



Списки (UITableView и UICollectionView)

Занятие №5

Дмитрий Тараев

План лекции



- Отметиться!
- Повторение
- UIScrollView
- UITableView
 - UITableView data source & delegate
- UICollectionView
 - data source & delegate
 - UICollectionViewLayout
- Демо

Прошлая лекция

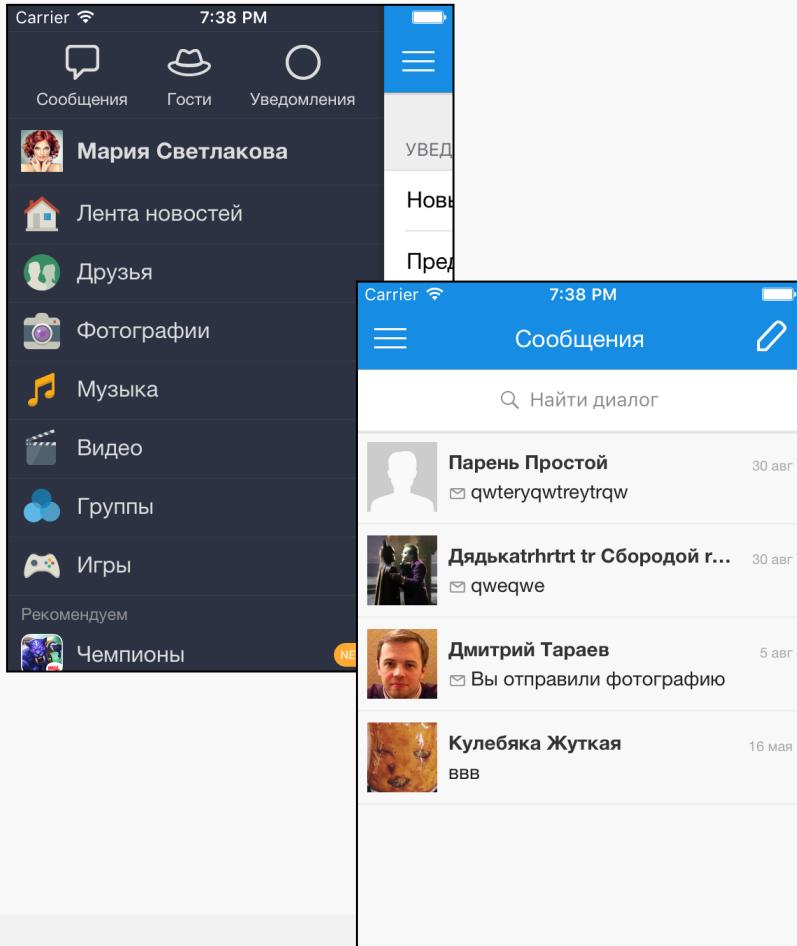


- Про Objective-C
 - посмотрели пример, переписанный на нём?
- MVC
 - Model
 - View
 - Controller
- Другие архитектурные паттерны
- Способы коммуникации между слоями
 - delegate
 - notification

