

```

//
// AppDelegate.swift
// Homeowner2
//
// Created by Chelsea Irizarry on 10/20/16.
// Copyright © 2016 Chelsea Irizarry. All rights reserved.
//

import UIKit

@UIApplicationMain
class AppDelegate: UIResponder, UIApplicationDelegate {

    var window: UIWindow?

    func application(_ application: UIApplication,
didFinishLaunchingWithOptions launchOptions:
[UIApplicationLaunchOptionsKey: Any]?) -> Bool {
        // Override point for customization after application launch.

        //Create ItemStore
        let itemStore = ItemStore()

        //Access the ItemsViewController and set its item store
        let navController = window!.rootViewController as!
        UINavigationController
        let itemsController = navController.topViewController as!
        ItemsViewController

        itemsController.itemStore = itemStore

        return true
    }

    func applicationWillResignActive(_ application: UIApplication) {
        // Sent when the application is about to move from active to
inactive state. This can occur for certain types of temporary
interruptions (such as an incoming phone call or SMS message) or when
the user quits the application and it begins the transition to the
background state.
        // Use this method to pause ongoing tasks, disable timers, and
invalidate graphics rendering callbacks. Games should use this method
to pause the game.
    }

    func applicationDidEnterBackground(_ application: UIApplication) {
        // Use this method to release shared resources, save user
data, invalidate timers, and store enough application state
information to restore your application to its current state in case
it is terminated later.

```

```

        // If your application supports background execution, this
method is called instead of applicationWillTerminate: when the user
quits.
    }

    func applicationWillEnterForeground(_ application: UIApplication)
{
    // Called as part of the transition from the background to the
active state; here you can undo many of the changes made on entering
the background.
    }

    func applicationDidBecomeActive(_ application: UIApplication) {
    // Restart any tasks that were paused (or not yet started)
while the application was inactive. If the application was previously
in the background, optionally refresh the user interface.
    }

    func applicationWillTerminate(_ application: UIApplication) {
    // Called when the application is about to terminate. Save
data if appropriate. See also applicationDidEnterBackground:.
    }
}

```

```

// DetailViewController.swift
// Homeowner2
//
// Created by Chelsea Irizarry on 11/11/16.
// Copyright © 2016 Chelsea Irizarry. All rights reserved.
//

```

```

import UIKit

class DetailViewController: UIViewController, UITextFieldDelegate {

    @IBOutlet var nameField: UITextField!
    @IBOutlet var serialNumberField: UITextField!
    @IBOutlet var valueField: UITextField!
    @IBOutlet var dateLabel: UILabel!

    var item: Item!{
        didSet{
            navigationItem.title=item.name
        }
    }

    @IBAction func backgroundTapped(_ sender: UITapGestureRecognizer)

```

```

{
    view.endEditing(true)
}

func textFieldShouldReturn(_ textField: UITextField) -> Bool {
    textField.resignFirstResponder()
    return true
}

override func viewWillDisappear(_ animated: Bool) {
    super.viewWillDisappear(animated)

    //Clear first responder
    view.endEditing(true)

    //"Save" changes to item
    item.name=nameField.text ?? ""
    item.serialNumber = serialNumberField.text

    if let valueText = valueField.text, let value =
numberFormatter.number(from: valueText) {
        item.valueInDollars = value.intValue
    }
    else {
        item.valueInDollars = 0
    }
}

override func viewWillAppear(_ animated: Bool) {
    super.viewWillAppear(animated)

    nameField.text = item.name
    serialNumberField.text = item.serialNumber
    valueField.text = numberFormatter.string(from:
item.valueInDollars as NSNumber)
    dateLabel.text = dateFormatter.string(from: item.dateCreated
as Date)
}

let numberFormatter: NumberFormatter = {
    let formatter = NumberFormatter()
    formatter.numberStyle = .decimal
    formatter.minimumFractionDigits = 2
    formatter.maximumFractionDigits = 2
    return formatter
} ()

let dateFormatter: DateFormatter = {
    let formatter = DateFormatter()
    formatter.dateStyle = .medium
} ()

