Kotlin

Introduction

... by JetBrains

What is Kotlin?

What is Kotlin?

Kotlin is...

Cross-platform
Statically typed
General-purpose

Kotlin compiles to...

Java ByteCode
JavaScript
Native Code

What is Kotlin?

Of course, runs on every **Java Virtual Machine (JVM)**

Libraries written in Kotlin are compatible to be used and run in a Java project...

... and vice versa

What is Kotlin?

Inspired by Java, but...

cleaner

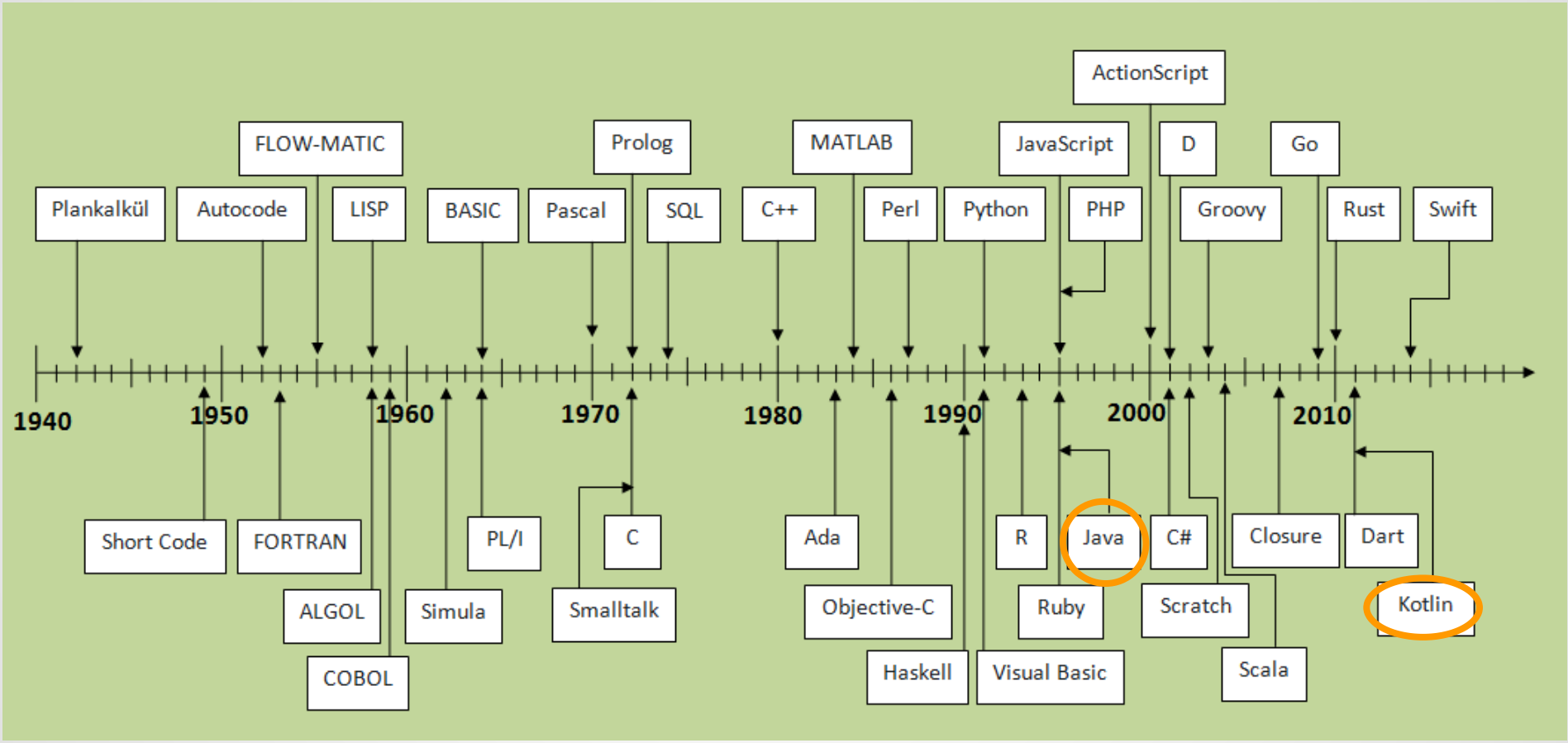
simpler

faster to compile

a mix of OOP and functional programming

Why Kotlin?

