实验编号： 8 **四川师大《IOS》实验报告 2018** 年 **10** 月 **31**日

### **计算机科学学院** 2016 级 4 班 实验名称： TextField和TableView \_

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

**实验\_一\_ \_\_\_\_\_\_** Swift类和子类**\_\_\_\_\_\_\_\_**

1. 实验目的及要求
2. 掌握TextField和Delegate、Notification的使用；
3. 掌握表视图TableView以及Datasource、Delegate等的使用；
4. 实验内容
5. 将之前作业实现的Person 类和子类导入新项目，新建多个 Student 和 Teacher 对象并放入一数组中；
6. 数组排序后采用 TableView 的形式显示所有Student和Teacher对象的信息，要求每一个对象一个Cell，同时学生和教师显示不同的信息，选择一个Cell 后给出选择反馈；
7. 修改界面，增加添加新学生的部分(键盘要正确显示和消失)，要求实现TableView的增删改;

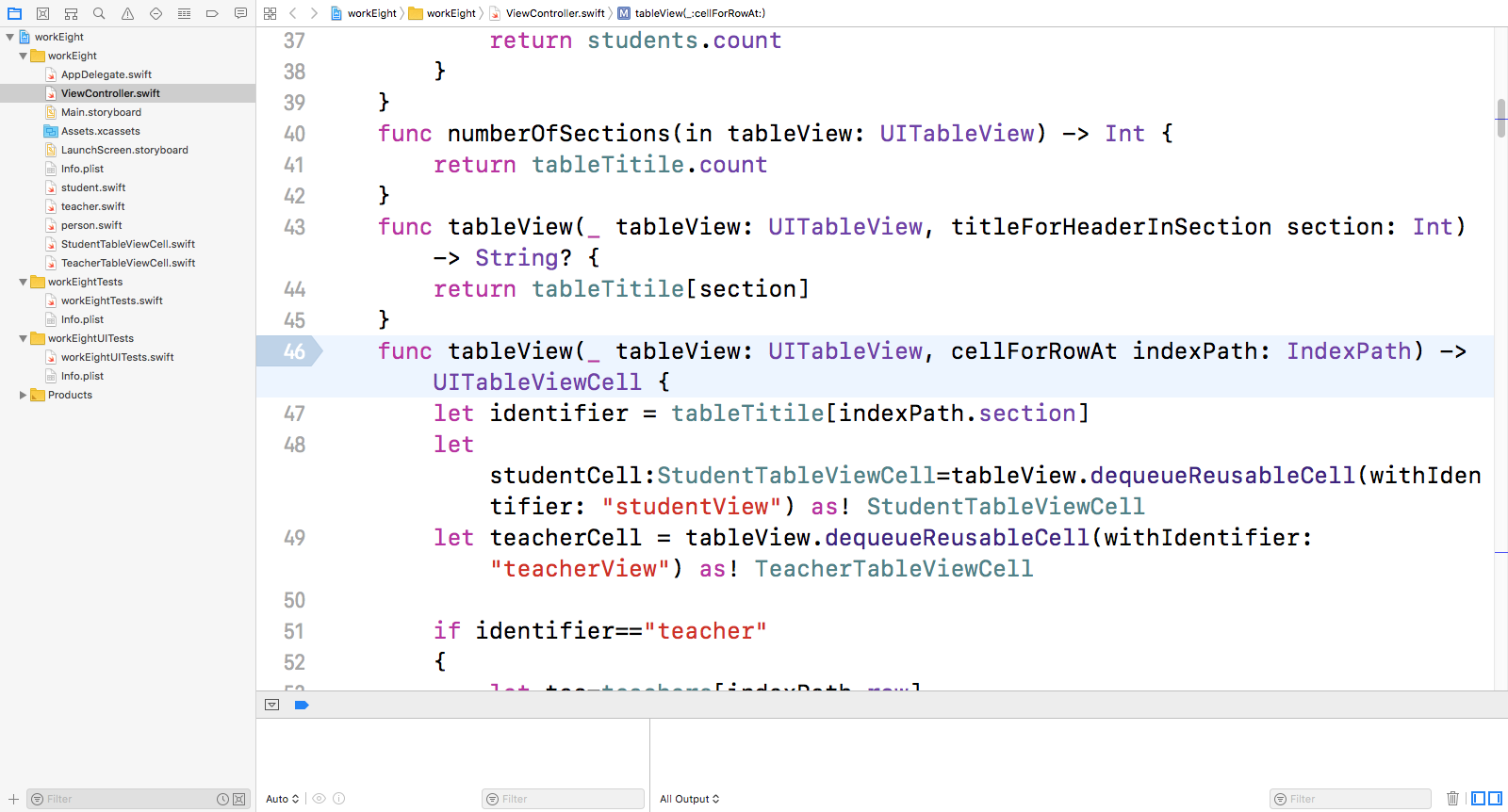
说明：键盘显示消失需要基于Notification 实现自适应键盘的界面布局;注册KeyboardDidShow和KeyBoardWillHide通知，并处理相应的action，进行界面的布局调整；

