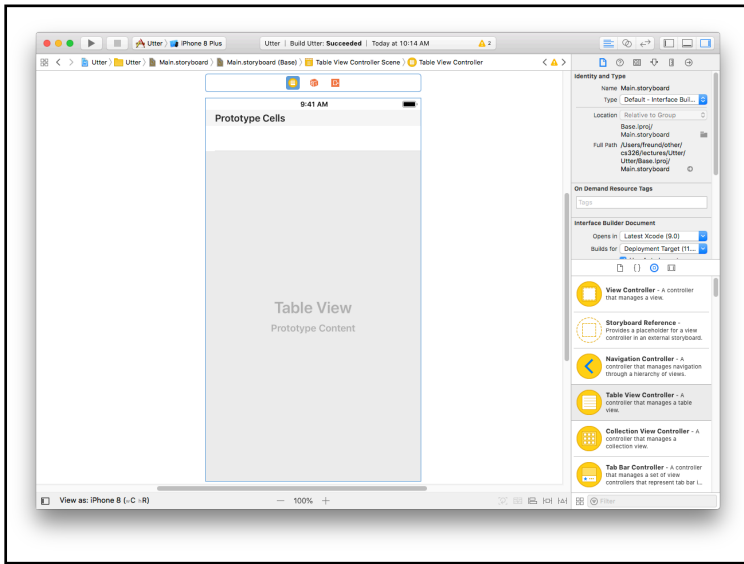


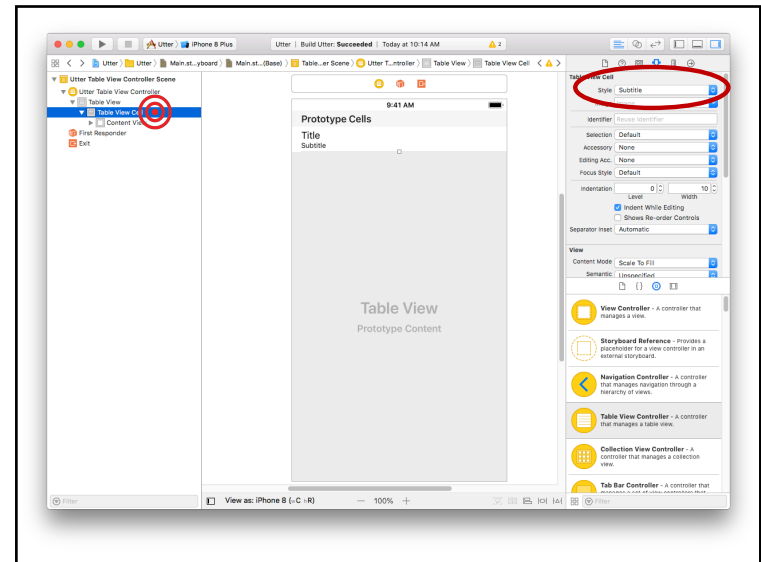
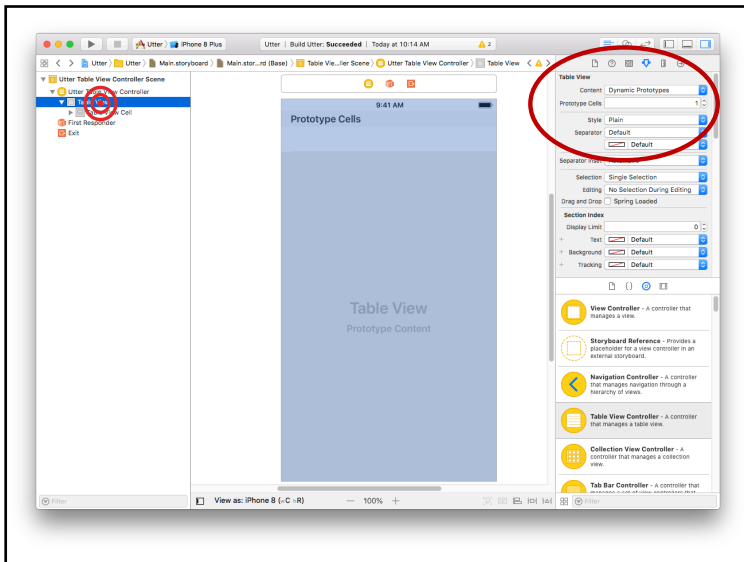