Timeline



https://javaconceptoftheday.com/history-of-programming-languages/

Java is pretty old... arguably...

But why?

Since 2019 it is the favored programming language for Android applications at Google

Underwent an incredible growth since then

Kotlin

compared to Java

https://www.imaginarycloud.com/blog/kotlin-vs-java/

Null Safety

Java

Every object defaults to null

Kotlin

By default there is no null value until you specify it like this

```
val number: Int? = null
```

Extension Functions

Java

Not available

(available by using Project Lombok)

(maybe by inheritance, but you don't own every object you'd like to extend)

Kotlin

A standard language construct by simply prefixing the function name with the class name it should be added to

Code

Java

Verbose (e.g. getter, setters)

Been here before some of the more modern language constructs were invented (e.g. async)

Kotlin

Very concise language (to the point)

Fewer lines of code

Better to code / read / maintain

Coroutines

Java

Background threads (ExecutorService, etc.)

Kotlin

Own threadpool

Part of the language

```
1  fun main() = runBlocking { // this: CoroutineScope
2      launch { // launch a new coroutine and continue
3          delay(1000L) // non-blocking delay for 1 second
4          println("World!") // print after delay
5      }
6      println("Hello") // main coroutine continues
7  }
8
9  Output:
10 Hello
11 World!
```

Data Classes

Java

Manually (verbose)

Solved by using Project Lombok

Kotlin

A language construct

Automatically implements

getters, setters,

hashCode(),

equals(),

toString()...

Smart Casts

Java

Developer has to check the types

Kotlin

Casting checks are handled by the smart casts feature

Redundant checks are removed

No Checked Exceptions

Java

Checked exceptions are available

(IMHO this is a good thing)

Kotlin

No checked exceptions

(IMHO this is a bad thing, because, well ... exception handling)

Higher-Order Funcs & Lambdas

```
1  max(strings, { a, b -> a.length < b.length })
2
3  /**
4  The function max is a higher-order function,
5  as it takes a function value as its second argument.
6  This second argument is an expression that is itself a function,
7  called a function literal,
8  which is equivalent to the following named function:
9  **/
10
11 fun compare(a: String, b: String): Boolean = a.length < b.length
```

available in Java as well, to some extent

Primitive Types

Java

Variables of primitive types are not an object

Kotlin

All variables are objects

Public Fields

Java

Available, but should not be used

Kotlin

Not available at all

Wildcard Types

(Generics)

Java

? can be used to specify a type of <any>

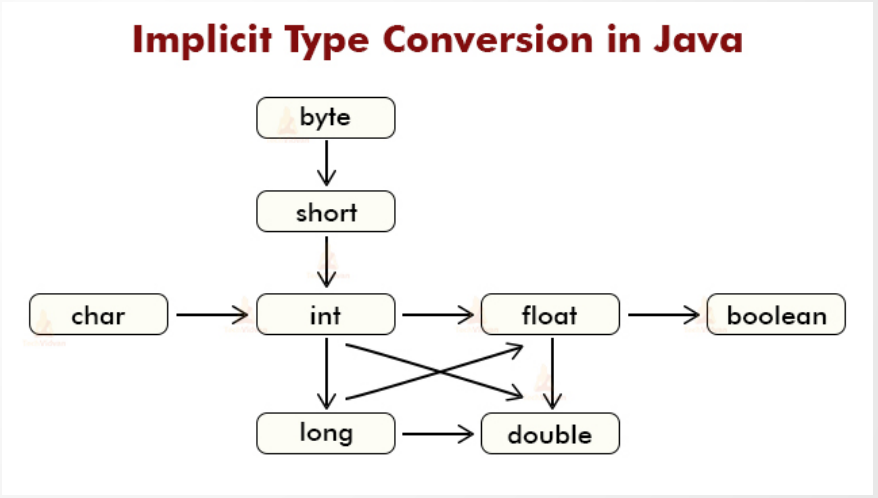
Kotlin

Not available. declaration-site variance and type projections as alternative

Explicit Conversions

Java

Supports implicit conversions (called 'widening')



Kotlin

No implicit conversions. You have to convert explicitly.

Continue here...

https://kotlinlang.org

(Try Kotlin, then Why Kotlin?)

then...

https://kotlinlang.org/docs/home.html