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. 将之前作业实现的Person 类和子类导入新项目，新建多个 Student 和 Teacher 对象并放入一数组中；
3. 数组排序后采用 TableView 的形式显示所有Student和Teacher对象的信息，要求每一个对象一个Cell，同时学生和教师显示不同的信息，选择一个Cell 后给出选择反馈；
4. 修改界面，增加添加新学生的部分(键盘要正确显示和消失)，要求实现TableView的增删改;

说明：键盘显示消失需要基于Notification 实现自适应键盘的界面布局;注册KeyboardDidShow和KeyBoardWillHide通知，并处理相应的action，进行界面的布局调整；

* 程序代码：

文件列表



//

// ViewController.swift

// workEight

//

// Created by jiang on 2018/11/11.

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

//

import UIKit

class ViewController: UIViewController,UITableViewDelegate,UITableViewDataSource{

@IBOutlet weak var MyTableView: UITableView!

@IBOutlet weak var chooseLable: UILabel!

var tableTitile=["teacher","student"]

var images=["baiyun","chengxi","furong","jinping","nangang","qilihe","shangjie","wuhou"]

var students=[Student]()

var teachers=[Teacher]()

let stuONe=Student(firstName: "宇童", lastName: "蒋", age: 20, gender: .female, stuNo: "0001")

let stuTwo=Student(firstName: "童宇", lastName: "蒋", age: 20, gender: .female, stuNo: "0002")

let stuThree=Student(firstName: "蒋宇", lastName: "童", age: 20, gender: .female, stuNo: "0003")

let stuFour=Student(firstName: "蒋童", lastName: "宇", age: 20, gender: .female, stuNo: "0004")

let teacOne=Teacher(firstName: "贵洋", lastName: "李", age: 40, gender: .male, title: "教授")

let teacTwo=Teacher(firstName: "路", lastName: "张", age: 41, gender: .female, title: "副教授")

let teacThree=Teacher(firstName: "颜值 ", lastName: "高", age: 42, gender: .male, title: "副教授")

let teacFour=Teacher(firstName: "良", lastName: "谭", age: 43, gender: .female, title: "教授")

let addStudentAlert=UIAlertController(title: "添加学生", message:nil, preferredStyle: .alert)

let addTeacherAlert=UIAlertController(title: "添加老师", message: nil, preferredStyle: .alert)

let editStudentAlertOne=UIAlertController(title: "修改id", message: nil, preferredStyle: .alert)

let editStudentAlertTwo=UIAlertController(title: "修改fristName", message: nil, preferredStyle: .alert)

let editTeacherAlertOne=UIAlertController(title: "修改职称", message: nil, preferredStyle: .alert)

let editTeacherAlertTwo=UIAlertController(title: "修改fristName", message: nil, preferredStyle: .alert)

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

if section == 0 {

return teachers.count

} else {

return students.count

}

}

func numberOfSections(in tableView: UITableView) -> Int {

return tableTitile.count

}

func tableView(\_ tableView: UITableView, titleForHeaderInSection section: Int) -> String? {

return tableTitile[section]

}

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

let identifier = tableTitile[indexPath.section]

let studentCell:StudentTableViewCell=tableView.dequeueReusableCell(withIdentifier: "studentView") as! StudentTableViewCell

let teacherCell = tableView.dequeueReusableCell(withIdentifier: "teacherView") as! TeacherTableViewCell

if identifier=="teacher"

