**1、自定义cell**

(1)CustomTableViewCell类

import UIKit

class CumtomTableViewCell: UITableViewCell {

//属性用来存储用于显示到UI上的值

//@IBOutlet用于引用storyboard上的一个UI组件

//@IBAction用于响应一个UI组件的交互事件

@IBOutlet weak var nameLabel: UILabel!

@IBOutlet weak var provinceLabel: UILabel!

@IBOutlet weak var partLabel: UILabel!

@IBOutlet weak var thumbImageView: UIImageView!

}

（2）AreaTableViewController类

import UIKit

class AreaTableViewController: UITableViewController {

//名称数组

var areas = ["徐州","济南","石家庄","武汉","长沙","三亚","杭州","合肥","桂林","广州"]

//省份数组

var provinces = ["江苏省","山东省","河北省","湖北省","湖南省","海南省","浙江省","安徽省","广西省","广东省"]

//地区数组

var parts = ["华东","华东","华北","华中","华中","华南","华东","华东","广西","华南"]

//图标名称数组

var areaImages = ["xuzhou","jinan","shijiazhuang","wuhan","changsha","sanya","hangzhou","hefei","guilin","guangzhou"]

// MARK: - Table view data source

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

return areas.count

}

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

//as做类型转换，从一个类型转换到另一个类型

//as! 强制转换（失败app会崩溃，使用as!一般在转换的时候比较确定转换是成功的）

//as？安全转换（失败不会崩溃）

// 将 UITableViewCell 转换成 CumtomTableViewCell

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

//cell中名称标签赋值

cell.nameLabel.text = areas[indexPath.row]

//通过读取数组元素得到对应图片名称，设置imageView上的图片

cell.thumbImageView.image = UIImage(named: areaImages[indexPath.row])

//cell中省份标签赋值

cell.provinceLabel.text = provinces[indexPath.row]

//cell中地区标签赋值

cell.partLabel.text = parts[indexPath.row]

//将图片的圆角半径设置成宽度的一半，调节layer.cornerRadius属性

cell.thumbImageView.layer.cornerRadius = cell.thumbImageView.frame.size.height/2

//让layer设置生效，将裁边选项值设为true

cell.thumbImageView.clipsToBounds = true

return cell

}

}

**2、弹出选中行**

在上一个工程AreaTableViewController类里增加下面这个方法

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

//定义一个弹出框

let menu = UIAlertController(title: "您好", message: "您选择了第\(indexPath.row + 1)行", preferredStyle: .actionSheet)

//定义一个按钮用于用户交互

let option2 = UIAlertAction(title: "取消", style: .cancel, handler: nil)

let option3 = UIAlertAction(title: "我去过", style: .default) { (\_) in

//取得点击的单元格

let cell = tableView.cellForRow(at: indexPath)

//将单元格打钩

cell?.accessoryType = .checkmark

}

//将按钮加入到弹出框中

menu.addAction(option2)

menu.addAction(option3)

//显示菜单

self.present(menu, animated: true, completion: nil)

}

**3、修正重复打钩的bug**

（1）CustomTableViewCell类 同上

（2）AreaTableViewController类

import UIKit

class AreaTableViewController: UITableViewController {

var areas = ["徐州","济南","石家庄","武汉","长沙","三亚","杭州","合肥","桂林","广州"]

var provinces = ["江苏省","山东省","河北省","湖北省","湖南省","海南省","浙江省","安徽省","广西省","广东省"]

var parts = ["华东","华东","华北","华中","华中","华南","华东","华东","广西","华南"]

var areaImages = ["xuzhou","jinan","shijiazhuang","wuhan","changsha","sanya","hangzhou","hefei","guilin","guangzhou"]

//定义一个visited数组，记录cell是否被选中

var visited = [Bool](repeatElement(false, count: 10))

override func viewDidLoad() {

super.viewDidLoad()

}

// MARK: - Table view dalegate

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

//定义一个弹出框

let menu = UIAlertController(title: "您好", message: "您选择了第\(indexPath.row + 1)行", preferredStyle: .actionSheet)

//定义一个按钮用于用户交互

let option2 = UIAlertAction(title: "取消", style: .cancel, handler: nil)

let option3 = UIAlertAction(title: "我去过", style: .default) { (\_) in

//取得点击的单元格

let cell = tableView.cellForRow(at: indexPath)

//将单元格打钩

cell?.accessoryType = .checkmark

//将打钩状态保存到visited数组中

self.visited[indexPath.row] = true

}

//将按钮加入到弹出框中

menu.addAction(option2)

menu.addAction(option3)

//显示菜单

self.present(menu, animated: true, completion: nil)

