实验编号： 9 **四川师大《IOS》实验报告 2018** 年 **11** 月 **7** 日

### **计算机科学学院** 2016 级 4 班 实验名称： Gesture、UIAlertController、 ScrollView

姓名：\_樊琳\_\_ 学号：2016110408\_ 指导老师：\_\_李贵洋\_\_ 实验成绩:\_\_\_\_\_

**实验 九 \_\_\_\_\_** Gesture、UIAlertController、 ScrollView **\_\_\_\_\_\_\_**

1. 实验目的及要求
2. 理解并掌握iOS多点手势识别的相关技术；
3. 实验内容
4. Gesture
   1. 分别采用代码随机位置大小生成和直接拖拽的方式产生多个视图；
   2. 采用简单的动画进行移动；
   3. 给视图加上阴影(layer)；
   4. 可全部清空子视图；
   5. 视图支持手势（pan移动、tap删除、pinch缩放、rotation旋转）；

提示：Pinch的scale属性可用于调整frame

rotation需要用transform属性实现

1. 实现UIAlertController交互
   1. 显示ActionSheet并进行交互；
   2. 显示Login Alert并进行交互；
2. 一个界面使用两个scrollView
   1. 在一个scrollView中可进行多张图片横屏滚动浏览(相册)，需要有pagecontrol进行提示；
   2. 在另一个scrollView中可放大缩小；

提示：需用delegate

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. Gesture
   1. 分别采用代码随机位置大小生成和直接拖拽的方式产生多个视图；
   2. 采用简单的动画进行移动；
   3. 给视图加上阴影(layer)；
   4. 可全部清空子视图；
   5. 视图支持手势（pan移动、tap删除、pinch缩放、rotation旋转）；

提示：Pinch的scale属性可用于调整frame

rotation需要用transform属性实现

* 程序代码：

//

// AppDelegate.swift

// 9

//

// Created by student on 2018/11/21.

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

//

import UIKit

@UIApplicationMain

class AppDelegate: UIResponder, UIApplicationDelegate {

var window: UIWindow?

func application(\_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool {

// Override point for customization after application launch.

return true

}

func applicationWillResignActive(\_ application: UIApplication) {

// Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.

// Use this method to pause ongoing tasks, disable timers, and invalidate graphics rendering callbacks. Games should use this method to pause the game.

}

func applicationDidEnterBackground(\_ application: UIApplication) {

// Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.

// If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.

}

func applicationWillEnterForeground(\_ application: UIApplication) {

// Called as part of the transition from the background to the active state; here you can undo many of the changes made on entering the background.

}

func applicationDidBecomeActive(\_ application: UIApplication) {

// Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.

}

func applicationWillTerminate(\_ application: UIApplication) {

// Called when the application is about to terminate. Save data if appropriate. See also applicationDidEnterBackground:.

}

}

//

// ViewController.swift

// 9

//

// Created by student on 2018/11/21.

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

//

import UIKit

class ViewController: UIViewController {

override func viewDidLoad() {

super.viewDidLoad()

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

}

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

let x=Int(arc4random())%Int(view.bounds.width)

let y=Int(arc4random())%Int(view.bounds.height)

let lable = UILabel(frame: CGRect(x: x, y: y, width: 30, height: 30))

lable.text="a"

lable.textAlignment = .center

lable.backgroundColor=UIColor.green

view.addSubview(lable)

lable.layer.shadowColor=UIColor.gray.cgColor

lable.layer.shadowOffset=CGSize(width: 5, height: 5)

lable.layer.shadowOpacity=1

lable.isUserInteractionEnabled=true

let panr=UIPanGestureRecognizer(target: self, action: #selector(pan(recongnizer:)))

lable.addGestureRecognizer(panr) }

@objc func pan(recongnizer:UIPanGestureRecognizer){

switch recongnizer.state {

case .changed:

fallthrough

case .ended:

let translation=recongnizer.translation(in: self.view)

recongnizer.view?.center.x+=translation.x

recongnizer.view?.center.y+=translation.y

recongnizer.setTranslation(.zero, in: self.view)

default:

break

}

}

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

for lable in view.subviews{

if lable is UILabel{

UIView.animate(withDuration: 1) {

let x=Int(arc4random())%Int(self.view.bounds.width)

let y=Int(arc4random())%Int(self.view.bounds.height)

lable.center.x=CGFloat(x)

lable.center.y=CGFloat(y)

}

}

}

}

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

for lable in view.subviews{

if lable is UILabel{

lable.removeFromSuperview()

}

}

}

}

//

// CircleView.swift

// 9

//

// Created by student on 2018/11/21.

