实验编号： 13 **四川师大《IOS》实验报告 2018** 年 **12** 月 **12** 日

### **计算机科学学院** 2016 级 4 班 实验名称： 多线程和网络程序设计 \_

姓名：\_\_\_蒋宇童\_\_\_\_\_\_ 学号：\_\_2016110419\_\_\_\_\_\_\_ 指导老师：\_\_李贵洋\_\_ 实验成绩:\_\_\_\_\_

**实验 十三 \_**多线程和网络程序设计**\_\_\_\_\_**

1. 实验目的及要求
2. 理解并掌握iOS多线程编程的相关技术；
3. 掌握GCD关键技术，包括block、dispatch等；
4. 掌握WebView的使用；
5. 掌握URLSession的使用，
6. 掌握第三网络库Alamofire的使用方法；
7. 掌握Json的解析。
8. 实验内容
9. 采用多线程技术，实现一个大数加程序。
   1. 正确理解DispatchQueue的使用
   2. 从1 到 9999999
   3. 不能阻塞UI主线程
10. Web浏览器;
    1. 使用WebView控件写成一个简易的浏览器，有浏览器的基本功能；
11. 使用网络库进行天气Json数据的解析
    1. APP有两个界面，第一个界面：tableview显示一个城市列表
    2. 第二个界面，显示选择城市的天气数据
    3. 使用第三方网络Alamofire进行网络的连接，获取网络天气数据；
    4. 对获取到的网络数据进行Json的解析；
    5. 天气数据库位置：http://t.weather.sojson.com/api/weather/city/101270101

可选其他 Web API 进行解析:

1. 免费 JSON API: <http://www.sojson.com/api/>
2. 源代码管理 GitHub API: <https://developer.github.com/v3/>
3. 图形识别: <https://imagga.com/>

4. 摄影社区: <https://500px.com/>

1. 实验主要流程、基本操作或核心代码、算法片段（该部分如不够填写，请另加附页）
2. 采用多线程技术，实现一个大数加程序。
   1. 正确理解DispatchQueue的使用
   2. 从1 到 9999999
   3. 不能阻塞UI主线程

* 程序代码：

//

// ViewController.swift

// Thread

//

// Created by jiang on 2018/12/15.

// Copyright © 2018年 蒋宇童. All rights reserved.

//

import UIKit

class ViewController: UIViewController {

@IBOutlet weak var lbcount: UILabel!

@IBOutlet weak var lbSum: UILabel!

var count=0

override func viewDidLoad() {

super.viewDidLoad()

Timer.scheduledTimer(withTimeInterval: 1, repeats: true) { (timer) in

self.count+=1

self.lbcount.text="\(self.count)"

}

// Do any additional setup after loading the view, typically from a nib.

}

override func didReceiveMemoryWarning() {

super.didReceiveMemoryWarning()

// Dispose of any resources that can be recreated.

}

@IBAction func sum(\_ sender: Any) {

var sum=0;

DispatchQueue.global().async {

print("1")

for i in 1...99999999

{

sum += i

}

DispatchQueue.main.async {

self.lbSum.text = "\(sum)"

print("1")

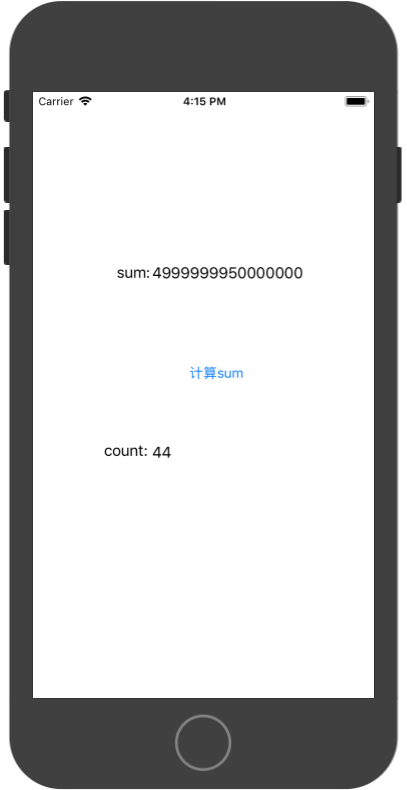
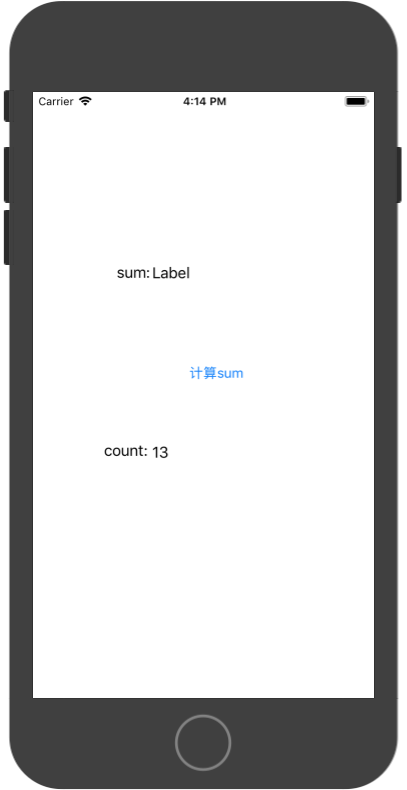
}