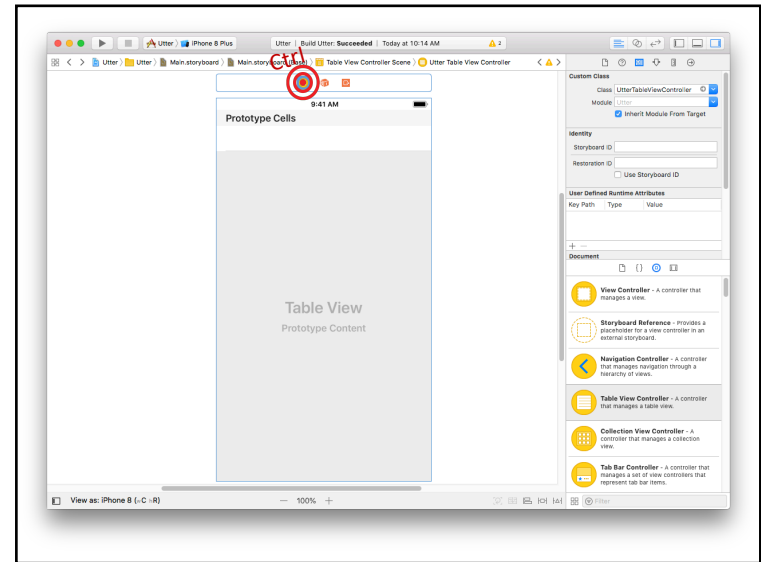
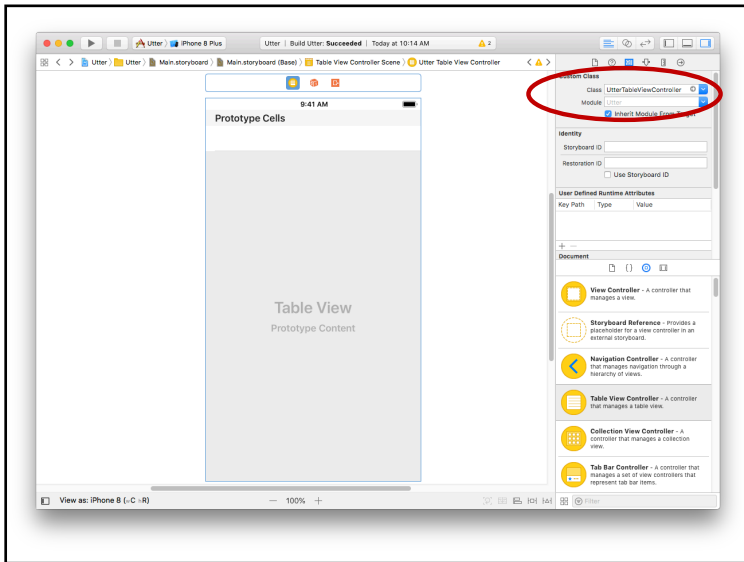
Dynamic Tables