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

//

import UIKit

@IBDesignable

class CircleView: UIView {

@IBInspectable var fillcolor:UIColor?

@IBInspectable var strokecolor:UIColor?

func setup(){

let panr=UIPanGestureRecognizer(target: self, action: #selector(pan(recongnizer:)))

self.addGestureRecognizer(panr)

let pinchr=UIPinchGestureRecognizer(target: self, action: #selector(pinch(recongnizer:)))

self.addGestureRecognizer(pinchr)

let tapr=UITapGestureRecognizer(target: self, action: #selector(tap(recongnizer:)))

self.addGestureRecognizer(tapr)

tapr.numberOfTouchesRequired=1

tapr.numberOfTouchesRequired=2

}

@objc func tap(recongnizer:UITapGestureRecognizer){

switch recongnizer.state {

case .recognized:

print("double tap")

default:

break

}

}

@objc func pan(recongnizer:UIPanGestureRecognizer){

switch recongnizer.state {

case .changed:

fallthrough

case .ended:

let translation=recongnizer.translation(in: self)

center.x+=translation.x

center.y+=translation.y

recongnizer.setTranslation(.zero, in: self)

default:

break

}

}

@objc func pinch(recongnizer:UIPinchGestureRecognizer){

switch recongnizer.state {

case .changed:

fallthrough

case .ended:

bounds.size=CGSize(width: bounds.width\*recongnizer.scale, height: bounds.height\*recongnizer.scale)

recongnizer.scale=1

default:

break

}

}

override init(frame: CGRect) {

super.init(frame: frame)

setup()

}

required init?(coder aDecoder: NSCoder) {

super.init(coder: aDecoder)

setup()

}

override func draw(\_ rect: CGRect) {

let path=UIBezierPath(ovalIn: rect)

fillcolor?.setFill()

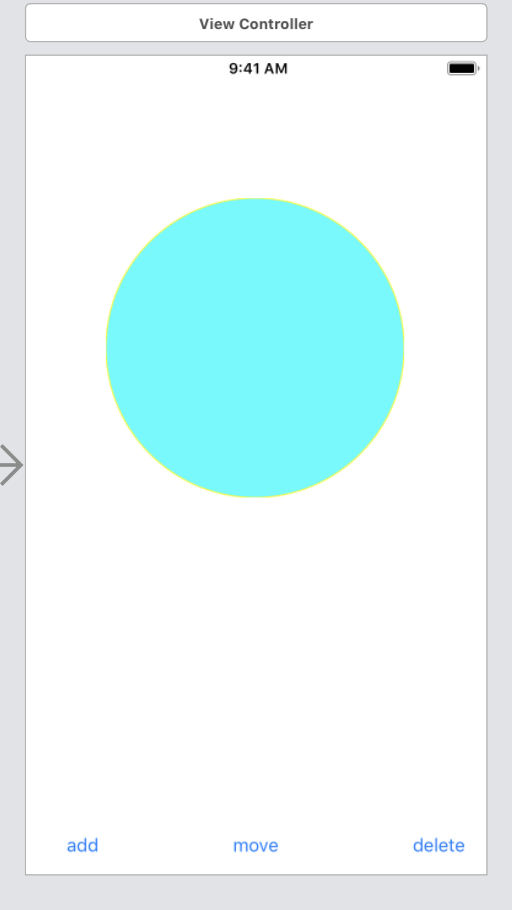
strokecolor?.setStroke()

path.fill()

