

Kotlin - a Java++ ?

...a talk by Anders Skarby



Very time limited talk

Not going to be able to teach you  **Kotlin** !

(at least not in the timeframe of this talk)

...however

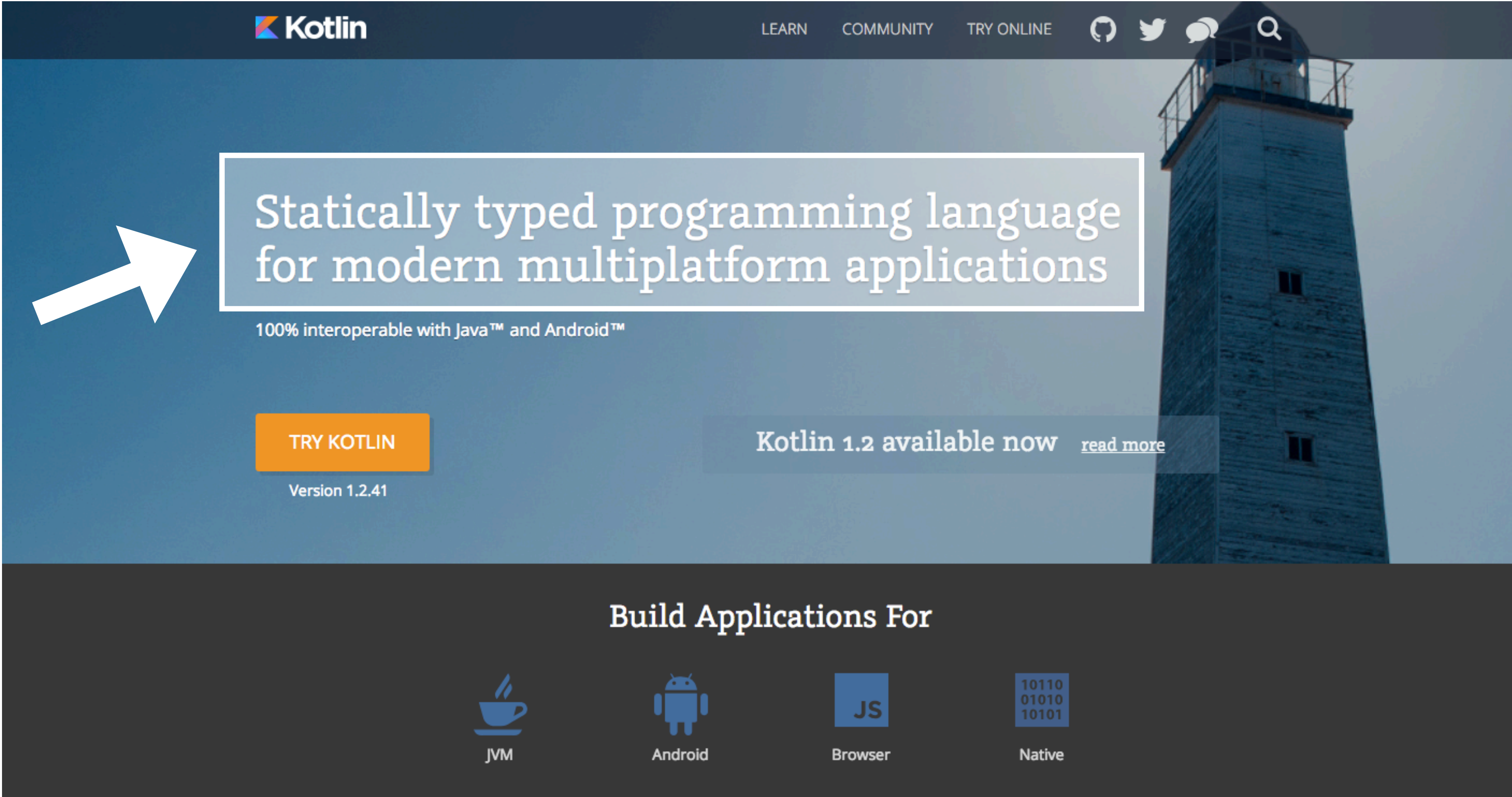
Will be able to look at very cool features


(about 10 of them, or so)





Outline of talk

1. **What** is  **Kotlin** ?
2. **How** and **why** do I use  **Kotlin** ?
3. Nine (if we get around to them all) **cool things** about  **Kotlin** !

What is Kotlin?



 **Kotlin**

LEARN COMMUNITY TRY ONLINE    

Statically typed programming language
for modern multiplatform applications


100% interoperable with Java™ and Android™


TRY KOTLIN


Version 1.2.41

Kotlin 1.2 available now [read more](#)


Build Applications For

 JVM

 Android

 JS



Browser

 Native

What is Kotlin?

Statically typed programming language
for modern multiplatform applications

Statically typed programming language, ie.:

- IDE can help out (a lot)
- Multi-paradigm
- **Not** bound to execution on the JVM
 - Can invoke  -code from  **Kotlin**, and vice versa

How and why do I use Kotlin ?

How?

- Not in a professional context (at least not to the degree I want to)
- Hobby projects, for backend-focused code.

Why?


- Runs on JVM, performance is same as Java
- Nicer / Easier to **read** and write
- Great integration with



Tooling

- Developed by , hence support in:




- Support for Eclipse (also made by ):



Nine **cool things** about **Kotlin** !

All focused on language-centric constructs!

1. Small differences, nicer syntax
2. Statements and expressions
3. Everything is an object (where did my primitives go?)
4. Data classes
5. Classes, Inheritance and Delegation
6. Objects
7. Extension methods
8. Java inter-op
9. Automatic conversion of code

A film strip background with the text "THE END" in a bold, black, serif font. The film strip has a yellowish, aged appearance with visible sprocket holes on the left and right edges. The text is centered horizontally and vertically.

THE END

Grab the presentation and code from:

<https://github.com/askarby/short-kotlin-talk-2018>