{

let tec=teachers[indexPath.row]

teacherCell.teacherFirstNmae.text = tec.lastName+tec.firstName

teacherCell.teacherTitleLable.text=tec.title

teacherCell.teacherImage.image=UIImage(named: images[indexPath.row%7])

teacherCell.teacherImage.layer.cornerRadius=teacherCell.teacherImage.frame.size.width/2

teacherCell.teacherImage.clipsToBounds=true

return teacherCell

}

else

{

let stu=students[indexPath.row]

studentCell.studentId.text=stu.stuNo

studentCell.studentFirstName.text=stu.lastName+stu.firstName

studentCell.studentImge.image=UIImage(named: images[7-indexPath.row%7])

studentCell.studentImge.layer.cornerRadius=studentCell.studentImge.frame.size.width/2

studentCell.studentImge.clipsToBounds=true

return studentCell

}

}

override func viewDidLoad() {

super.viewDidLoad()

students.append(stuONe)

students.append(stuTwo)

students.append(stuThree)

students.append(stuFour)

teachers.append(teacOne)

teachers.append(teacTwo)

teachers.append(teacThree)

teachers.append(teacFour)

addStudentAlert.addTextField{(textField) in

textField.placeholder="输入您的ID"

}

addStudentAlert.addTextField{(textField) in

textField.placeholder="输入输入你的firstName"

}

addStudentAlert.addTextField{(textField) in

textField.placeholder="输入输入你的lastName"

}

addStudentAlert.addTextField{(textField) in

textField.placeholder="输入您的性别"

}

addStudentAlert.addTextField{(textField) in

textField.placeholder="输入你的年龄"

}

addStudentAlert.addAction(UIAlertAction(title: "添加", style: .default, handler:{(action) in self.studentAdd() }))

addStudentAlert.addAction(UIAlertAction(title: "取消", style: .default, handler:nil))

editTeacherAlertOne.addTextField{(textField) in

textField.placeholder="请输入修改后的职称"

}

editTeacherAlertTwo.addTextField{(textField) in

textField.placeholder="请输入修改后的fristName"

}

editTeacherAlertOne.addAction(UIAlertAction(title: "确定", style: .default, handler: {(action)

in self.editTeacherOne()

}))

editTeacherAlertOne.addAction(UIAlertAction(title: "取消", style: .default, handler: nil

))

editTeacherAlertTwo.addAction(UIAlertAction(title: "确定", style: .default, handler: {(action)

in self.editTeacherTwo()

}))

editTeacherAlertTwo.addAction(UIAlertAction(title: "取消", style: .default, handler: nil

))

editStudentAlertOne.addTextField{(textField) in

textField.placeholder="请输入修改后的职称"

}

editStudentAlertTwo.addTextField{(textField) in

textField.placeholder="请输入修改后的fristName"

}

editStudentAlertOne.addAction(UIAlertAction(title: "确定", style: .default, handler: {(action)

in self.editStudentOne()

}))

editStudentAlertOne.addAction(UIAlertAction(title: "取消", style: .default, handler: nil

))

editStudentAlertTwo.addAction(UIAlertAction(title: "确定", style: .default, handler: {(action)

in self.editStudentTwo()

}))

editStudentAlertTwo.addAction(UIAlertAction(title: "取消", style: .default, handler: nil

))

addTeacherAlert.addTextField{(textField) in

textField.placeholder="输入您的职称"

}

addTeacherAlert.addTextField{(textField) in

textField.placeholder="输入输入你的firstName"

}

addTeacherAlert.addTextField{(textField) in

textField.placeholder="输入输入你的lastName"

}

addTeacherAlert.addTextField{(textField) in

textField.placeholder="输入您的性别"

}

addTeacherAlert.addTextField{(textField) in

textField.placeholder="输入你的年龄"

}

addTeacherAlert.addAction(UIAlertAction(title: "添加", style: .default, handler:{(action) in self.teacherAdd()}))

addTeacherAlert.addAction(UIAlertAction(title: "取消", style: .default, handler:nil))

}

func editTeacherOne()

{

guard let zhicheng=editTeacherAlertOne.textFields?.first?.text

else

{

return

}

teachers[(MyTableView.indexPathForSelectedRow?.row)!].title=zhicheng

MyTableView.reloadData()

}

func editTeacherTwo()

{

guard let firstName=editTeacherAlertTwo.textFields?.first?.text

else

{

return

}

teachers[(MyTableView.indexPathForSelectedRow?.row)!].firstName=firstName

MyTableView.reloadData()

}

func editStudentOne()

{

guard let id=editStudentAlertOne.textFields?.first?.text else {

return

}

students[(MyTableView.indexPathForSelectedRow?.row)!].stuNo=id

MyTableView.reloadData()