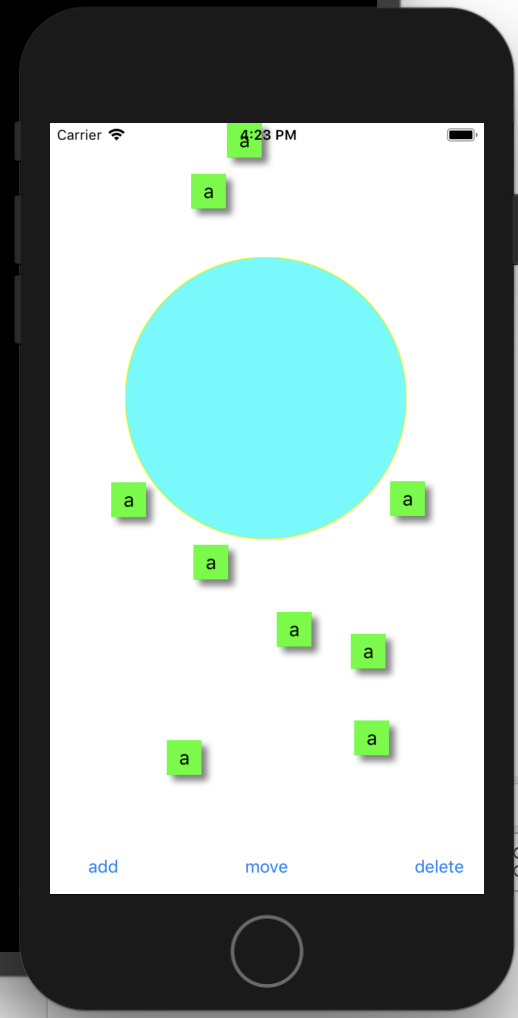
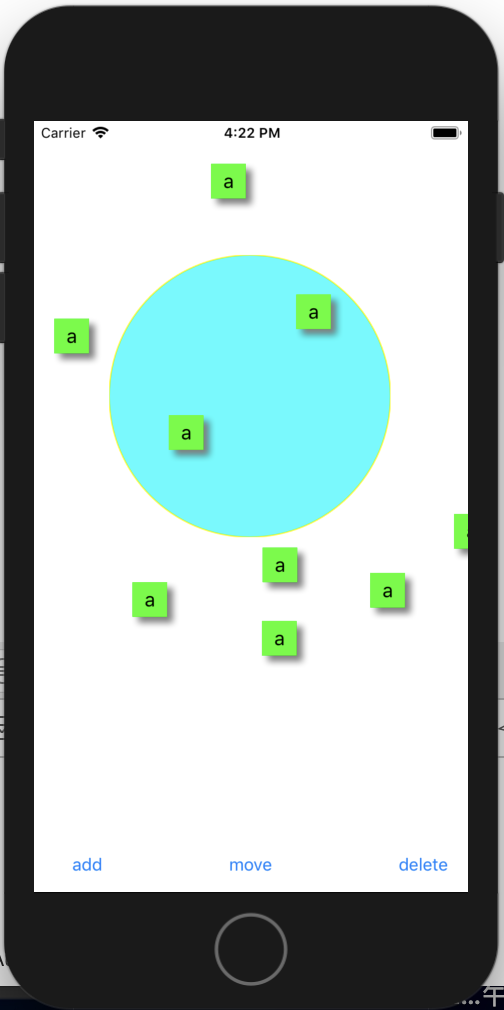
path.stroke()

}

}



* 运行结果：



1. 实现UIAlertController交互
   1. 显示ActionSheet并进行交互；
   2. 显示Login Alert并进行交互；

* 程序代码：

//

// AppDelegate.swift

// myalert

//

// Created by student on 2018/11/21.

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

//

import UIKit

@UIApplicationMain

class AppDelegate: UIResponder, UIApplicationDelegate {

var window: UIWindow?

func application(\_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool {

// Override point for customization after application launch.

return true

}

func applicationWillResignActive(\_ application: UIApplication) {

// Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.

// Use this method to pause ongoing tasks, disable timers, and invalidate graphics rendering callbacks. Games should use this method to pause the game.

}

func applicationDidEnterBackground(\_ application: UIApplication) {

// Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.

// If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.

}

func applicationWillEnterForeground(\_ application: UIApplication) {

// Called as part of the transition from the background to the active state; here you can undo many of the changes made on entering the background.

}

func applicationDidBecomeActive(\_ application: UIApplication) {

// Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.

}

func applicationWillTerminate(\_ application: UIApplication) {

// Called when the application is about to terminate. Save data if appropriate. See also applicationDidEnterBackground:.

}

}

//

// ViewController.swift

// myalert

//

// Created by student on 2018/11/21.