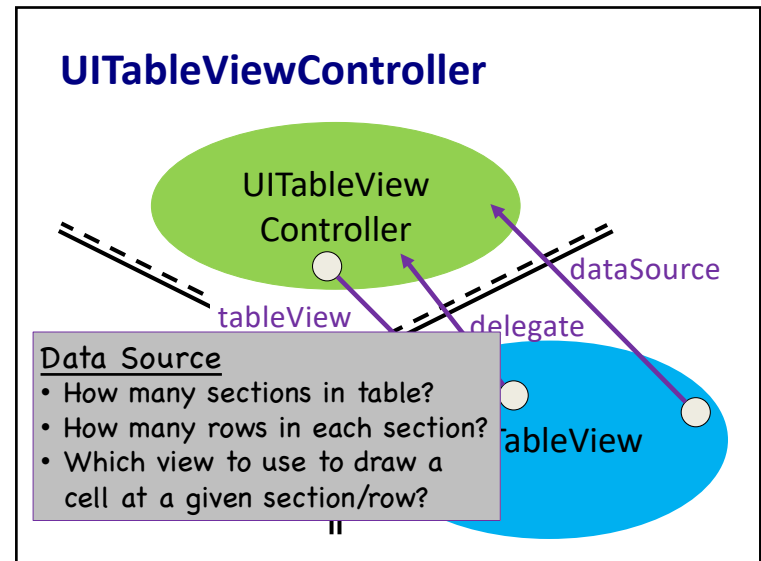
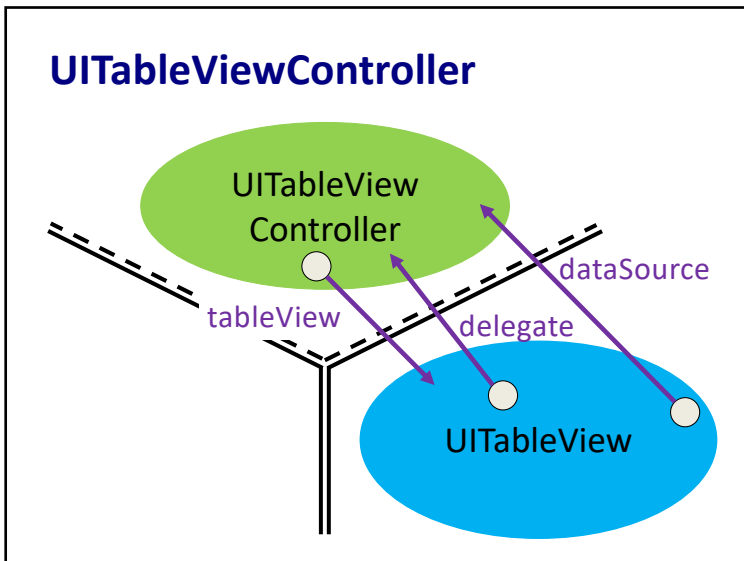
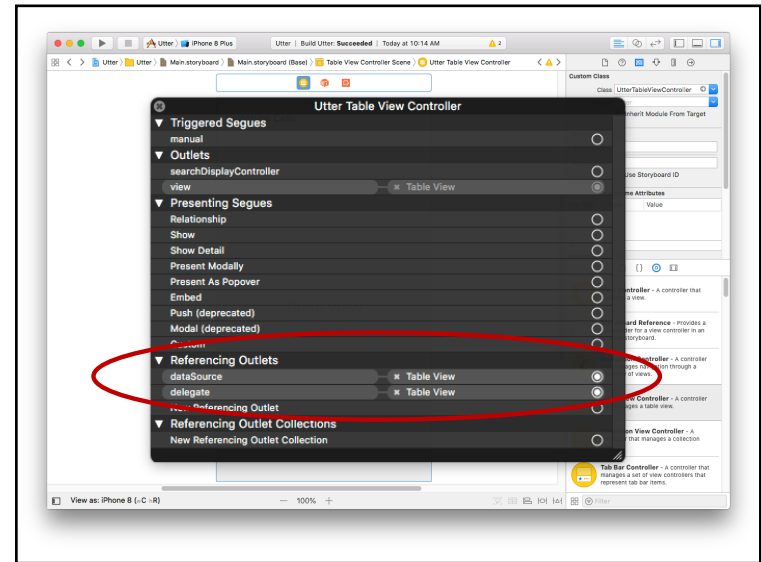
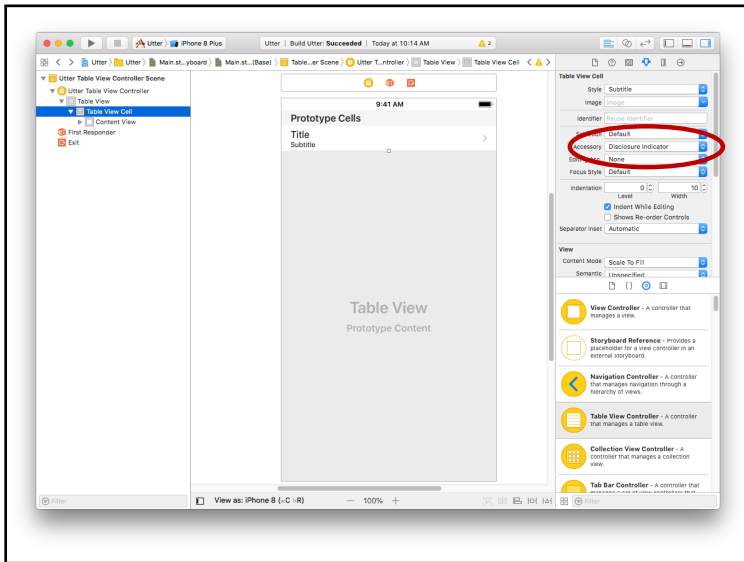
- Show pile of data as a table...
- Two parts:
 - View: UITableViewCells to present data
 - Delegates and Data Sources: how to populate table with data and interact with it.

