**Обзор инструментария**

**План занятия:**

1. Что такое язык Swift?
2. Основные типы переменных в языке
3. Создание переменных в языке Swift
4. Опциональные типы

Swift объектно-ориентированный язык программирования, который используется для нативной разработки под операционную систему IOS. Язык был представлен компанией Apple в 2014 году, и с тех пор активно развивается.

Swift, как и другие языки, оперирует такими понятиями, как переменные, функции, циклы, классы, структуры и тд. И в этом уроке мы посмотрим на синтаксис языка, разберем основные типы данных и их особенность, также поговорим про функции, посмотрим как их создавать и получать ответ от них.

Если говорить про основу Swift, то как и в любом другом языке, в основе лежит переменная. Swift является строго типизированным языком, это значит, что мы явно должны указать тип переменной, исключение, если мы создаем переменную, и сразу присваиваем ей значение, и то, с числовыми значениями лучше указывать тип, к примеру Float или Double и если мы попытаемся сложить переменные разных типов, это вызовет сразу ошибку.

В Swift как и во многих других языках существуют основные типы переменных:

* String - Текстовая строка
* Character – один символ
* Int - Целочисленное значение (8,16,32 бита)
* Uint – Числа без знака (8,16,32 бита)
* Double - 64-битное число с плавающей запятой, используется для более точных переменных
* Float - 32-битное число с плавающей запятой
* Bool - Логическое значение (true или false)
* Any - неопределенный тип, когда мы не знаем, что придет в нашу переменную, используется редко но иногда без него не обойтись.

Существуют и другие типы, их мы обязательно разберем в следующих лекциях и коснемся на практике, но в основном при разработке используются эти типы. Также можно создавать свои типы, это мы тоже обязательно будем делать в дальнейших уроках и это много проще, чем звучит.

Отдельно стоит поговорить про опциональные типы, по сути это те же типы данных, но давайте лучше разберем на примере:

Допустим есть переменная ***a*** к примеру тип **Int**, а главное то, что по ходу выполнения программы она нам может как пригодится, так и не пригодится, поэтому логично сделать ее необязательной или *опциональной.* По сути любой тип, который мы рассмотрели может быть опциональным. А теперь посмотрим, как все это выглядит в коде.

Сначала создадим простые переменные.



Синтаксис создания переменной очень простой: сначала мы указываем var или let.

* **var** - для создания простой перезаписываемой переменной. К примеру значения переменных ***a*** и ***b*** мы можем менять по ходу выполнения кода
* **let** - константа, значения этих переменных мы не можем менять. В переменную ***x*** мы не сможем больше ничего записать, а только использовать то значение, которое мы в нее записали при ее создании. Важный совет, везде где возможно используйте let, так вы меньше будете выделять памяти под переменную и не будете понапрасну расходовать ресурсы.

Обратите внимание на переменную ***text***, тут мы не определили тип, но Swift умный язык, и если мы явно указываем значение переменной, то можем отпустить момент типизации переменной. Мы также могли у переменной **a** не писать тип **Int***.*

-----

А теперь давайте посмотрим на опциональный тип.



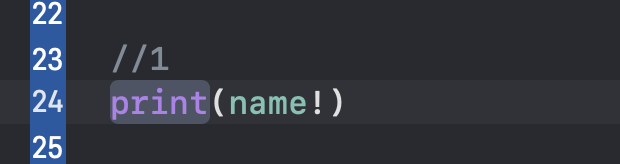
При создании опционального типа, мы просто ставим знак **?** после типа переменной. **Очень важно:**

* Опциональным типам мы можем как присваивать так и не присваивать значение, если не присвоим, его значение будет **nil** и на практике мы не раз с этим столкнемся и более детально разберем.
* Константу мы можем делать опциональной при наличии у него значения, в противном случае он всегда будет **nil** и в него мы уже не сможем ничего записать

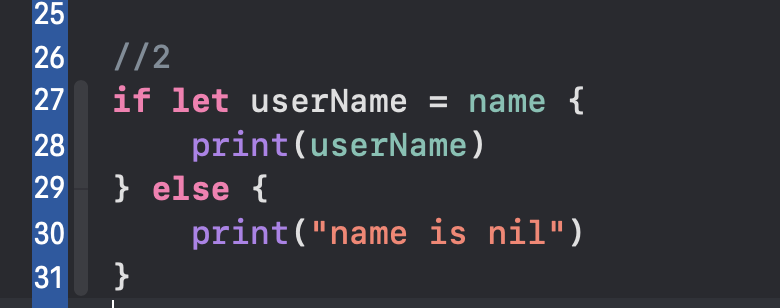
С опциональными переменными нужно работать осторожно, если по ходу выполнения программы у нас в коде встретиться **nil**, а мы это никак не проверяем, то программа вернет фатальную ошибку и вылетает. Строгая типизация и проверка опциональных типов - это дань Apple безопасности системы.

Развертка (uunwrapping) опциональных типов:

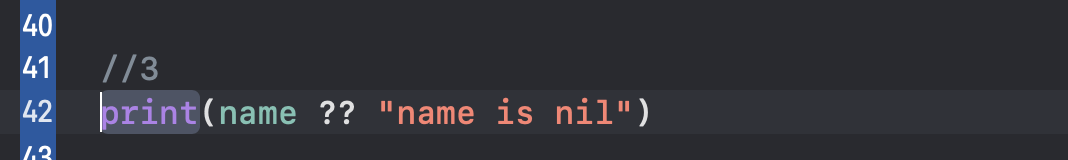
* + - 1. Force unwrapping – добавление знака (!)



* + - 1. Optional binding через условие if/else



* + - 1. nil coalescing operator – присваивание значения если nil



ДЗ:

Создать переменные для анкет (имя, год рождения и тд). Вообще поиграться с созданием переменных, вывести их, задать разные значения

Переменные наличие животных

Если есть животные, вывести имя, если нет, то советующее значение

https://docs.swift.org/swift-book/LanguageGuide/TheBasics.html