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

//

import UIKit

class ViewController: UIViewController {

override func viewDidLoad() {

super.viewDidLoad()

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

}

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

let alert=UIAlertController(title: "action sheet", message: "choose color", preferredStyle: .actionSheet)

alert.addAction(UIAlertAction(title: "red", style: .default, handler: { (action) in

self.view.backgroundColor=UIColor.red

}))

alert.addAction(UIAlertAction(title: "green", style: .default, handler: { (action) in

self.view.backgroundColor=UIColor.green

}))

alert.addAction(UIAlertAction(title: "blue", style: .default, handler: { (action) in

self.view.backgroundColor=UIColor.blue

}))

alert.addAction(UIAlertAction(title: "yelow", style: .default, handler: { (action) in

self.view.backgroundColor=UIColor.yellow

}))

present(alert,animated: true,completion: nil)

}

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

let alert=UIAlertController(title: "alert", message: "userinfo", preferredStyle: .alert)

alert.addAction(UIAlertAction(title: "Login", style: .default, handler: { (action) in

guard let username = alert.textFields?.first?.text,let password=alert.textFields?.last?.text else {

return

}

print("username=\(username) password=\(password)")

}))

alert.addAction(UIAlertAction(title: "Cancel", style: .cancel, handler: { (action) in

}))

alert.addTextField { (textField) in

textField.placeholder="username"

}

alert.addTextField { (textField) in

textField.placeholder="password"