Cell Types in Plain Style









DataSource Delegate Methods

- UITableView calls this delegate method to get number of sections:
 - `numberOfSections(in tableView: UITableView) -> Int`
- UITableView calls this delegate method to get number of rows in a section:
 - `tableView(_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int`
- Optional, if sections have headers:
 - `tableView(_ tableView: UITableView, titleForHeaderInSection: Int) -> String?`

DataSource Delegate Methods

- UITableView calls this delegate method to get a view to show a specific table row:

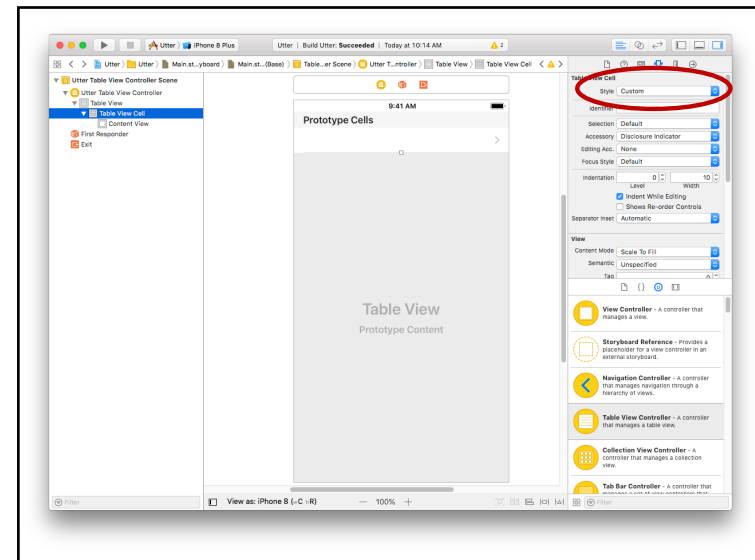
```
protocol UITableViewDataSource {  
    func tableView(_ tableView: UITableView,  
                  cellForRowAt: IndexPath) -> UITableViewCell  
}
```

