

iOS Developer. Basic

(Обновленный)

Длительность курса: 134 академических часа

1 Основы программирования

- | | | |
|---|--|---|
| 1 | Настройка XCode и основы мобильной разработки | изучаем интерфейс XCode.
Основы создания приложений |
| 2 | Данные и хранение | изучим что такое данные и как они хранятся на устройстве |
| 3 | Принципы ООП | основные принципы ООП |
| 4 | Простые типы данных | числа и строки
Рассмотрим простые типы данных и научимся с ними выполнять простые операции |
| 5 | Коллекции | массивы, словари, наборы |
| 6 | Функции | знакомство с функциями |
| 7 | Объекты ООП | рассмотрим понятия объекта |

1	Playground, Типы данных, кортежи, опционалы	разбираем основные типы данных
2	Коллекции (массивы, словари, множества)	знакомимся с коллекциями Swift
3	Циклы, ветвление	изучаем основные методы управления циклом выполнения программы, условия остановки и смены логики дальнейшего выполнения
4	Функции, замыкания	учимся работать с функциями. Знакомимся с функциями высшего порядка (замыканиями) Домашние задания 1 Реализовать все экраны приложения Цель: Реализовать все экраны приложения. В табличных представлениях использовать динамическую подгрузку. Все шрифты, цвета прописать в расширениях. Все ячейки посторить на XIB. Все должно быть покрыто UITest.
5	TDD, ООП, POP	изучаем принципы разработки через тестирование, основы ООП в SWIFT знакомимся с протокол-ориентированным программированием
6	Классы	изучаем объект класс
7	Структуры	изучаем объект структура
8	Перечисления	изучаем объект перечисление Домашние задания 1 Написать основные отличия между объектами в ООП Цель: Написать основные отличия между объектами в ООП. Придумать пример полиморфизма, и реализовать его. Создать очередь выполнения используя замыкания.

- 1 XCode, Storyboard, объекты UI, создание программно объектов, XIB** знакомимся со средой разработки, средой графического построения интерфейсов, основам визуальных блоков в программе и через графический интерфейс

- 2 TableView, collectionView** знакомимся с табличными представлениями

- 3 Переходы** изучаем переходы в приложении
Изучим segue, его виды, настройки

- 4 Навигация в приложении** разберем основы навигации
Домашние задания
 - 1** Реализовать все экраны приложения
Цель: Реализовать все экраны приложения.
В табличных представлениях использовать динамическую подгрузку.
Все шрифты, цвета прописать в расширениях.
Все ячейки построить на XIB.

- 1 **Разбираем домашнее задание** ответы на вопросы по пройденному материалу и ДЗ

- 2 **FileManager, UserDefaults** рассмотрим две формы хранения информации в файловой системе Напишем хранение логина и пароля в файловой системе

- 3 **CoreData** изучим графовую платформу хранения

- 4 **Realm** ознакомимся с библиотекой REALM

Домашние задания

- 1 Реализовать хранение данных приложения

Цель: Реализовать хранение данных приложения следующим образом:

Личные данные хранить в файловой системе.

Данных для коллекций в CoreData, остальное в REALM.

- 1 **URL, URLRequest, URLSession, URLSessionDelegate** рассмотрим основные классы работы с сетью

- 2 **Создание сетевого слоя, ResultType** создадим свой сетевой слой с безопасным типом данных

- 3 **Codable, DynamicJSON** научиться управлять выполнением задач с объектах OperationQueue;
создать очередь загрузки после авторизации.

Домашние задания

- 1 Создать сетевой слой, используя API

Цель: Создать сетевой слой, используя API, интегрировать его со слоем хранения.

- 1 Разбираем домашнее задание ч.2** ответы на вопросы по пройденному материалу и ДЗ

- 2 Thread, RanLoop** разберем и научимся работать с потоками

- 3 GSD** изучим работу с очередями на более высоком уровне используя GCD

- 4 Operation, OperationQueue** научимся управлять выполнением задач с объектах OperationQueue

Домашние задания
 - 1** Создать очередь загрузки в загрузочном контроллере

Цель: Создать очередь загрузки в загрузочном контроллере, добавить плавную загрузку изображений.

1 Порождающие паттерны

познакомимся с порождающими паттернами

2 Структурные паттерны

познакомимся со структурными паттернами

3 Поведенческие паттерны

познакомимся с поведенческими паттернами

Домашние задания

1 Создать фабрику моделей

Цель: На базе сетевого слоя и базы данных, создать фабрику моделей.

- 1 Разбираем домашнее задание ч.3** ответы на вопросы по пройденному материалу и ДЗ

- 2 SOLID, SOA, MVC, MVVM** изучим основы создания архитектуры приложения

- 3 Протоколы, делегаты** рассмотрим протоколы и шаблон делегат в создании архитектуры

- 4 CleanSwift, VIPER** познакомимся с двумя архитектурами, используемыми в крупных проектах

Домашние задания
 - 1** Контроллер авторизации переписать на CleanSwift

Цель: Контроллер авторизации переписать на CleanSwift, остальное приложение на MVVM.

1 **Xcode Instruments** изучим готовую среду тестирования приложения

2 **Fabric, Crashlitics** изучим облачные средства аналитики и сбора исключений

Домашние задания

1 Интегрировать в приложение Fabric, Crashlitics

Цель: Интегрировать в приложение Fabric, Crashlitics.

1 Публикация в AppStore

познакомиться с iTunesConnect, сертификатами, testflight; отправить приложение на публикацию.

Домашние задания

1 Создать отдельный проект

Цель: Создать отдельный проект.
В нем реализовать авторизацию на SwiftUI, с учетом архитектуры основного приложения.
Покрыть тестами

ВАЖНО:

Сдать ссылку на репозиторий курсового проекта.
В репозитории обязательно должен быть заполнен файл README.md с описанием проекта.

2 Защита проектных работ

защитить проект и получить рекомендации экспертов.