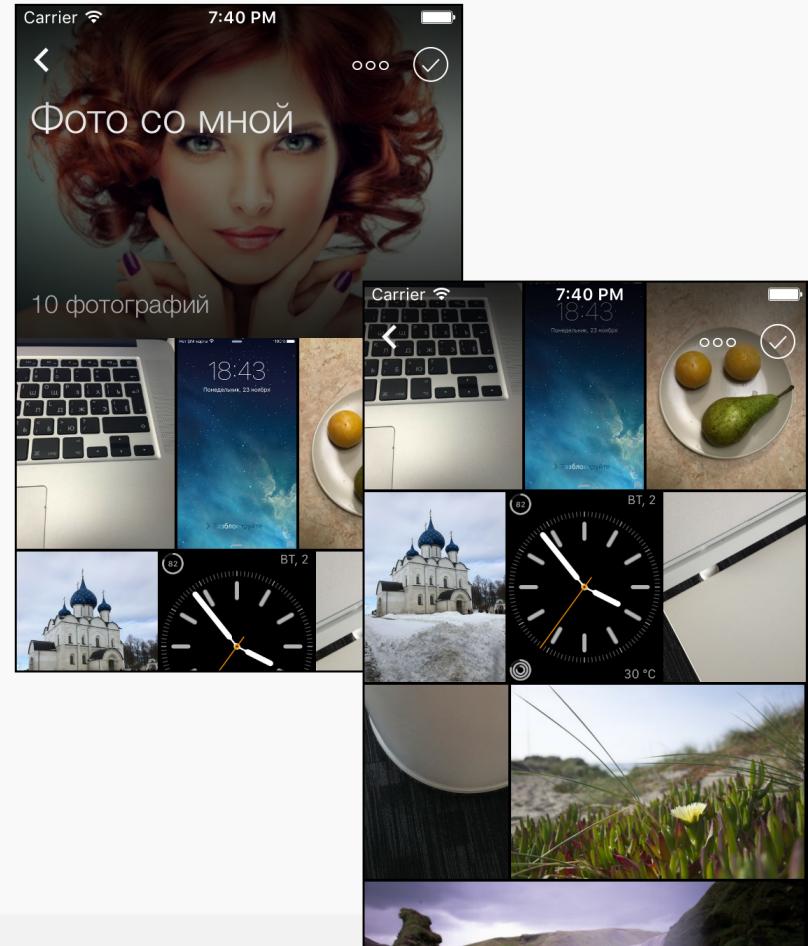
Примеры списков



UITableView



UICollectionView



UIScrollView 1/3

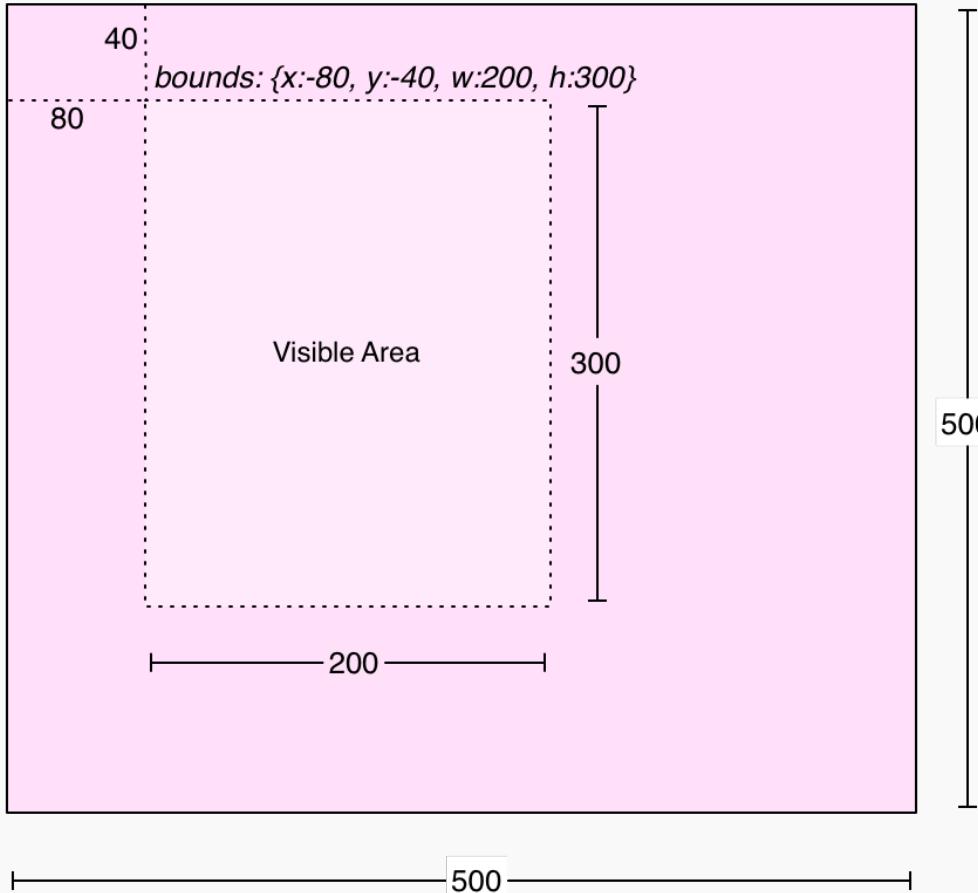


- Для контента, который не помещается на экране
- Особенности
 - масштабирование (zooming)
 - перетаскивание (panning)
- Координаты
 - frame
 - contentView
 - contentSize, contentOffset, contentInset

