GENERAL ASSEMBLY

TABLE VIEWS

Zeke Abuhoff

Lead iOS Instructor, General Assembly

LEARNING OBJECTIVES

+ Override table view methods
+ Deploy prototype cells to show data dynamically
+ Connect table views to view controllers via the delegate pattern

TABLE VIEWS

DYNAMIC DISPLAY

Individual Objects (let name = “Plankton”)	Collections (Arrays, Dictionaries)
Individual UI Elements (Label, Button, Image)	? DISCUSS ? ? ?

TABLE VIEW CONTROLLER

Like view controllers and UIViewController, unique table view controllers are made by subclassing UITableViewController.

```
class TableViewController: UITableViewController { }
```

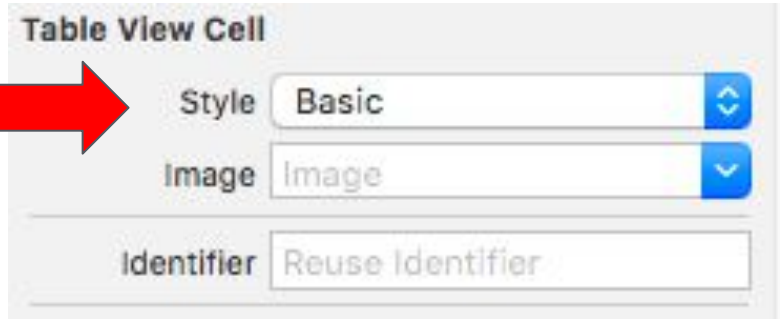
To define the behavior of your table view controller's table view, you will override table view methods.

```
override func tableViewNumberOfSections() { }           // abridged
override func tableViewNumberOfRowsPerSection() { }          // abridged
override func tableViewCellForRowAtIndexPath() { }      // abridged
```

TABLE VIEW CELL

Reuse Identifier

This string identifies the cell prototype. It can be anything.



Style

For practice, use “Basic”.

In real apps, you’ll almost always use “Custom”.

TABLE VIEW CONTROLLER

Practice:

1) Create a new table view controller.
2) Make it display 4 cells that each show the word “Prototype”.
3) Make it display 2 sections: the first with 3 cells that show the word “Prototype”, the second with 2 cells that show the word “Other”.

TABLE VIEWS

Exercise:

1) Create a new view controller.
2) Add a table view to that view controller.
3) Connect the table view to code with an outlet.

TABLE VIEWS

Exercise:

1) Create a new view controller.
2) Add a table view to that view controller.
3) Connect the table view to code with an outlet.

Question:

How will you make the table view display what you want it to?

TABLE VIEWS

To let a view controller handle logic for a table view, you must make that view controller conform to the UITableViewDelegate and UITableViewDataSource **protocols**.

You also have to set the table view's **delegate** and data source to be the view controller.

Vocab

Protocol - a set of specifications that describe what a class can do

Delegate - an object that's been designated to handle some part of another object's functionality

TABLE VIEWS

The viewDidLoad method is a typical place to set delegates and data sources. This is because viewDidLoad gets called at the beginning of a view controller's life cycle.

```
class ViewController: UIViewController, UITableViewDelegate, UITableViewDataSource {
     @IBOutlet weak var tableView: UITableView!
     Override func viewDidLoad() {
          super.viewDidLoad()
          tableView.delegate = self
          tableView.dataSource = self
     }
}
```

TABLE VIEWS

Practice:

1) Create a new view controller.
2) Add a table view to it.
3) Make your view controller the delegate and data source for its table view.
4) Make the table view display 3 cells that show the word “Orange”.