}

}

}

* 运行结果：



1. Web浏览器;
   1. 使用WebView控件写成一个简易的浏览器，有浏览器的基本功能；

* 程序代码：

//

// ViewController.swift

// WebView

//

// Created by jiang on 2018/12/15.

// Copyright © 2018年 蒋宇童. All rights reserved.

//

import UIKit

import WebKit

class ViewController: UIViewController {

@IBOutlet weak var webView: WKWebView!

@IBOutlet weak var urlTF: UITextField!

override func viewDidLoad() {

super.viewDidLoad()

if let url = URL(string: "http://www.163.com")

{

webView.load(URLRequest(url: url))

}

// Do any additional setup after loading the view, typically from a nib.

}

override func didReceiveMemoryWarning() {

super.didReceiveMemoryWarning()

// Dispose of any resources that can be recreated.

}

@IBAction func go(\_ sender: Any) {

if let url=URL(string: urlTF.text!)

{

webView.load(URLRequest(url: url))

}

}

@IBAction func back(\_ sender: Any) {

webView.goBack()

}

@IBAction func forword(\_ sender: Any) {

webView.goForward()

}

@IBAction func reload(\_ sender: Any) {

webView.reload()

}

}

* 运行结果：



1. 使用网络库进行天气Json数据的解析
   1. APP有两个界面，第一个界面：tableview显示一个城市列表
   2. 第二个界面，显示选择城市的天气数据
   3. 使用第三方网络Alamofire进行网络的连接，获取网络天气数据；
   4. 对获取到的网络数据进行Json的解析；
   5. 天气数据库位置：http://t.weather.sojson.com/api/weather/city/101270101

* 程序代码：

//

// CitysTableViewController.swift

// exp13\_3

//

// Created by student on 2018/12/17.

// Copyright © 2018年 Jxkicker. All rights reserved.

//

import UIKit

import Alamofire

class CitysTableViewController: UITableViewController {

var cityinfo = Cityinfo()

let citise = ["北京": 101010100, "上海": 101020100, "天津": 101030100, "重庆": 101040100, "哈尔滨": 101050101, "长春": 101060101, "沈阳": 101070101, "呼和浩特": 101080101, "石家庄": 101090101, "太原": 101100101, "西安": 101110101, "济南": 101120101, "乌鲁木齐": 101130101, "拉萨": 101140101, "西宁": 101150101, "兰州": 101160101, "银川": 101170101, "郑州": 101180101, "南京": 101190101, "武汉": 101200101, "杭州": 101210101, "合肥": 101220101, "福州": 101230101, "南昌": 101240101, "长沙": 101250101, "贵阳": 101260101, "成都": 101270101, "广州": 101280101, "昆明": 101290101, "南宁": 101300101, "海口": 101310101, "香港": 101320101, "澳门": 101330101, "台北县": 101340101]

let Citys = ["成都","北京","上海","重庆","天津","黑龙江"]

let url = URL(string:"http://t.weather.sojson.com/api/weather/city/101270101")!

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

}

// 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 citise.count

}

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

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

cell.textLabel?.text = Array(citise.keys)[indexPath.row]

cell.detailTextLabel?.text="\(Array(citise.values)[indexPath.row])"

return cell

}

/\*

// 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: UITableViewCellEditingStyle, 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?) {

if let nextPage = segue.destination as? CityWeatherViewController{

if let indexPath = tableView.indexPath(for: sender as! UITableViewCell){

let number = "\(Array(citise.values)[indexPath.row])"

let url\_ = "http://t.weather.sojson.com/api/weather/city/\(number ?? "101270101")"

let url = URL(string:url\_)!