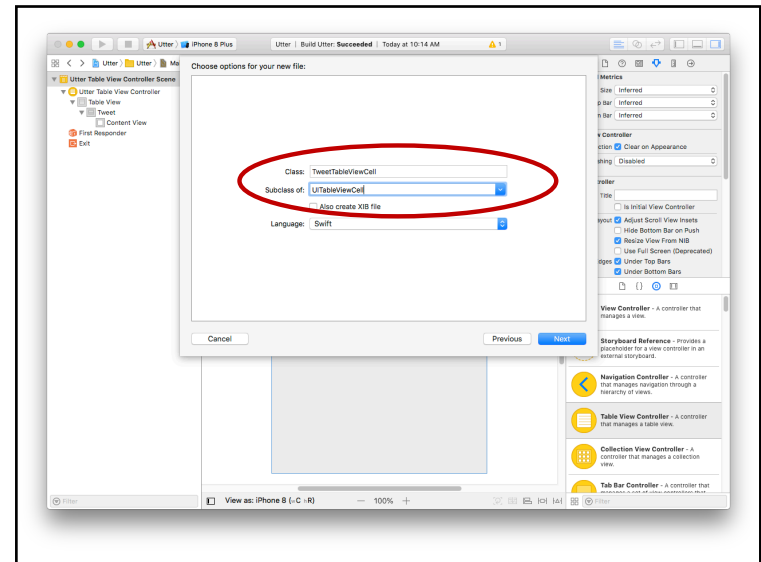
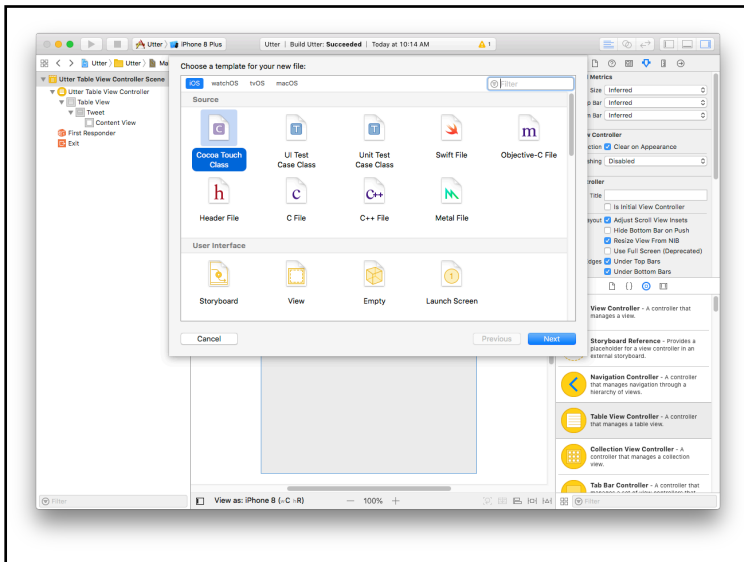
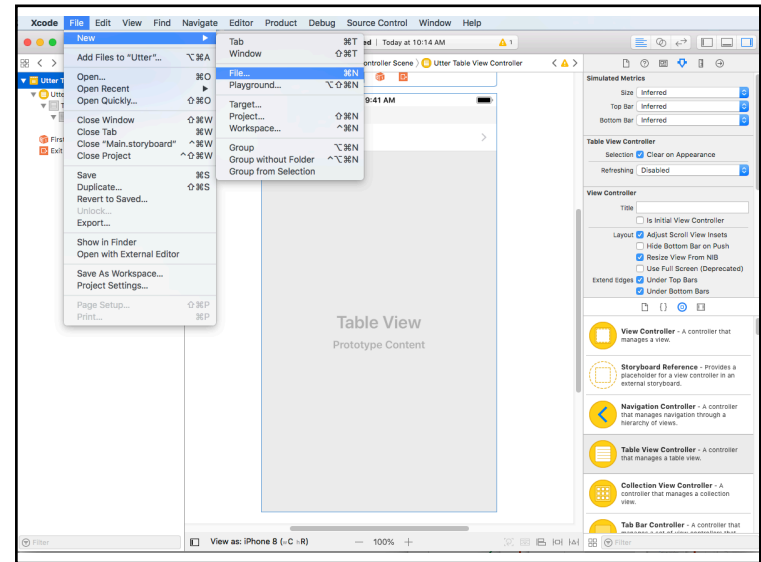
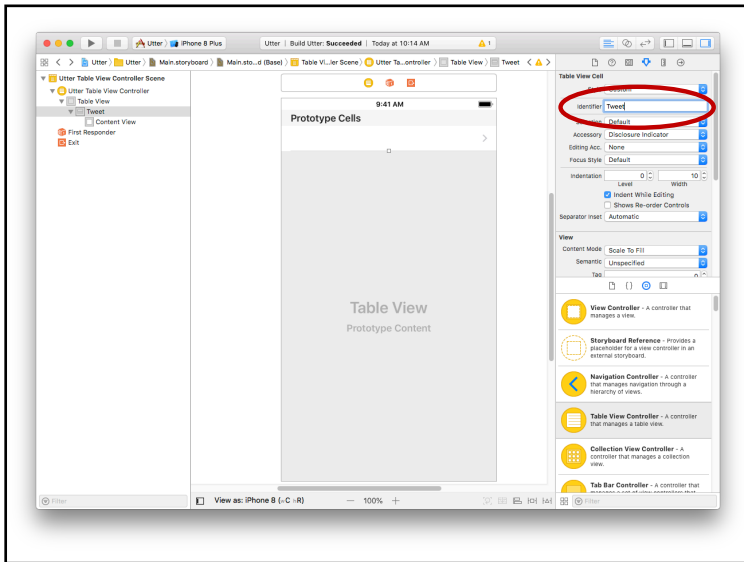
struct with section and row number

