

A thick black L-shaped frame surrounds the text. It starts at the top left, goes right, then down, then right again at the bottom.

ФУНКЦИОНАЛЬНОЕ И ЛОГИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

Практика-6 Kotlin

Лукашин Антон

Высшая школа программной инженерии
Институт Компьютерных Наук и Технологий



История

- Начало работ – 2010 год
 - 2011 - анонс
 - 2012 – *milestone-1*
 - 2012 – *исходный код открыт*
- Первый релиз – 15 февраля 2016
- Май 2017 – стандарт для Android





Философия

- Промышленный язык разработки
- Базируется на JVM и поддерживает библиотеки Java
- Легко переходить с Java на Kotlin
- Мультипарадигменный
- Лучше чем Java – попытка втянуть лучшее отовсюду
- Язык должен занять позицию между Java и Scala
- Более строгий и безопасный



Особенности

- Классы
 - *Public by default*
 - *Final by default*
 - *Primary constructor*
 - *Extension methods*
- Типы данных
 - *Tuples*
 - *Spread operator* – массив в параметры функции (*vararg*)
- Val и var
- Nullable
 - *?* – для указания что переменная\параметр могут быть *null*
 - *Elvis operator ?:* (что-то если не *null*, иначе ...)
 - *foo()?.bar()?.baz()*
- Kotlin Script



Functional programming

- Function as first class citizen

```
val test = fun(a: Int): String { return "Hello" }
```

- High-Order function

```
fun doSmth(value: Int, op: (Int) -> (Int)): Int {return op(value)}
```

- Immutable data – ~~side effects~~

- Currying

```
val doSmth2 = fun(op: (Int) -> (Int)): Int {return doSmth(5, op)}
```

- Lambdas

```
val joinedToString = items.fold("Elements:", { acc, i -> acc + " " + i })
```

- Nested functions



Пример

■ Quicksort

```
fun <T : Comparable<T>> List<T>.quickSort(): List<T> =  
    if(size < 2) this  
    else {  
        val pivot = first()  
        val (smaller, greater) = drop(1).partition { it <= pivot}  
        smaller.quickSort() + pivot + greater.quickSort()  
    }
```

■ a



Задачи

- Reverse строки (куда ж без него)
- Расстояние Хэмминга для двух ДНК
- Факториал

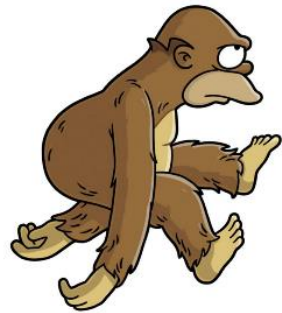


За кадром

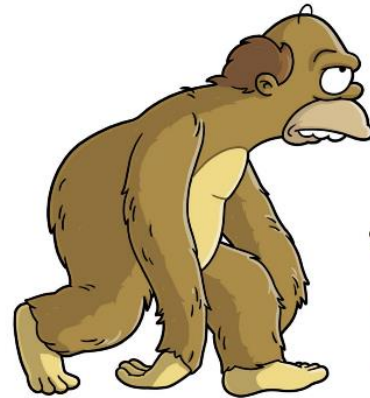
- coroutines
- Parallel programming
- Kotlin Script
- And 99% of Kotlin language



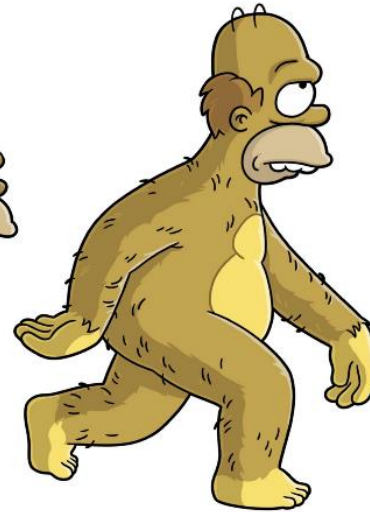
MACHINE



ASSEMBLY



PROCEDURAL



OBJECT ORIENTED



FUNCTIONAL

MATT GROENING