```

```

        formatter.timeStyle = .none
        return formatter
    } ()
}

//
// ItemsViewController.swift
// Homeowner2
//
// Created by Chelsea Irizarry on 10/20/16.
// Copyright © 2016 Chelsea Irizarry. All rights reserved.
//

import UIKit

//Define a UITableViewController subclass named ItemsViewController
class ItemsViewController: UITableViewController {

    var itemStore: ItemStore!

    required init?(coder aDecoder: NSCoder) {

        super.init(coder: aDecoder)

        navigationItem.leftBarButtonItem = editButtonItem
    }

    override func viewWillAppear(_ animated: Bool) {
        super.viewWillAppear(animated)

        tableView.reloadData()
    }

    override func prepare(for segue: UIStoryboardSegue, sender: Any?)
    {
        //If triggered segue is the "Show Item" segue
        if segue.identifier == "ShowItem" {
            //Figure out which row was just tapped
            if let row = tableView.indexPathForSelectedRow?.row {
                //Get the item associated with this row and pass it
                let item = itemStore.allItems[row]
                let detailViewController = segue.destination as!
                DetailViewController
                detailViewController.item = item
            }
        }
    }
}

```

```

//Adding Rows
@IBAction func addNewItem(sender: AnyObject){

    //Create a new item and add it to the store
    let newItem = itemStore.createItem()

    //Figure out where that item is in the array
    if let index = itemStore.allItems.index(of: newItem) {
        let indexPath = NSIndexPath(row: index, section: 0)

        //Insert this new row into the table
        tableView.insertRows(at: [indexPath as IndexPath],
with: .automatic)
    }
}

//Deleting Rows
override func tableView(_ tableView: UITableView, commit
editingStyle: UITableViewCellEditingStyle, forRowAt indexPath:
IndexPath) {

    //If the table view is asking to commit a delete command...
    if editingStyle == .delete {
        let item = itemStore.allItems[indexPath.row]

        let title = "Delete \(item.name)?"
        let message = "Are you sure you want to delete this item?"

        let ac = UIAlertController(title: title, message: message,
preferredStyle: .actionSheet)

        let cancelAction = UIAlertAction(title: "Cancel",
style: .cancel, handler: nil)
        ac.addAction(cancelAction)

        let deleteAction = UIAlertAction(title: "Delete",
style: .destructive, handler: { (action: UIAlertAction) -> Void in
            //Remove the item from the store
            self.itemStore.removeItem(item: item)

            //Also remove that row from the table view with an
animation
            self.tableView.deleteRows(at: [indexPath],
with: .automatic)
        })
        ac.addAction(deleteAction)

        //Present the alert controller

```

```

        present(ac, animated: true, completion: nil)
    }
}

//Moving Rows
override func tableView(_ tableView: UITableView, moveRowAt
sourceIndexPath: IndexPath, to destinationIndexPath: IndexPath) {
    //Update the model
    itemStore.moveItemAtIndex(fromIndex: sourceIndexPath.row,
toIndex: destinationIndexPath.row)
}

    override func tableView(_ tableView: UITableView,
numberOfRowsInSection section: Int) -> Int {
        return itemStore.allItems.count
    }

    override func tableView(_ tableView: UITableView, cellForRowAt
indexPath: IndexPath) -> UITableViewCell {
        //Get a new or recycled cell
        let cell = tableView.dequeueReusableCell(withIdentifier:
"ItemCell", for: indexPath) as! ItemCell

        //Update the labels for the new preferred text size
        cell.updateLabels()

        //Set the text on the cell with the description of the item
        //that is at the nth index of items, where n = row this cell
        //will appear in on the tableview
        let item = itemStore.allItems[indexPath.row]

        //Configure the cell with the Item
        cell.nameLabel.text = item.name
        cell.serialNumberLabel.text = item.serialNumber
        cell.valueLabel.text = "$\(item.valueInDollars)"

        return cell
    }

    override func viewDidLoad() {
        super.viewDidLoad()

        tableView.rowHeight = UITableViewAutomaticDimension
        tableView.estimatedRowHeight = 65
    }
}

```