//

// ListOfSongsVC.swift

// MusicAssistant

//

// Created by Ciro on 6/30/17.

// Copyright © 2017 Ciro. All rights reserved.

//

import UIKit

var artist = [String]()

var songTitle = [[String]]()

var keys = [Int: String]()

var myIndexForRow = 0

var myIndexForSection = 0

var count = 0

var getKey = 0

var newSongTitle = [[String]]()

var sortedArtist = [String]()

class ListOfSongsVC: UITableViewController {

//function to set up sections titles

override func tableView(\_ tableView: UITableView, titleForHeaderInSection section: Int) -> String? {

return sortedArtist[section]

}

//function to count number of sections

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

return sortedArtist.count

}

//function to count number of rows in the table view

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

return newSongTitle[section].count

}

//function to setup tableview initial value

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

let cell = tableView.dequeueReusableCell(withIdentifier: "cell", for: indexPath)

cell.textLabel?.text = newSongTitle[indexPath.section][indexPath.row]

cell.detailTextLabel?.text = sortedArtist[indexPath.section]

cell.textLabel?.font = UIFont(name: "Avenir Next", size: 24)

tableView.tableFooterView = UIView()

return cell

}

override func viewDidLoad() {

super.viewDidLoad()

saveFileOnSystem()

orderAlphabetically()

tableView.reloadData()

}

//function to setup array on file

func saveFileOnSystem() {

//check to see if it is the first lauch

let lauchedBefore = UserDefaults.standard.bool(forKey: "launchedBefore")

//if launched before use SongList already created on device, if not create new SongList

if lauchedBefore {

let defaults = UserDefaults.standard

songTitle = defaults.array(forKey: "SongList") as! [Array<String>]

let defaultArtist = UserDefaults.standard

artist = defaultArtist.stringArray(forKey: "Artist") ?? [String]()

keys = retrieveDictionary(withKey: "Keys")!

} else {

let defaults = UserDefaults.standard

defaults.set(songTitle, forKey: "SongList")

let defaultsArtist = UserDefaults.standard

defaultsArtist.set(artist, forKey: "Artist")

save(dictionary: keys, forKey: "Keys")

UserDefaults.standard.set(true, forKey: "launchedBefore")

}

}

//save key dictionary to system

func save(dictionary: [Int: String], forKey key: String) {

let archiver = NSKeyedArchiver.archivedData(withRootObject: keys)

UserDefaults.standard.set(archiver, forKey: key)

}

//retrieving keys dictionary from system

func retrieveDictionary(withKey key: String) -> [Int: String]? {

// Check if data exists

guard let data = UserDefaults.standard.object(forKey: key) else {

return nil

}

// Check if retrieved data has correct type

guard let retrievedData = data as? Data else {

return nil

}

// Unarchive data

let unarchivedObject = NSKeyedUnarchiver.unarchiveObject(with: retrievedData)

return unarchivedObject as? [Int: String]

}

//function to reload data when page is open

override func viewDidAppear(\_ animated: Bool) {

tableView.reloadData()

}

//function to go to songViewVC when row is pressed

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

myIndexForSection = indexPath.section

myIndexForRow = indexPath.row

performSegue(withIdentifier: "songViewSegue", sender: self)

}

//function to add delete button when row is swiped

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

if editingStyle == UITableViewCellEditingStyle.delete{

// songTitle[indexPath.section].remove(at: indexPath.row)

// if songTitle[indexPath.section].count < 1 {

// artist.remove(at: indexPath.section)

// songTitle.remove(at: indexPath.section)

// }

// orderAlphabetically()

// tableView.reloadData()

//

// //update entry from defaults

// let defaults = UserDefaults.standard

// defaults.set(songTitle, forKey: "SongList")

// let defaultsArtist = UserDefaults.standard

// defaultsArtist.set(artist, forKey: "Artist")

}

}

//function to order alphabetically

func orderAlphabetically(){

var newArtist = artist

sortedArtist = artist

sortedArtist.sort{$0 < $1}

newSongTitle = [[String]]()

for currentArtist in sortedArtist {

for i in 0..<newArtist.count {

if currentArtist == newArtist[i] {

newSongTitle.append(songTitle[i])

break

}

}

}

for i in 0..<songTitle.count {

if songTitle.count > 1 {

newSongTitle[i].sort {$0 < $1}

}

}

}

}