}

func editStudentTwo()

{

guard let firstName=editStudentAlertTwo.textFields?.first?.text else {

return

}

students[(MyTableView.indexPathForSelectedRow?.row)!].firstName=firstName

MyTableView.reloadData()

}

func studentAdd() {

guard let studentId = addStudentAlert.textFields![0].text, let studentFristName = addStudentAlert.textFields![1].text, let studentLastName = addStudentAlert.textFields![2].text,let age = Int(addStudentAlert.textFields![4].text!) else {

return

}

let studentgender: Gender

switch addStudentAlert.textFields![3].text! {

case "男":

studentgender = .male

case "女":

studentgender = .female

default:

studentgender = .female

}

let student = Student(firstName: studentFristName, lastName: studentLastName, age: age, gender: studentgender, stuNo:studentId)

addStudentAlert.textFields![0].resignFirstResponder()

addStudentAlert.textFields![1].resignFirstResponder()

addStudentAlert.textFields![2].resignFirstResponder()

addStudentAlert.textFields![3].resignFirstResponder()

students.append(student)

MyTableView.reloadData()

}

func teacherAdd() {

guard let teacherTitle = addTeacherAlert.textFields![0].text, let teacherFristName = addTeacherAlert.textFields![1].text, let teacherLastName = addTeacherAlert.textFields![2].text,let age = Int(addTeacherAlert.textFields![4].text!) else {

return

}

let teachergender: Gender

switch addTeacherAlert.textFields![3].text! {

case "男":

teachergender = .male

case "女":

teachergender = .female

default:

teachergender = .female

}

let teacher = Teacher(firstName: teacherFristName, lastName: teacherLastName, age: age, gender: teachergender, title:teacherTitle)

addTeacherAlert.textFields![0].resignFirstResponder()

addTeacherAlert.textFields![1].resignFirstResponder()

addTeacherAlert.textFields![2].resignFirstResponder()

addTeacherAlert.textFields![3].resignFirstResponder()

teachers.append(teacher)

MyTableView.reloadData()

}

override func didReceiveMemoryWarning() {

super.didReceiveMemoryWarning()

// Dispose of any resources that can be recreated.

}

@IBAction func addBtn(\_ sender: UIButton) {

let alert=UIAlertController(title: nil, message: nil, preferredStyle:.actionSheet)

alert.addAction(UIAlertAction(title: "添加老师", style: .destructive, handler:{ (action) in

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

}))

alert.addAction(UIAlertAction(title: "添加学生", style: .destructive, handler:{ (action) in

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

}))

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

}

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

guard let section=MyTableView.indexPathForSelectedRow?.section else {

return

}

if section==0

{

let alert = UIAlertController(title: nil, message: nil, preferredStyle: .actionSheet)

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

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

}))

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

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

}))

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

}

else

{

let alert = UIAlertController(title: nil, message: nil, preferredStyle: .actionSheet)

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

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

}))

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

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

}))

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

}

}

@IBAction func deleteBtn(\_ sender: UIButton) {

MyTableView.isEditing = !MyTableView.isEditing

}

func tableView(\_ tableView: UITableView, editingStyleForRowAt indexPath: IndexPath) -> UITableViewCellEditingStyle {

return .delete

}

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

if editingStyle == UITableViewCellEditingStyle.delete {

if indexPath.section == 0 {

teachers.remove(at: indexPath.row)

} else {

students.remove(at: indexPath.row)

}

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

}

}

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

let identifier = tableTitile[indexPath.section]

if identifier=="teacher"

{

chooseLable.text="You choose Teacher\(teachers[indexPath.row].lastName)\(teachers[indexPath.row].firstName)"

}

else

{

chooseLable.text="You choose Student\(students[indexPath.row].lastName)\(students[indexPath.row].firstName)"

}

}

func tableView(\_ tableView: UITableView, canMoveRowAt indexPath: IndexPath) -> Bool {

return true

}

func tableView(\_ tableView: UITableView, moveRowAt sourceIndexPath: IndexPath, to destinationIndexPath: IndexPath) {

if sourceIndexPath.section != destinationIndexPath.section {

tableView.reloadData()

} else {

if sourceIndexPath.section == 0 {

teachers.insert(teachers.remove(at: sourceIndexPath.row), at: destinationIndexPath.row)

} else {

students.insert(students.remove(at: sourceIndexPath.row), at: destinationIndexPath.row)

}

}

}

}

//

// StudentTableViewCell.swift

// workEight

//

// Created by jiang on 2018/11/12.