UIScrollView 2/3



*contentOffset: {x:80, y:40}
contentSize: {w:500, h:500}*



- **bounds**
- **contentOffset**
- **contentSize**
- **contentInset**

UIScrollView 3/3



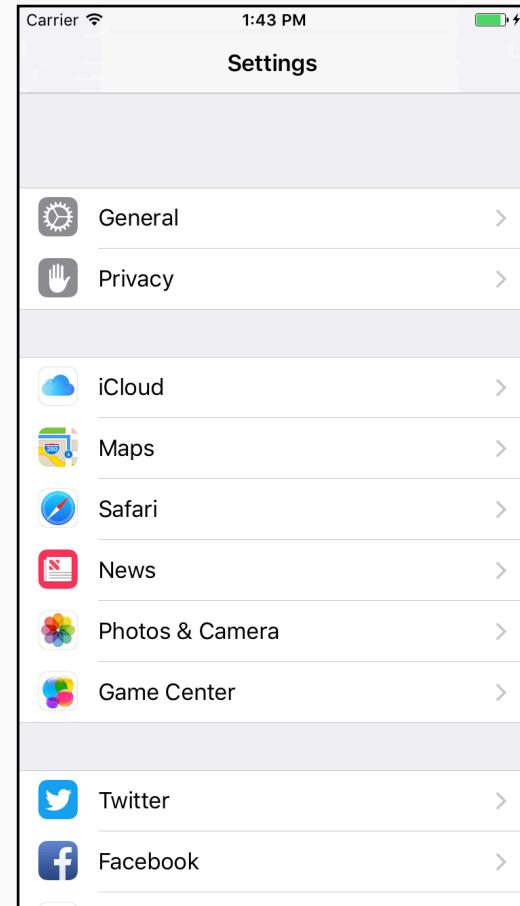
важно!

- Весь контент находится в памяти
- Применение
 - фотогалерея
 - витрина книг в iBooks

UITableView 1/10



- UITableView
 - static / dynamic
 - plain / grouped
 - разбита на секции? да / нет
 - разные виды ячеек



UITableView 2/10



- Классы:
 - UITableView
 - UITableViewCell
- Протоколы:
 - UITableViewDataSource
 - UITableViewDelegate

UITableView 3/10



- Создание списка:
 - с помощью Interface Builder
 - storyboard
 - отдельный xib-файл
 - с помощью кода

UITableView 4/10



- Data source & delegate (протоколы)
 - Data source - «что» показывается
 - Delegate - «как» показывается
- Паттерн «делегат»
 - объект, который удовлетворяет указанному протоколу, т.е. реализует объявленные методы
 - другой объект в нужные моменты вызывает эти методы у своего делегата
- Data Source (по сути, это - подвид делегата)
 - примерно то же, но эти методы возвращают данные, то есть объект, реализующий протокол, является источником данных

UITableViewController vs. UIViewController



- UITableViewController



- автоматически назначается dataSource и delegate
- уже есть **tableView** (для него она self.view)
open var tableView: UITableView!
- есть «из коробки» **pull to refresh**
var refreshControl: UIRefreshControl?



- ограничивает в реализации сложного дизайна экрана

UITableView 5/10



- Data source
 - СКОЛЬКО секций
func numberOfRows(in tableView: UITableView) -> Int
 - СКОЛЬКО ячеек в секции
func tableView(_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int
 - какая должна быть ячейка
func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell

UITableView 6/10



- Data source (другое)
 - UIView для header'а и footer'а
 - режим редактирования
 - удаление ячеек
 - вставка ячеек
 - перетаскивание ячеек



tableView(_:cellForRowAt:)



```
1. override func tableView(  
    tableView: UITableView,  
    cellForRowAt indexPath: IndexPath) -> UITableViewCell {  
2.     let cell = tableView.dequeueReusableCell(withIdentifier:  
        "CellIdentifier", for: indexPath)  
3.     cell.textLabel?.text = cellsdataArray[indexPath.row]  
4.     return cell  
5. }
```

