实验编号：4 **四川师大《IOS高级开发技术》实验报告 2018** 年**9**月**19**日

**计算机科学学院**2016 级 4班 实验名称： 作业三 \_

姓名：\_谭靖薇\_ 学号：\_2016110437\_\_ 指导老师：\_李贵洋\_ 实验成绩:\_\_\_\_\_

**实验\_三\_ \_\_\_\_\_作业三\_\_\_\_\_**

1. 实验目的及要求
2. 实验内容

**作业1（Date、String、文件、URL）**

1. 显示当前准确的中文时间，包括北京、东京、纽约、伦敦，格式为（2016年9月28日星期三 上午10:25）
   1. 显示中文需要设置locale
2. 处理字符串
   1. 新建字符串：“Swift is a powerful and intuitive programming language for iOS, OS X, tvOS, and watchOS.”；
   2. 返回字符串从第6个字符到第20个字符的子串；
   3. 删除其中所有的OS字符；
3. 将1、2题的时间和字符串存入一个字典中，并存入沙盒中的Document某文件中；
4. 从网上下载一张照片并保存到本地沙盒的Document的某文件中；
5. 从网上查找访问一个JSon接口文件，并采用JSONSerialization和字典对其进行简单解析；
6. 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）

作业1

import Foundation

import Glibc

func getDate(date: Date, zone: Int = 0) -> String {

let formatter = DateFormatter() //实例化格式化类

formatter.dateFormat = "yyyy年MM月dd日EEEE aa KK:mm" //指定格式化的格式 formatter.locale = Locale.current

if zone >= 0 { //当传入的为正数时，在东半区

formatter.timeZone = TimeZone(abbreviation: "UTC+\(zone):00")

} else { //当传入的为负数时，在西半区

formatter.timeZone = TimeZone(abbreviation: "UTC\(zone):00")

}

let dateString = formatter.string(from: now) //将传入的日期格式化为字符串

return dateString

}

let now = Date() //获取当前时间日期

let beijing = getDate(date: now, zone: +8) //获取当前北京的时间

print("北京: \(beijing)")

let tokyo = getDate(date: now, zone: 9) //获取当前东京的时间

print("东京: \(tokyo)")

let newYork = getDate(date: now, zone: -5) //获取当前纽约的时间

print("纽约: \(newYork)")

let london = getDate(date: now) ////获取当前伦敦的时间

print("伦敦: \(london)")

//a) 新建字符串：“Swift is a powerful and intuitive programming language for iOS, OS X, tvOS, and watchOS.”；

var str2 = "Swift is a powerful and intuitive programming language for iOS, OS X, tvOS, and watchOS."

//b) 返回字符串从第6个字符到第20个字符的子串

var si = str2.index(str2.startIndex,offsetBy:5)

var ei = str2.index(str2.startIndex,offsetBy:19)

var result1 = str2.substring(with:si..<ei)

print(result1)

//c) 删除其中所有的OS字符

let result2 = str2.replacingOccurrences(of:"OS",with:"")

print(result2)

//3. 将1、2题的时间和字符串存入一个字典中，并存入沙盒中的Document某文件中

let dic = ["date": ["beijing": beijing, "tokyo": tokyo, "newYork": newYork, "london": london], "string": result2] as AnyObject //将字典转换为任意类型，方便后面写入文件

let defaultDoc = FileManager.default //获取默认工作路径

//获取工作路径下的Document文件夹

if var path = defaultDoc.urls(for: .documentDirectory, in: .userDomainMask).first?.path {

path.append("/test.txt") //在文件夹路径下增加一个test.txt

print(dic.write(toFile: path, atomically: true)) //新建上面指定的文件，并将数据写入

//方法一

let h3filePath:String = NSHomeDirectory() + "/Documents/dictionary.plist"

dictionary.write(toFile: h3filePath, atomically: true)

//方法二

// 1、获得沙盒的根路径

let home=NSHomeDirectory()//获取沙盒根路径

//获取文档目录路

let docPath = home+"/Documents/"

// 3、获取文本文件路径

let filePath = docPath + "data.plist"

dict3.write(toFile: filePath,atomically : true)

//4. 从网上下载一张照片并保存到本地沙盒的Document的某文件中；

Let image = try Data(contentsOf: URL(

string:"https://ss0.bdstatic.com/5aV1bjqh\_Q23odCf/static/superman/img/logo/bd\_logo1\_31bdc765.png")!) //通过指定的url获取图片，并转换为二进制数据 if var url = defaultDoc.urls(for: .documentDirectory, in: .userDomainMask).first {

url.appendPathComponent("image.png")

try image.write(to: url) //将转换后的二进制数据存储为png图片

}

//5. 从网上查找访问一个JSon接口文件，并采用JSONSerialization和字典对其进行简单解析；

let url = URL(string: "http://www.weather.com.cn/data/sk/101110101.html")! //api的路径

let data = try Data(contentsOf: url)

//将json转换为二进制数据

let json = try JSONSerialization.jsonObject(with: data, options: .allowFragments) //序列化json

//解析json数据

if let dic = json as? [String: Any] {

if let weather = dic["weatherinfo"] as? [String: Any] {

let city = weather["city"]!

let temp = weather["temp"]!

let wd = weather["WD"]!

let ws = weather["WS"]!

print("城市: \(city), 温度: \(temp), 风向: \(wd), 风力: \(ws)")

}

}

1. 实验结果的分析与评价（该部分如不够填写，请另加附页）

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。