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

//

import UIKit

class StudentTableViewCell: UITableViewCell {

@IBOutlet weak var studentImge: UIImageView!

@IBOutlet weak var studentFirstName: UILabel!

@IBOutlet weak var studentId: UILabel!

override func awakeFromNib() {

super.awakeFromNib()

// Initialization code

}

override func setSelected(\_ selected: Bool, animated: Bool) {

super.setSelected(selected, animated: animated)

// Configure the view for the selected state

}

}

//

// TeacherTableViewCell.swift

// workEight

//

// Created by jiang on 2018/11/12.

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

//

import UIKit

class TeacherTableViewCell: UITableViewCell {

@IBOutlet weak var teacherTitleLable: UILabel!

@IBOutlet weak var teacherImage: UIImageView!

@IBOutlet weak var teacherFirstNmae: UILabel!

override func awakeFromNib() {

super.awakeFromNib()

// Initialization code

}

override func setSelected(\_ selected: Bool, animated: Bool) {

super.setSelected(selected, animated: animated)

// Configure the view for the selected state

}

}

//

// person.swift

// workEight

//

// Created by jiang on 2018/11/12.

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

//

import Foundation

enum Gender:Int {

case male,female

static func >(gender1:Gender,gender2:Gender)->Bool

{

return gender1.rawValue>gender2.rawValue

}

}

class Person: CustomStringConvertible

{

var firstName:String = ""

var lastName:String = ""

var age:Int = 0

var gender:Gender

var fullName:String

{

get

{

return self.firstName+" "+self.lastName

}

}

init(firstName:String, lastName:String,age:Int,gender:Gender)

{

self.firstName=firstName

self.lastName=lastName

self.age=age

self.gender=gender

}

convenience init(name:String,age:Int)

{

self.init(firstName:name,lastName:"",age:age,gender:Gender.female)

}

static func ==(p1:Person,p2:Person)->Bool

{

return p1.fullName==p2.fullName&&p1.age==p2.age

}

static func != (p1:Person,p2:Person)->Bool

{

return p1.fullName != p2.fullName||p1.age != p2.age

}

var description: String {

return "fullName: \(self.fullName), age: \(self.age), gender: \(self.gender)"

}

}

//

// teacher.swift

// workEight

//

// Created by jiang on 2018/11/12.

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

//

import Foundation

class Teacher:Person

{

var title:String

init(firstName: String, lastName: String, age: Int, gender: Gender,title:String) {

self.title=title

super.init(firstName: firstName, lastName: lastName, age: age, gender: gender)