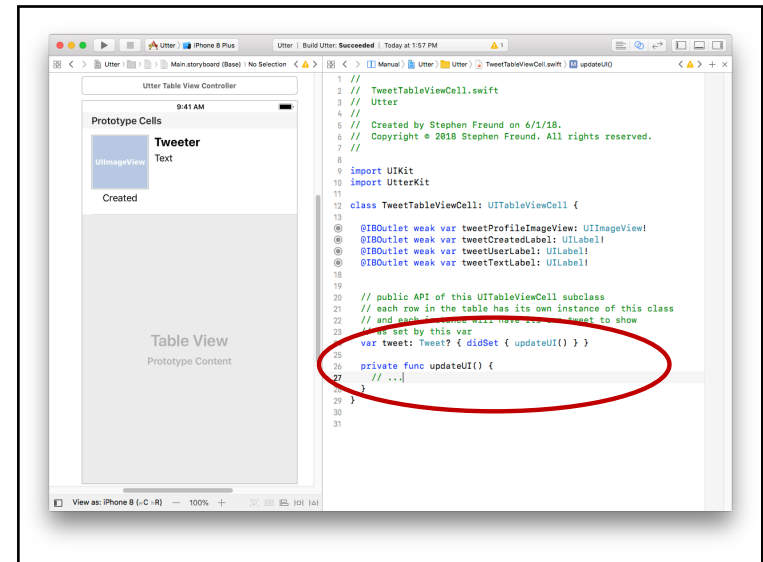
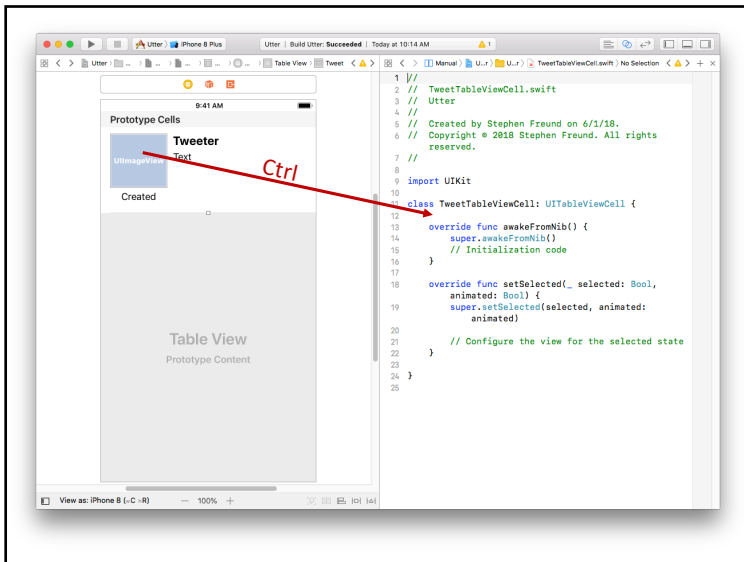
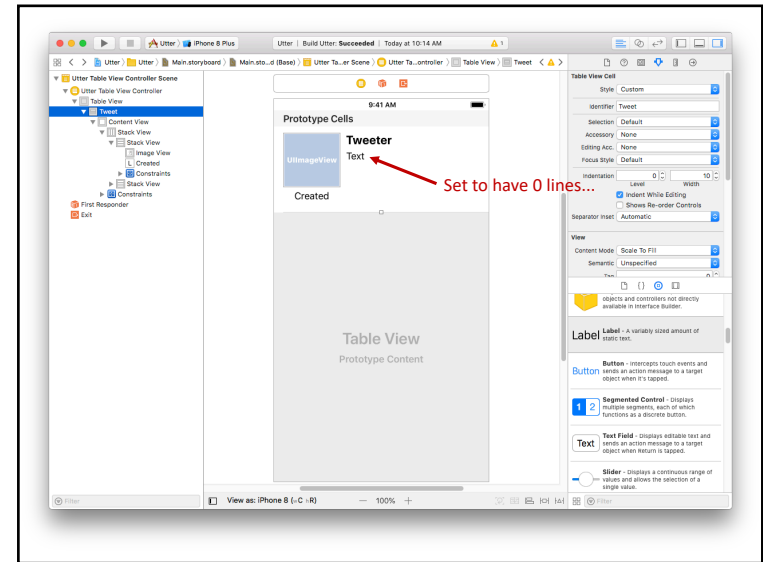
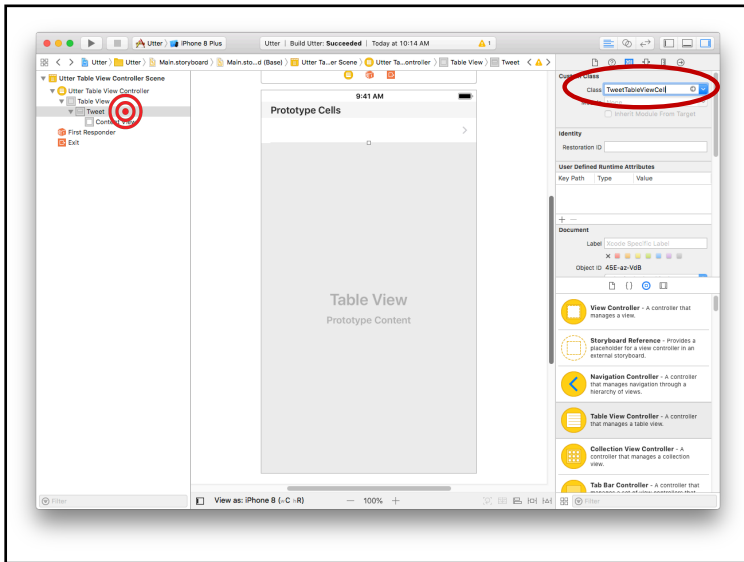
- Reuse cells, only assign to rows that are showing
- **Warning:** reuse means we have to be careful with multithreading – cell may be re-purposed while off running something on global queue.