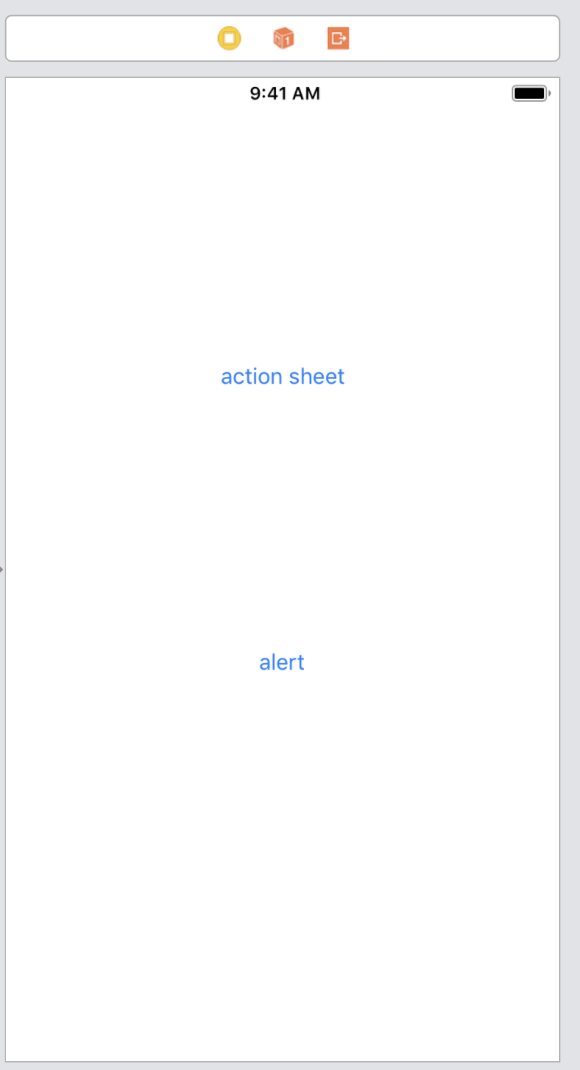
textField.isSecureTextEntry=true

}

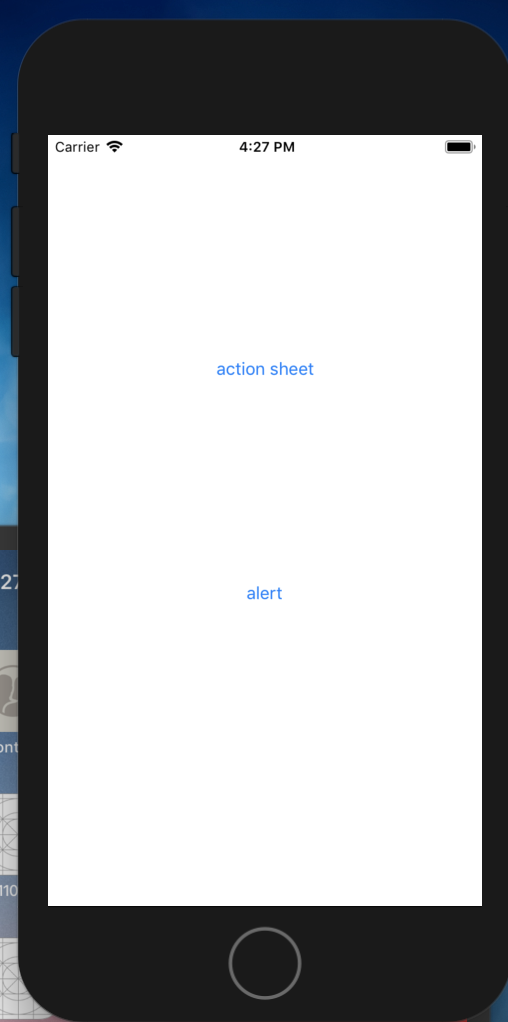
present(alert,animated: true,completion: nil)

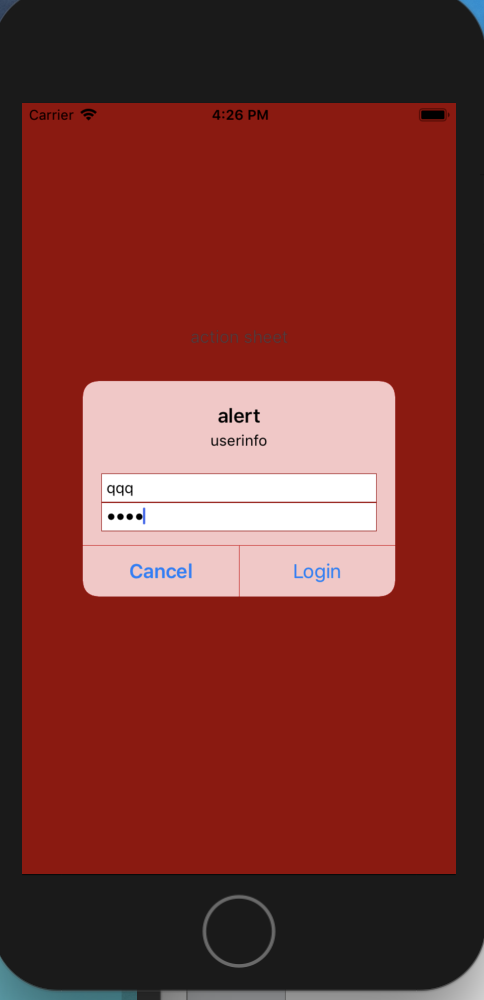
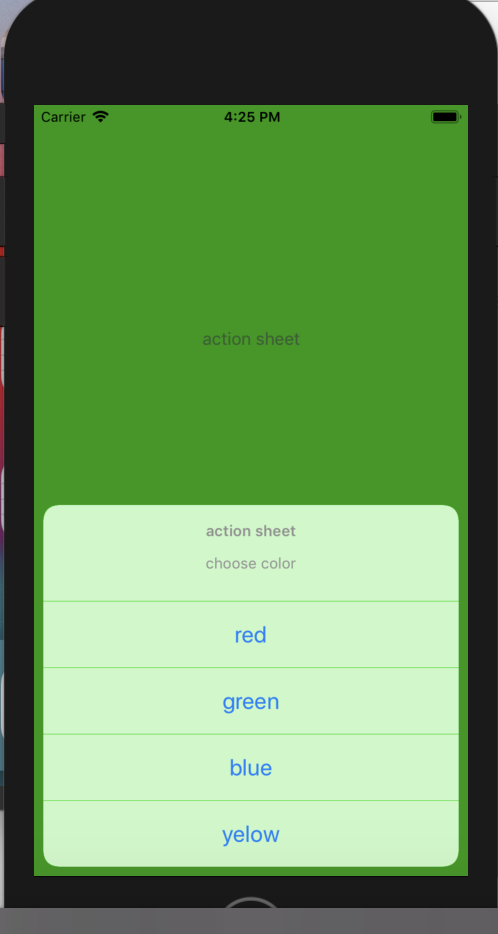
}

}



* 运行结果：





1. 一个界面使用两个scrollView
   1. 在一个scrollView中可进行多张图片横屏滚动浏览(相册)，需要有pagecontrol进行提示；
   2. 在另一个scrollView中可放大缩小；

提示：需用delegate

* 程序代码：

//

// AppDelegate.swift

// myscollview

//

// Created by student on 2018/11/21.