}

override var description: String

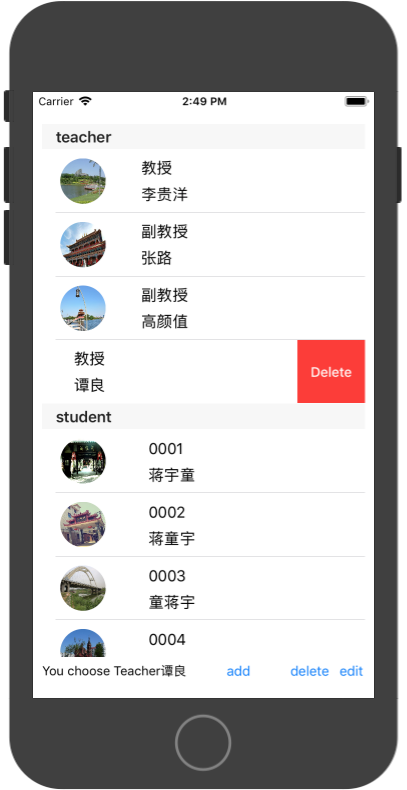
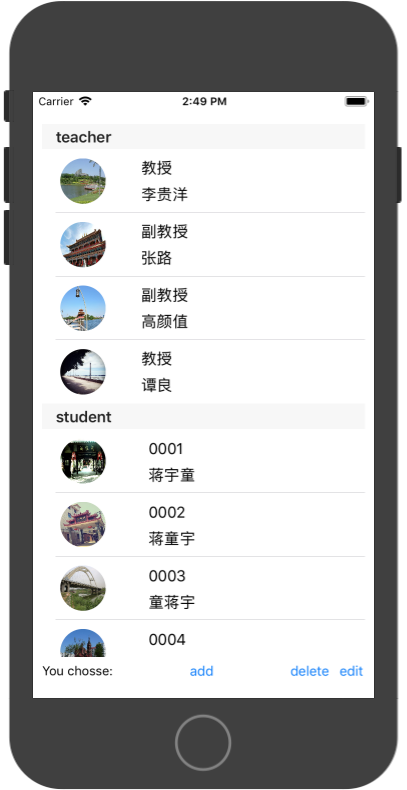
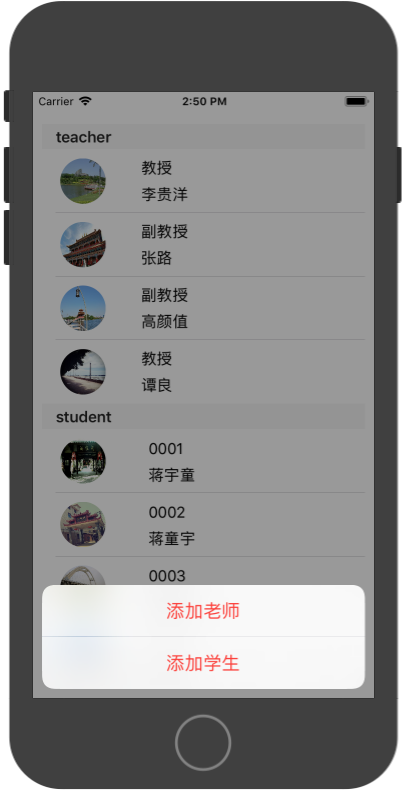
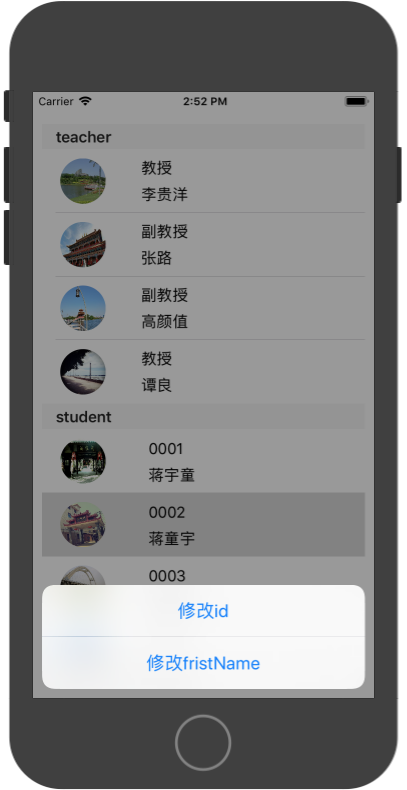
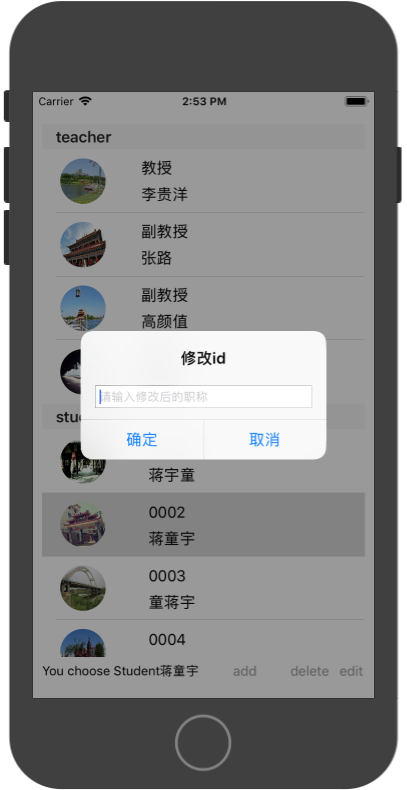
{

return "fullName: \(self.fullName), age: \(self.age), gender: \(self.gender) title:\(self.title)"

}

}

* 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）

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

这个实验是做的一个tableview的实验，这个实验我做了很久。主要是老师和学生在同一个tableview里面展示，所以就有两个section。主要是我添加列表的时候我要判断是学生还是老师，在写弹出框的时候上面都要写两份。但是这次都实验也是比较都学习。在这个实验我同时用了tableview和alert这样子我觉得可以融汇贯通。

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。