}

//下面一行是为了分隔提示

// MARK: - Table view data source

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

return areas.count

}

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

//as做类型转换，从一个类型转换到另一个类型

//as! 强制转换（失败app会崩溃，使用as!一般在转换的时候比较确定转换是成功的）

//as？安全转换（失败不会崩溃）

// 将 UITableViewCell 转换成 CumtomTableViewCell

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

//cell中名称标签赋值

cell.nameLabel.text = areas[indexPath.row]

//通过读取数组元素得到对应图片名称，设置imageView上的图片

cell.thumbImageView.image = UIImage(named: areaImages[indexPath.row])

//cell中省份标签赋值

cell.provinceLabel.text = provinces[indexPath.row]

//cell中地区标签赋值

cell.partLabel.text = parts[indexPath.row]

//将图片的圆角半径设置成宽度的一半，调节layer.cornerRadius属性

cell.thumbImageView.layer.cornerRadius = cell.thumbImageView.frame.size.height/2

//让layer设置生效，将裁边选项值设为true

cell.thumbImageView.clipsToBounds = true

//重用单元格时判断是否被打钩

if visited[indexPath.row]{

cell.accessoryType = .checkmark

}else{

cell.accessoryType = .none

}

return cell

}

}

**4、左滑删除行**

（1）CustomTableViewCell类 同上

（2）AreaTableViewController类

import UIKit

class AreaTableViewController: UITableViewController {

var areas = ["徐州","济南","石家庄","武汉","长沙","三亚","杭州","合肥","桂林","广州"]

var provinces = ["江苏省","山东省","河北省","湖北省","湖南省","海南省","浙江省","安徽省","广西省","广东省"]

var parts = ["华东","华东","华北","华中","华中","华南","华东","华东","广西","华南"]

var areaImages = ["xuzhou","jinan","shijiazhuang","wuhan","changsha","sanya","hangzhou","hefei","guilin","guangzhou"]

var visited = [Bool](repeatElement(false, count: 10))

override func viewDidLoad() {

super.viewDidLoad()

}

// MARK: - Table view dalegate

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

//定义一个弹出框

let menu = UIAlertController(title: "您好", message: "您选择了第\(indexPath.row + 1)行", preferredStyle: .actionSheet)

//定义一个按钮用于用户交互

let option2 = UIAlertAction(title: "取消", style: .cancel, handler: nil)

let option3 = UIAlertAction(title: "我去过", style: .default) { (\_) in

//取得点击的单元格

let cell = tableView.cellForRow(at: indexPath)

//将单元格打钩

cell?.accessoryType = .checkmark

//将打钩状态保存到visited数组中

self.visited[indexPath.row] = true

}

//将按钮加入到弹出框中

menu.addAction(option2)

menu.addAction(option3)

//显示菜单

self.present(menu, animated: true, completion: nil)

}

//下面一行是为了分隔提示

// MARK: - Table view data source

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

return areas.count

}

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

//as做类型转换，从一个类型转换到另一个类型

//as! 强制转换（失败app会崩溃，使用as!一般在转换的时候比较确定转换是成功的）

//as？安全转换（失败不会崩溃）

// 将 UITableViewCell 转换成 CumtomTableViewCell

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

//cell中名称标签赋值

cell.nameLabel.text = areas[indexPath.row]

//通过读取数组元素得到对应图片名称，设置imageView上的图片

cell.thumbImageView.image = UIImage(named: areaImages[indexPath.row])

//cell中省份标签赋值

cell.provinceLabel.text = provinces[indexPath.row]

//cell中地区标签赋值

cell.partLabel.text = parts[indexPath.row]

//将图片的圆角半径设置成宽度的一半，调节layer.cornerRadius属性

cell.thumbImageView.layer.cornerRadius = cell.thumbImageView.frame.size.height/2

//让layer设置生效，将裁边选项值设为true

cell.thumbImageView.clipsToBounds = true

//重用单元格时判断是否被打钩

if visited[indexPath.row]{

cell.accessoryType = .checkmark

}else{

cell.accessoryType = .none

}

return cell

}

//插入或者删除指定行

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

//删除行

if editingStyle == .delete{

//从数据源中删除相应的行

//移除areaImages数组中图标名称

self.areaImages.remove(at: indexPath.row)

//移除areas数组中名称

self.areas.remove(at: indexPath.row)

//移除provinces数组中省份名称

self.provinces.remove(at: indexPath.row)

//移除parts数组中地区名称

self.parts.remove(at: indexPath.row)

//删除访问记录

self.visited.remove(at: indexPath.row)

//删除某一行刷新

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

//整体刷新

//tableView.reloadData()

}else if editingStyle == .insert{

}

}

}