实验编号：3 **四川师大《IOS高级开发技术》实验报告 2018** 年**9**月**12**日

**计算机科学学院**2016 级 4班 实验名称： 作业四 \_

姓名：\_\_谭靖薇 学号：2016110437\_ 指导老师：\_李贵洋\_ 实验成绩:\_\_\_\_\_

**实验\_四\_ \_\_\_\_\_作业四\_\_\_\_\_**

1. 实验目的及要求
2. 实验内容

文件缓存处理：

* 1. 判断沙盒的Document目录下是否存在某文件夹，如果没有则新建一个该文件夹；
  2. 判断是否该文件夹下存在一个图片文件，如果存在该文件，读取该文件到一个图片对象中并进行显示，如果不存在则从网上下载一张图片并保存到该图片文件中。

自定制视图：

* 1. 从UIView中派生一个自定制的View；
  2. 绘制一个椭圆（或则自己喜欢的任何图形）；
  3. 新建视图对象并进行显示；

代码版hello world（在视图控制器中加入代码）：

* 1. 代码中生成label（outlet）和button；
  2. 将label和button加入根view中；
  3. button添加像self（target）发射action（selector）的操作；
  4. 实现clicked响应代码（selector）

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）

作业一

let fileManager = FileManager.default

if var docPath = fileManager.urls(for: .documentDirectory, in: .userDomainMask).first{

docPath.appendPathComponent("tjw")

if !fileManager.fileExists(atPath: docPath.path){

try fileManager.createDirectory(at: docPath, withIntermediateDirectories: true, attributes: nil)

}

docPath.appendPathComponent("mypic.jpg")

if fileManager.fileExists(atPath: docPath.path){

let imageview = UIImageView(frame: CGRect(x: 0, y: 0, width: 400, height: 200))

imageview.image = UIImage(contentsOfFile:docPath.path)

}else{

var url = fileManager.urls(for: .documentDirectory, in: .userDomainMask).first!

let imageUrl = URL(string: "http://pic5.photophoto.cn/20071228/0034034901778224\_b.jpg")!

let imageData = try? Data(contentsOf: imageUrl)

url.appendPathComponent("mypic.jpg")

try? imageData?.write(to: url)

let image = UIImage(data: imageData!)

let imageview = UIImageView(frame: CGRect(x: 0, y: 0, width: 400, height: 200))

imageview.image = image

}

}

作业二

class MyViewController : UIViewController {

var label:UILabel!

@IBAction func clicked(){

print("i am clicked")

label?.text = "i am clicked"

}

override func loadView() {

let view = UIView(frame: CGRect(x: 0, y: 0, width: 300, height: 300))

view.backgroundColor = .red

label = UILabel(frame: CGRect(x: 70, y: 10, width: 200, height: 20))

label.text = "Hello World!"

label.textColor = .black

view.addSubview(label)

作业三

import UIKit

import PlaygroundSupport

class drawsth:UIView{

override func draw(\_ rect: CGRect) {

//获取绘图上下文

guard let context = UIGraphicsGetCurrentContext() else {

return

}

//创建一个矩形，它的所有边都内缩3点

let drawrect = self.bounds.insetBy(dx: 3, dy: 3)

//创建并设置路径

let path = CGMutablePath()

//绘制椭圆

path.addEllipse(in: drawrect)

//添加路径到图形上下文

context.addPath(path)

//设置笔触颜色

context.setStrokeColor(UIColor.orange.cgColor)

//设置笔触宽度

context.setLineWidth(6)

//设置填充颜色

context.setFillColor(UIColor.blue.cgColor)

//绘制路径并填充

context.drawPath(using: .fillStroke)

}

}

class MyViewController : UIViewController {

var label:UILabel!

@IBAction func clicked(){

print("i am clicked")

label?.text = "i am clicked"

}

override func loadView() {

let view = UIView(frame: CGRect(x: 0, y: 0, width: 300, height: 300))

view.backgroundColor = .red

label = UILabel(frame: CGRect(x: 70, y: 10, width: 200, height: 20))

label.text = "Hello World!"

label.textColor = .black

view.addSubview(label)

let button = UIButton(frame: CGRect(x: 70, y: 100, width: 200, height: 20))

button.setTitle("click me!", for: .normal)

button.addTarget(self, action: #selector(MyViewController.clicked), for: UIControlEvents.touchUpInside)

view.addSubview(button)

let arect = drawsth(frame: CGRect(x: 70, y: 120, width: 200, height: 100))

arect.backgroundColor = UIColor.clear

view.addSubview(arect)

}

}

let mvc = MyViewController()

PlaygroundPage.current.liveView = mvc

1. 实验结果的分析与评价（该部分如不够填写，请另加附页）

Git的地址：https://github.com/xiongmaobeibei/ios\_homework/

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。