Cells for Pre-Defined Styles

```
func tableView(_ tableView: UITableView,  
              cellForRowAt indexPath: IndexPath) -> UITableViewCell {  
    let cell =  
        tableView.dequeueReusableCell(withIdentifier: "Tweet",  
                                    for: indexPath)  
  
    let tweet = tweets[indexPath.section][indexPath.row]  
    cell.textLabel?.text = tweet.user.name  
    cell.detailTextLabel?.text = tweet.text  
  
    return cell  
}
```







A Few Details About Cells

- You can have multiple prototypes if table has different kinds of rows
 - each has unique identifier
- Use AutoLayout for everything in prototype cells so table can figure out how tall each row is.
- Create subclass of UITableViewCell to contain outlets for Prototype Cells...
 - only place outside of Controller you have outlets.

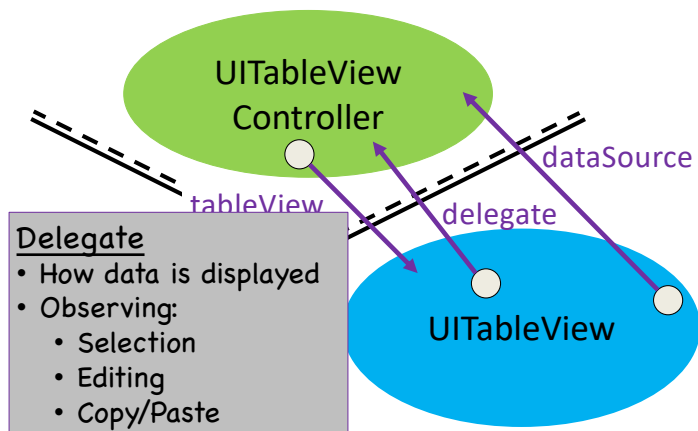
Cells for Custom Style

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

    if let tweetCell = cell as? TweetTableViewCell {
        let tweet = tweets[indexPath.section][indexPath.row]
        tweetCell.tweet = tweet
    }