AF.request(url).responseJSON {(response) in

// print(response.data!) //没有序列化的json 数据

// print(response.result.value!) // 序列化的json

let json\_ = response.result.value as! Dictionary<String,AnyObject>

let cityInfo = json\_["cityInfo"] as! Dictionary<String,AnyObject>

let todayinfo = json\_["data"] as! Dictionary<String,AnyObject>

// 2.得到湿度

self.cityinfo.shidu = todayinfo["shidu"] as! String

// 3.得到pm25

self.cityinfo.pm25 = "\(todayinfo["pm25"] as! Int)"

// 4.得到建议

self.cityinfo.ganmaoadvice = todayinfo["ganmao"] as! String

// 5.得到时间

self.cityinfo.time = json\_["time"] as! String

// 6.得到空气质量

self.cityinfo.quality = todayinfo["quality"] as! String

// 7.得到更新时间

self.cityinfo.updateTime = cityInfo["updateTime"] as! String

// 8.获取温度

self.cityinfo.wendu = todayinfo["wendu"] as! String

self.cityinfo.getcity=Array(self.citise.keys)[(self.tableView.indexPathForSelectedRow?.row)!]

}

nextPage.cityinfo=cityinfo

print("dsdjks\(cityinfo.wendu)")

print(cityinfo.getcity)

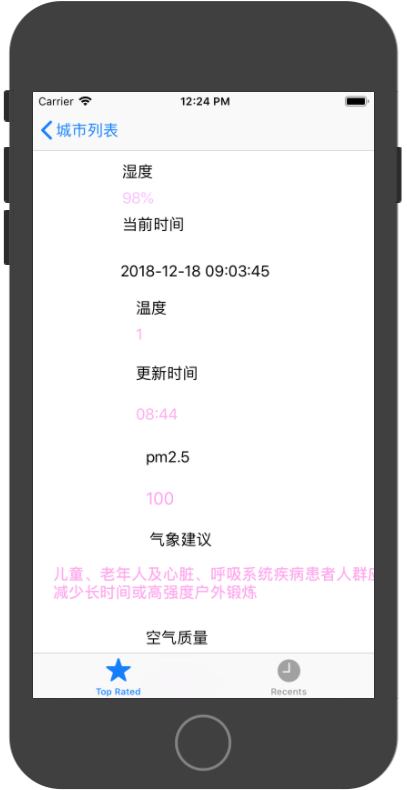
}

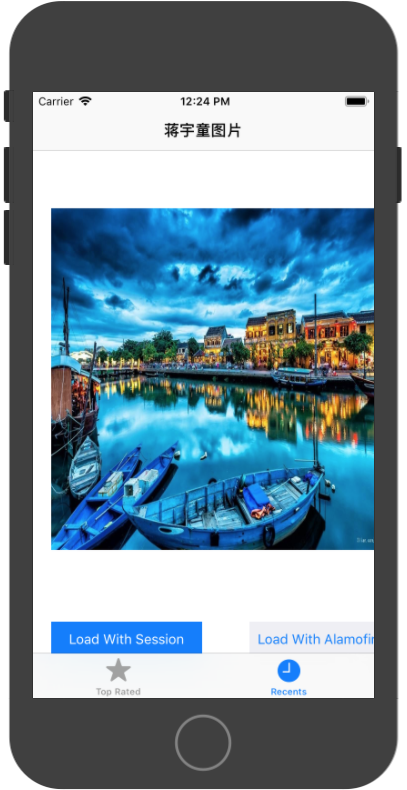
}

}

}

* 运行结果：





可选其他 Web API 进行解析:

1. 免费 JSON API: <http://www.sojson.com/api/>101270101
2. 源代码管理 GitHub API: <https://developer.github.com/v3/>
3. 图形识别: <https://imagga.com/>

4. 摄影社区: <https://500px.com/>

1. 实验结果的分析与评价（该部分如不够填写，请另加附页）

**Github地址：**[**https://github.com/jiangyutong/swiftWork/tree/master/代码**](https://github.com/jiangyutong/swiftWork/tree/master/代码)

通过这次实验我学会怎么去学习用Almofire去获取网络数据，其实这个实验我写了一天，因为Xcode9不能加载Almofire，所以有下载一个Xcode10。好嘛有读json数据的时候因为没有理解透json，所以就一直没有取出数据。但是后面了解了几句写出来了。

注：实验成绩等级分为（90－100分）优，（80－89分）良，(70-79分)中，（60－69分）及格，（59分）不及格。