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

//

import UIKit

@UIApplicationMain

class AppDelegate: UIResponder, UIApplicationDelegate {

var window: UIWindow?

func application(\_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool {

// Override point for customization after application launch.

return true

}

func applicationWillResignActive(\_ application: UIApplication) {

// Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.

// Use this method to pause ongoing tasks, disable timers, and invalidate graphics rendering callbacks. Games should use this method to pause the game.

}

func applicationDidEnterBackground(\_ application: UIApplication) {

// Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.

// If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.

}

func applicationWillEnterForeground(\_ application: UIApplication) {

// Called as part of the transition from the background to the active state; here you can undo many of the changes made on entering the background.

}

func applicationDidBecomeActive(\_ application: UIApplication) {

// Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.

}

func applicationWillTerminate(\_ application: UIApplication) {

// Called when the application is about to terminate. Save data if appropriate. See also applicationDidEnterBackground:.

}

}

//

// ViewController.swift

// myscollview

//

// Created by student on 2018/11/21.

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

//

import UIKit

class ViewController: UIViewController,UIScrollViewDelegate {

@IBOutlet weak var scrollView: UIScrollView!

@IBOutlet weak var pageControll: UIPageControl!

override func viewDidLoad() {

super.viewDidLoad()

// scrollView.minimumZoomScale = 0.2

// scrollView.maximumZoomScale = 5

scrollView.delegate = self

for i in 1...7 {

let imageView = UIImageView(image: UIImage(named: "\(i)"))

imageView.contentMode = .scaleAspectFit

imageView.frame = CGRect(x: CGFloat(i-1) \* scrollView.bounds.width, y: 0, width: scrollView.bounds.width, height: scrollView.bounds.height)

scrollView.addSubview(imageView)

}

scrollView.contentSize = CGSize(width: scrollView.bounds.width \* 7, height: scrollView.bounds.height)

scrollView.isPagingEnabled = true

scrollView.showsHorizontalScrollIndicator = false

pageControll.numberOfPages = 7

pageControll.currentPage = 0

}

@IBAction func pageControllClicked(\_ sender: UIPageControl) {

let currentPage = sender.currentPage

let rect = CGRect(x: CGFloat(currentPage) \* scrollView.bounds.width, y: 0, width: scrollView.bounds.width, height: scrollView.bounds.height)

scrollView.scrollRectToVisible(rect, animated: true)

