hxw//

// TaskTableViewController.swift

// segueApp

//

// Created by MacStudent on 2019-11-04.

// Copyright © 2019 MacStudent. All rights reserved.

//

import UIKit

class TaskTableViewController: UITableViewController {

var tasks : [String]?

var currentIndex = -1

override func viewDidLoad() {

super.viewDidLoad()

// Uncomment the following line to preserve selection between presentations

// self.clearsSelectionOnViewWillAppear = false

// Uncomment the following line to display an Edit button in the navigation bar for this view controller.

// self.navigationItem.rightBarButtonItem = self.editButtonItem

tasks = ["important : Buy milk", "clean kitchen", "wash dishes", "pay bills"]

}

// MARK: - Table view data source

override func numberOfSections(in tableView: UITableView) -> Int {

// #warning Incomplete implementation, return the number of sections

return 1

}

override func tableView(\_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int {

// #warning Incomplete implementation, return the number of rows

return tasks?.count ?? 0

}

override func tableView(\_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {

if var task = tasks?[indexPath.row]

{

let range = task.range(of: "important : ")

// print(range)

// let contain = tasks?.contains("important:"

// let cellidentifier = (contain == true) ?

// "importanttask" : "normaltask"

let cellidentifier = (range == nil) ?

"normaltext" : "importanttext"

if let range = task.range(of: "important : ") {

task.removeSubrange(range)

}

if let cell = tableView.dequeueReusableCell(withIdentifier: cellidentifier)

{

if let label = cell.viewWithTag(1) as?

UILabel{

label.text = task

}

print(tasks!)

return cell

}

}

return UITableViewCell()

}

/\*

// Override to support conditional editing of the table view.

override func tableView(\_ tableView: UITableView, canEditRowAt indexPath: IndexPath) -> Bool {

// Return false if you do not want the specified item to be editable.

return true

}

\*/

/\*

// Override to support editing the table view.

override func tableView(\_ tableView: UITableView, commit editingStyle: UITableViewCell.EditingStyle, forRowAt indexPath: IndexPath) {

if editingStyle == .delete {

// Delete the row from the data source

tableView.deleteRows(at: [indexPath], with: .fade)

} else if editingStyle == .insert {

// Create a new instance of the appropriate class, insert it into the array, and add a new row to the table view

}

}

\*/

/\*

// Override to support rearranging the table view.

override func tableView(\_ tableView: UITableView, moveRowAt fromIndexPath: IndexPath, to: IndexPath) {

}

\*/

/\*

// Override to support conditional rearranging of the table view.

override func tableView(\_ tableView: UITableView, canMoveRowAt indexPath: IndexPath) -> Bool {

// Return false if you do not want the item to be re-orderable.

return true

}

\*/

// MARK: - Navigation

// In a storyboard-based application, you will often want to do a little preparation before navigation

override func prepare(for segue: UIStoryboardSegue, sender: Any?) {

// Get the new view controller using segue.destination.

// Pass the selected object to the new view controller.

if let detailView = segue.destination as? TaskDetailViewController {

detailView.taskTable = self

if let tableViewCell = sender as? UITableViewCell {

if let index = tableView.indexPath(for: tableViewCell)?.row {

detailView.textString = tasks![index]

currentIndex = index

}

}

}

}

func updateText(text: String) {

guard tasks != nil && currentIndex != -1 else {

return

}

tasks![currentIndex] = text

// tableView.reloadData()

let indexPath = IndexPath(item: currentIndex, section: 0)

tableView.reloadRows(at: [indexPath], with: .none)

}

}