

Лекция №2

Введение в Swift

Гибадулин Олег



образование

Организационная часть

- Отметиться - важно
- О чем пойдет речь в сегодняшнем занятии
- Оставить отзыв (после занятия)



Структура лекции

- Hello world!
- Константы и переменные, типизация
- Операторы
- Опционалы
- Коллекции
- Управление потоком
- Функции
- Замыкания
- Перечисления

Историческая справка

Swift – был представлен компанией Apple в 2014 году в первую очередь для разработчиков iOS/macOS.

Objective-C

```
const int count = 10;
double price = 23.55;

NSString *firstMessage = @"Swift is awesome. ";
NSString *secondMessage = @"What do you think?";
NSString *message = [NSString stringWithFormat:@"%s%s", firstMessage, secondMessage];

NSLog(@"%s", message);
```

Swift

```
let count = 10
var price = 23.55

let firstMessage = "Swift is awesome. "
let secondMessage = "What do you think?"
var message = firstMessage + secondMessage

print(message)
```

Hello world!

```
1 print("Hello, world!")  
2 // Prints "Hello, world!"
```

← эта строчка кода - полноценная программа

```
4  
5 var text = "Hello, World"  
6 print(text)  
7
```

← тоже самое, только с переменной

Константы и переменные

Константы объявляются с ключевым словом `let`

Переменные со словом `var`

Типизация

Swift предоставляет свои собственные версии фундаментальных типов C и Objective-C, включая **Int** для целых чисел, **Double** и **Float** для значений с плавающей точкой, **Bool** для булевых значений, **String** для текста.

Swift также предоставляет мощные версии трех основных типов коллекций, **Array**, **Set** и **Dictionary**

Вывод типов

Можно добавить обозначение типа

```
var welcomeMessage: String
```

```
welcomeMessage = "Hello"
```


Кортежи

```
let http404Error = (404, "Not Found")
```

```
print(http404Error.0) // Выведет 404
```

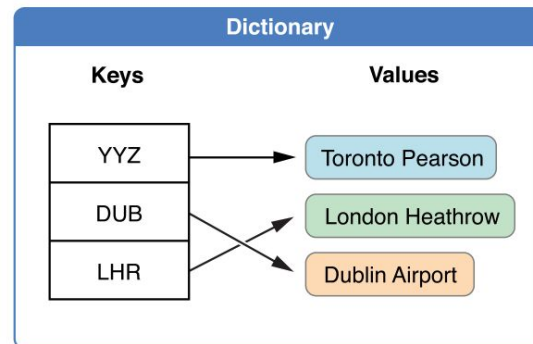
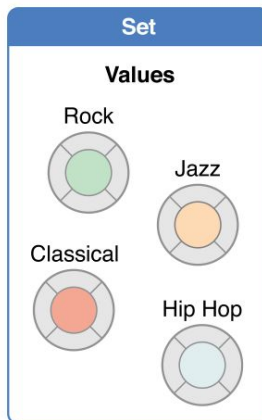
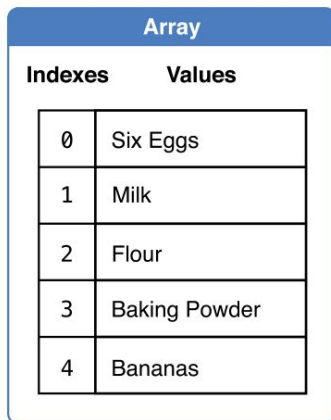
Операторы

Оператор — это специальный символ или выражение для проверки, изменения или сложения величин.

Опционалы

Опциональные типы используются в тех случаях, когда значение может отсутствовать. Опциональный тип подразумевает, что возможны два варианта: или значение есть, и его можно извлечь из опционала, либо его вообще нет.

Коллекции



Управление потоком

В Swift есть все знакомые нам операторы управления потоком из C-подобных языков. К ним относятся: циклы `for-in` и `while` для многократного выполнения задач, инструкции `if`, `guard` и `switch` для выполнения различных ветвлений кода в зависимости от определенных условий, а также такие инструкции, как `break` и `continue` для перемещения потока выполнения в другую точку вашего кода.

Функции

```
func greet(person: String) -> String {...}
```

Можно вернуть несколько значений с помощью кортежа (a,b)

Есть сквозные параметры

Вложенные функции

Замыкания

Замыкания - это самодостаточные блоки с определенным функционалом, которые могут быть переданы и использованы в вашем коде. Замыкания в Swift похожи на блоки в C и Objective-C, и лямбды в других языках программирования.

```
{ (параметры) -> тип результата in  
выражения  
}
```

Перечисления

Перечисления определяют общий тип для группы ассоциативных значений и позволяют работать с этими значениями в типобезопасном режиме в вашем коде.

Литература

<https://docs.swift.org/swift-book/>

<https://swiftbook.ru/content/languageguide/>

<https://itunes.apple.com/ru/course/developing-ios-11-apps-with-swift/id1309275316>

Вопросы

Спасибо за внимание!



образование