    return cell
}
```

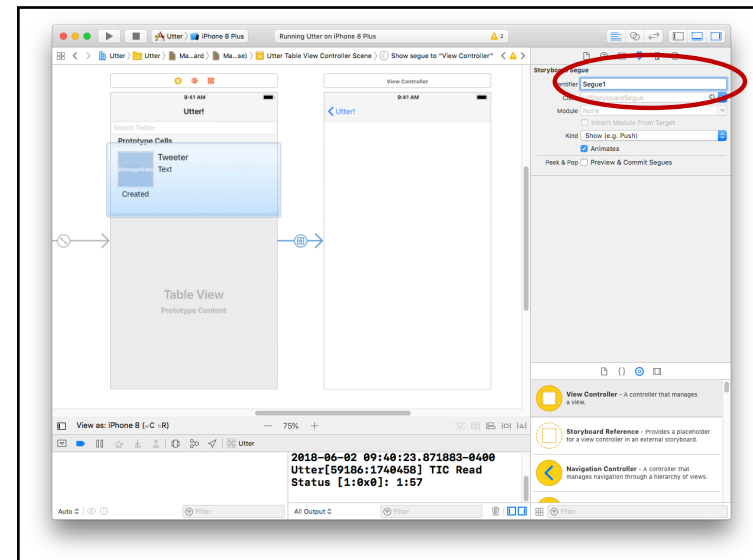
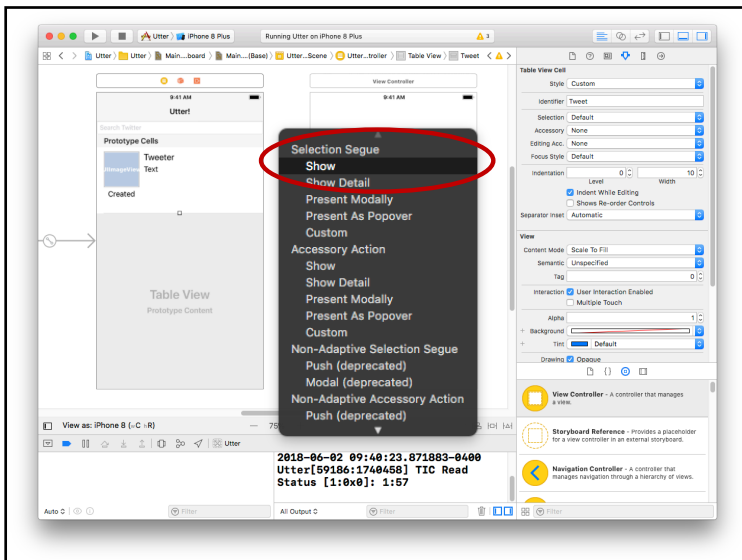
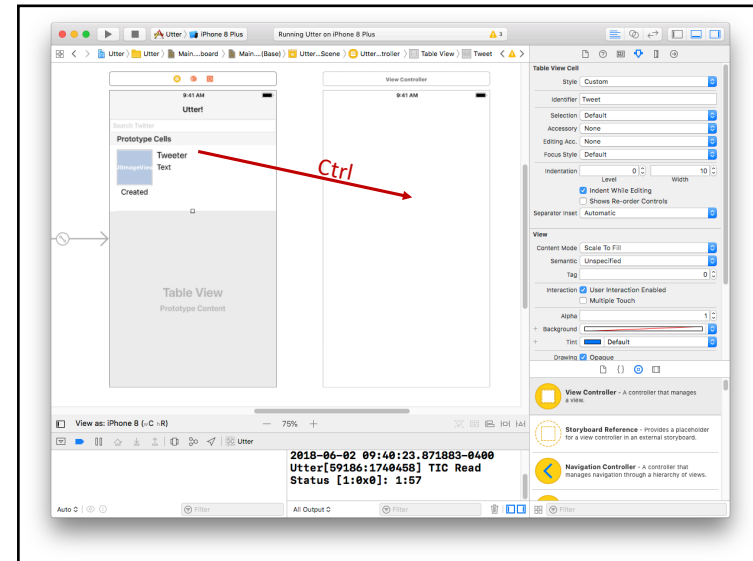
UITableViewController



Important Details of UITableView

- Controller tells table when data has changed
 - tableView.reloadData()
 - tableView.insertSections([...], with: .fade)
- Controller sets the height of each row
 - tableView.rowHeight
 - for automatic row height:
 - tableView.rowHeight = UITableViewAutomaticDimension
 - tableView.estimatedRowHeight = ...
- Many other methods: See Documentation!

Segues From Tables



Segues From TableViewController

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {  
    if let identifier = segue.identifier {  
        switch identifier {  
            case "Segue1":  
                if let cell = sender as? TweetTableViewCell,  
                    let indexPath = tableView.indexPath(for: cell),  
                    let destination = segue.destination as? MyVC {  
                    // prepare dest as usual here...  
                    destination.property =  
                        data[indexPath.section][indexPath.row]  
                }  
            case "Segue2": ...  
            default: break  
        }  
    }  
}
```

Search Text Field