}

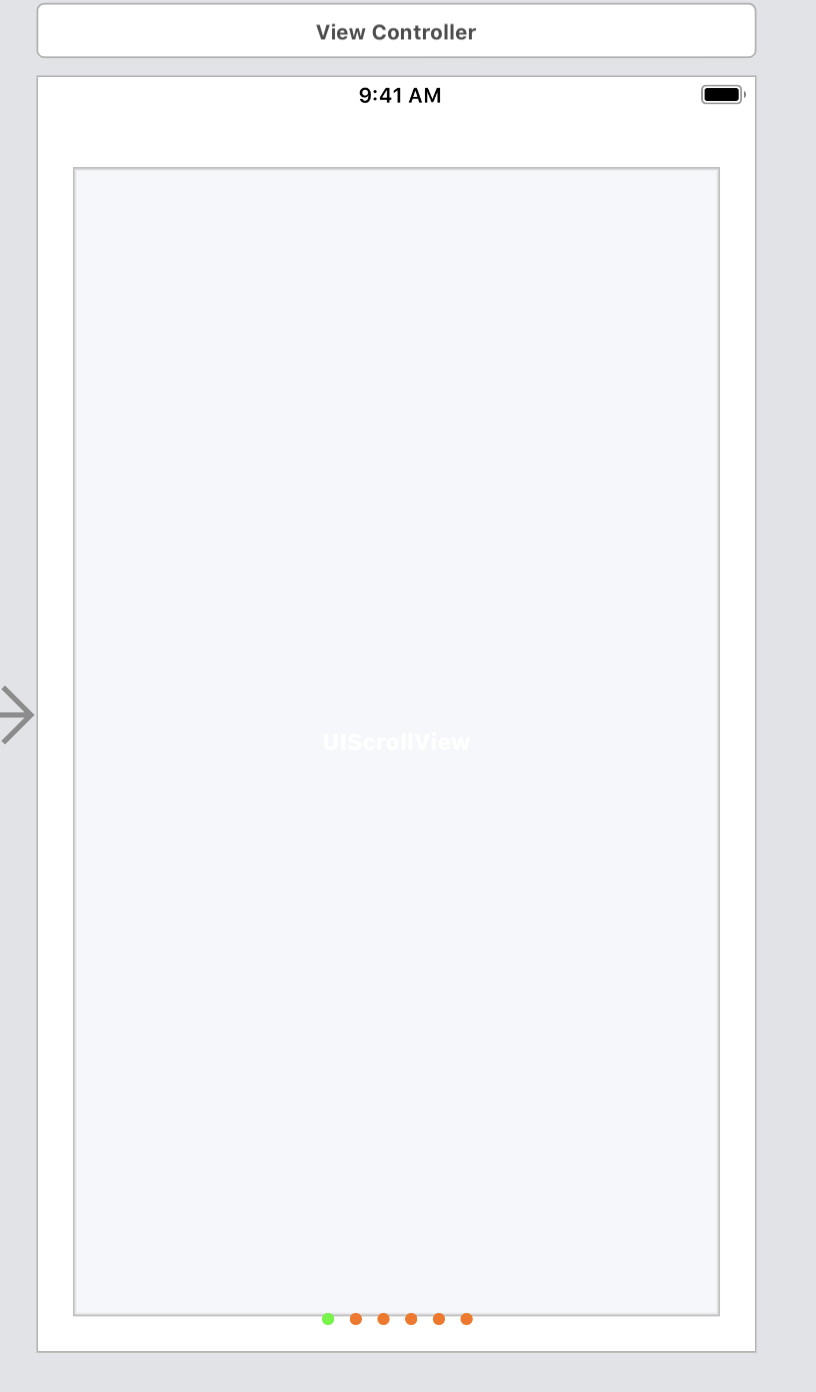
func scrollViewDidEndDecelerating(\_ scrollView: UIScrollView) {

let currentPage = scrollView.contentOffset.x/scrollView.bounds.width

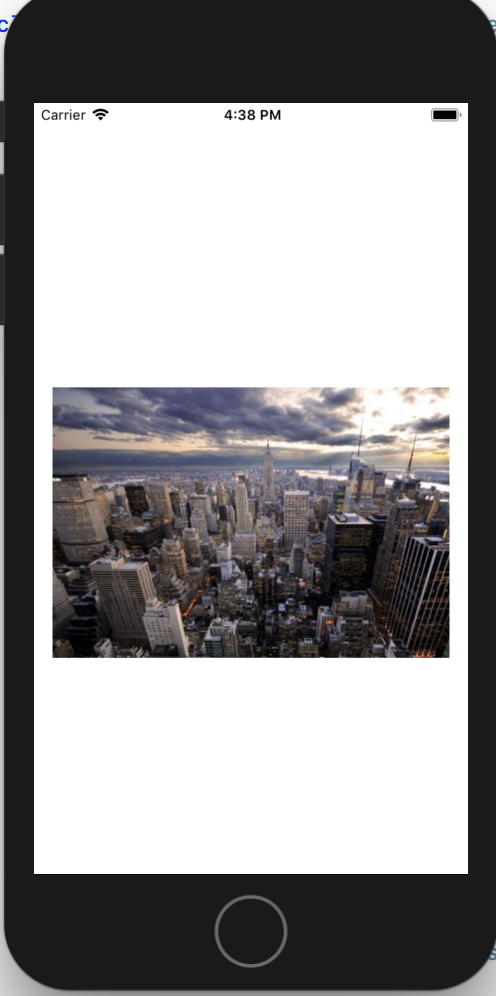
pageControll.currentPage = Int(currentPage)

}

}



* 运行结果：



1. 实验结果的分析与评价（该部分如不够填写，请另加附页）

这一章的内容还比较好玩，我也是跟着老师的录屏敲了一遍代码，自己对这一章的理解还是比较好，对于我来说不是很难。

Github地址：

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。