OCTO Technology

BOF: Kotlin Introduction

Is this as hype as everybody says ?

OCTOBER 2017 - Nicolas Guignard

KOTLIN

01. What is Kotlin ?

02. Getting started

03. Functions

04. Object oriented

05. Unit testing

06. Conclusion

01 > What is Kotlin ?

KOTLIN - What is it ?

- JetBrain
- JVM Language
- Can be compiled into Javascript
- Functional
- Object Oriented

02 > Getting started

KOTLIN - Getting started

- More concise than Java :
  - No getter / setter or no new keyword
  - Constructor in class declaration

- Functions / constructors can have default parameters

- Type inference : var / val / let

- String templating

- “If-else” is an expression
  - return value

KOTLIN - Getting started

- Try-catch-finally / try-with-resource same than Java

- For loops on ranges

- Support immutability by default :
  - val keyword
  - classes are immutable by default

- Interoperability with Java

- Find interface / abstract classes :
  - can derive or implement class

KOTLIN - Getting started

- “object” keyword is to create singleton / factory
  - static methods

- Collections : Java ++
  - List / Set / Map && Mutable / Immutable
  - Sequence

- Question mark : “?” to specify if your variable can be null
  - protection against Null values

KOTLIN - Getting started

Let’s have a look at some code now !

03 > Functions

KOTLIN - Functions

- **fun** keyword
- Don’t need to be a part of a class - e.g main.kt
- Can have default parameters
- Can extend existing types
- Function expression
  - 1 line of code

KOTLIN - Functions

- Can be **tail recursive**
  - **tailrec** keyword
  - Kotlin will optimise the recursion
  - Byte code representation : While loop

- **Extension** functions
  - Add functions to classes not owned by us

- **Infix** functions
  - **infix** keyword
  - Can be member or extension function
  - transform function in operators

KOTLIN - Functions

- **High Order Functions : Hall Of Fame**
  - Functions are ***1st class citizen***
  - can pass to and return from functions
  - can store functions in collections
  - Closures: Kt lambdas can mutate values **!= JAVA**
  - with and apply functions

- Map and Flatmap

KOTLIN - Functions

Let’s have a look at some code now !

04> Object Oriented ?

KOTLIN - Object Oriented

- **class** keyword

- Classes are public & final by default - functions too -
  - To derive a class : **open** (keyword) it - same to override a function from a class you derive from
  - Can have several constructors

- Classes can be : **class** / **abstract** / **sealed** / **data** / **interface**

- Data class :
  - To manipulate data
  - For ORM use
  - To store in collections
  - No need to implement equals / hashcode / toString / copy
  - Immutable

KOTLIN - Object Oriented

- Sealed classes :
  - Restrict class hierarchy
  - Enum on steroids

- Interface
  - Multiple implementations
  - Default functions
  - **override** keyword for functions

KOTLIN - Object Oriented

Let’s have a look at some code now !

05 > Unit Testing

KOTLIN - Unit testing

YES YOU CAN !

- Spek framework
  - similar to Mocha / Jasmine
- JUnit 4

06 > Conclusion

KOTLIN - Conclusion

- Spring boot
- Spark framework
- Spring 5
  - leverage Kotlin
- Community