- NSIndexPath
 - section
 - row
- cell identifier
 - или в коде, или в IB

UITableView 7/10: Self-Sizing Cells



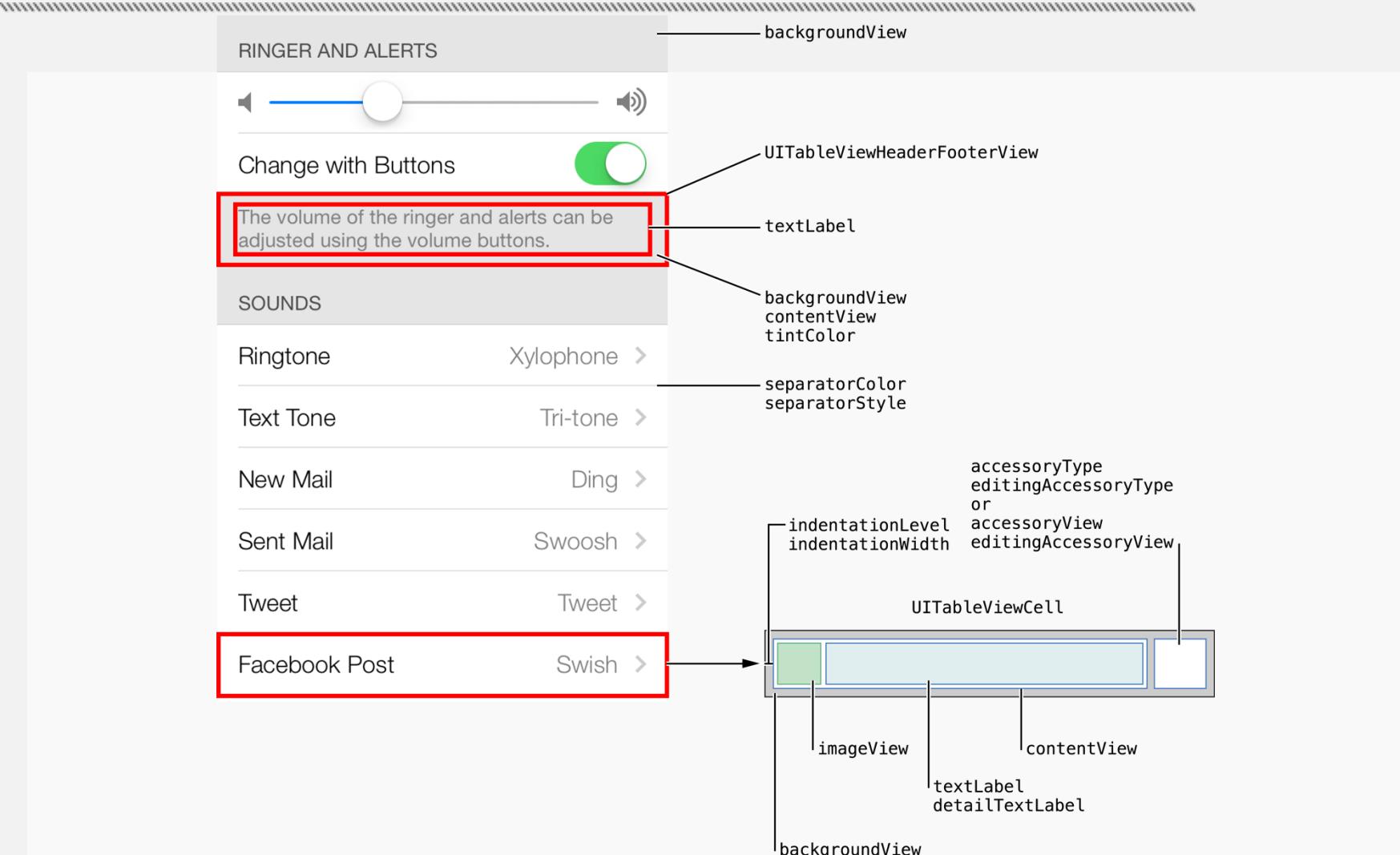
- Свойство `rowHeight` установить `UITableViewAutomaticDimension`
- Задать `estimatedRowHeight`
 - для приблизительного расчёта высоты `contentSize`
 - фактическая высота рассчитывается по мере появления ячеек на экране
- Если вдруг iOS 7
 - всё сложнее
 - очень подробно: <https://stackoverflow.com/a/18746930>

UITableView 8/10



- Delegate
 - была выбрана ячейка
`func tableView(_ tableView: UITableView, didSelectRowAt indexPath: IndexPath)`
 - нажатие на accessoryButton
`func tableView(_ tableView: UITableView, accessoryButtonTappedForRowWith indexPath: IndexPath)`
 - методы will / did
`func tableView(_ tableView: UITableView, willDisplay cell: UITableViewCell, forRowAt indexPath: IndexPath)`
 - название для header'a и footer'a
`func tableView(_ tableView: UITableView, titleForHeaderInSection section: Int) -> String?`
 - высота ячейки (далее более подробно)
`func tableView(_ tableView: UITableView, heightForRowAt indexPath: IndexPath) -> CGFloat`
 - `estimatedHeightForRowAt`
 - высота для header'a и footer'a

