<https://developer.apple.com/documentation/uikit/uitableviewdatasource/1614871-tableview?language=objc>

# Deleting from the Beast List

Now that we can add and view items in the list let's add the functionality to delete items from the list when we tap on them. This will complete our Beast List application.

### Implement the UITableViewDelegate protocol

The UITableViewDelegate protocol allows the view controller to implement methods that can deal with User Interactions with the Table View cells.

In our case we want the task to disappear when we tap on it. **To listen to this tap event so that we can execute a chunk of code, we are going to have to implement the UITableViewDelegate protocol**. First we will **declare that our View Controller is going to conform to the UITableViewDelegate protocol:**

extension ViewController: UITableViewDataSource, UITableViewDelegate { // add UITableViewDelegate

}

Then we have to tell the tableView that its delegate will be the View Controller. Go ahead and do this in viewDidLoad

override func viewDidLoad() {

super.viewDidLoad()

tableView.dataSource = self

tableView.delegate = self // add this line

}

**Now let's implement didSelectRowAtIndexPath method (part of the UITableViewDelegate protocol). What we write here will be implemented as soon as one of the table view cells are tapped.** We will also be passed an indexPath argument to this message which we can use to figure out which cell was clicked. **Rows of the cell are similar to indexes in an array (they both start out with 0).**

func tableView(\_ tableView: UITableView, didSelectRowAt indexPath: IndexPath) {

print("Section: \(indexPath.section) and Row: \(indexPath.row)")

tasks.remove(at: indexPath.row)

tableView.reloadData()

}

copy

After we have removed the particular Model object that was clicked from the array, **we have to call reloadData on our tableView to update our cells with the recent changes in our array.**

### More UITableViewDelegate methods

There are many UITableViewDelegate methods that you may use for interactive functionality with the Table View. First remember that if you are trying to implement a feature that deals with interactivity with the Table View you should see if there is a delegate method by looking at the docs [here](https://developer.apple.com/library/ios/documentation/UIKit/Reference/UITableViewDelegate_Protocol/).