UITableView 9/10

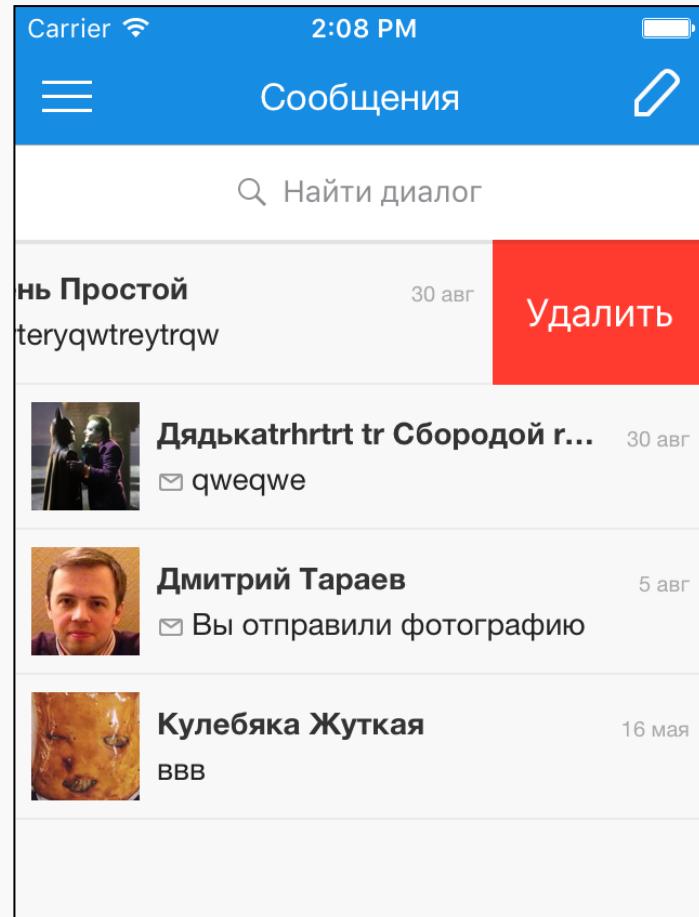


https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/UIKitUICatalog/UITableView.html#/apple_ref/doc/uid/TP40012857-UITableView-SW1

UITableView 10/10



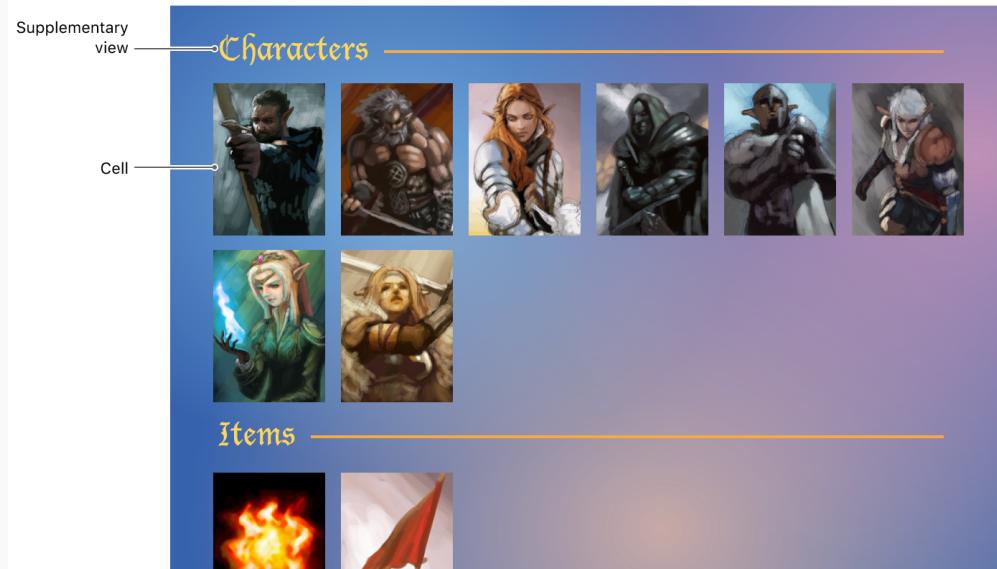
- Режим редактирования



UICollectionView 1/2



- UICollectionView
 - scroll direction
 - vertical / horizontal
 - layout
 - flow / custom
 - разбита на секции?
 - да / нет
(header, footer)



<https://developer.apple.com/documentation/uikit/uicollectionview>

UICollectionView 2/2



- Отличия от UITableView
 - IndexPath
 - section - item вместо section - row
 - Отдельный класс UICollectionViewLayout для пользовательского layout
 - UICollectionViewCell вместо UITableViewCell
 - UICollectionViewReusableView - для header и footer



1.UITableViewController

2.UIViewController с table view вручную

Заключение



- **UITableView & UICollectionView**
 - data source & delegate
- **Следующая лекция**
 - UIView
- **Отзыв**
- **Вопросы?**
 - по лекции
 - по домашним заданиям

Ссылки



- Подробнее про кастомный layout у collection view:
<https://www.objc.io/issues/3-views/collection-view-layouts/>