



# Kotlin

## CROSS PLATFORM



**Andreas Wenger - Stocard GmbH**

**@WengerAndi**  
**[github.com/awenger/multiplatform-kalkulator](https://github.com/awenger/multiplatform-kalkulator)**

# /whoami

- Android developer at Stocard
- Some experience with nodejs
- 1+ Year of Kotlin experience
- @WengerAndi

/whoareyou

# /why

- Share code
- Don't repeat yourself (DRY)
- Consistent code base

# /platform specific

- UI
- Platform specific apis/functionality

# /common

- Stdlib
  - Basic types
  - Collections + functional operators (filter/map)
  - Higher order functions (apply/with)
  - Exceptions
- Testing
- Common kotlin libraries
- Kotlin code

# /js|jvm|native

- Platform specific libraries
- Platform specific code

A diagram consisting of four colored squares arranged in a 2x2 grid. The top square is green and labeled 'JVM'. The bottom-left square is pink and labeled 'Native'. The bottom-middle square is olive green and labeled 'Common'. The bottom-right square is light blue and labeled 'JS'.

JVM

Native

Common

JS



# /init

- gradle
- settings.gradle
- build.gradle
- <lib>-common/build.gradle
- <lib>-native/build.gradle
- <lib>-jvm/build.gradle
- <lib>-js/build.gradle

```
apply plugin: 'kotlin-platform-common'  
dependencies {  
    compile "[..]:kotlin-stdlib-common:[...]"  
}
```

```
apply plugin: 'kotlin-platform-<platform>'  
dependencies {  
    compile "[..]:kotlin-stdlib-<platform>:[...]"  
    expectedBy project(':<lib>-common')  
}
```

# /sample

Input



Output

# /Kotlin/JS

- `dynamic` **type**
  - Disables type checking
  - Just like javascript
- `external` **modifier**
  - Definition of external (non-kotlin) interfaces

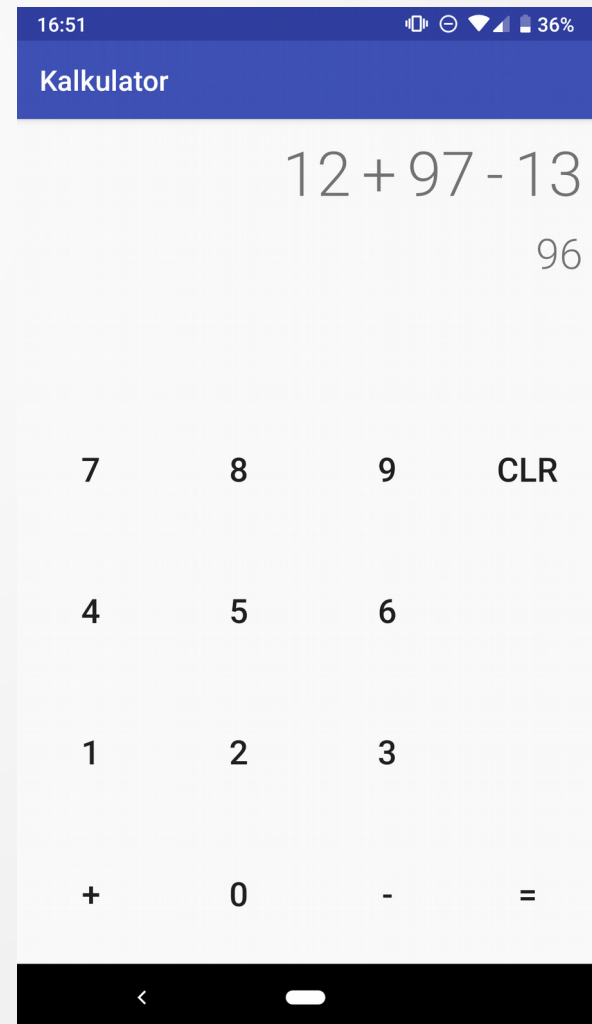
# /rest[ish] api

- restify
- GET /calculator to observe the state
- POST /calculator/inputs/ to add input

```
curl 'http://localhost:8082/calculator'
{"inputLine":"","resultPreview":""}
curl -XPOST 'http://localhost:8082/calculator/inputs/' \
-H 'content-type: application/json' -d '{"input":"1"}'
curl -XPOST 'http://localhost:8082/calculator/inputs/' \
-H 'content-type: application/json' -d '{"input":"+"}'
curl -XPOST 'http://localhost:8082/calculator/inputs/' \
-H 'content-type: application/json' -d '{"input":"1"}'
curl 'http://localhost:8082/calculator'
{"inputLine":"1+1","resultPreview":"2"}
```

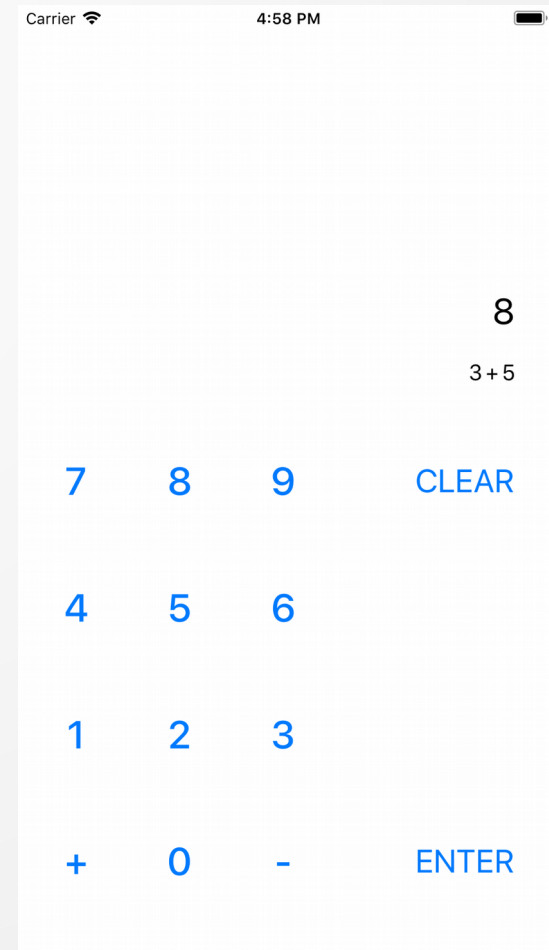
# /app -type android

- Single Activity
- No logic
- Embedded as lib
- maven



# /app -type iOS

- Single ViewController
- No Logic
- Embedded as Framework



# /expect-actual

- Missing functionality in commons

```
expect class HttpClient {  
    fun get(url: String): Response  
}
```

```
expect fun String.toUtf8EncodedByteArray() : ByteArray
```

- provides platform specific implementation

```
actual class HttpClient {  
    fun get(url: String): Response {  
        // delegate to OkHttp  
    }  
}
```

```
actual fun String.toUtf8EncodedByteArray() : ByteArray =  
    this.toByteArray(Charsets.UTF_8)
```

# /test

- Common module
- Platform specific modules
- Kotlin/JS via Jasmine, Mocha, Jest
  - Setup is a bit cumbersome
  - [blog.kotlin-academy.com/testing-common-modules-66b39d641617](http://blog.kotlin-academy.com/testing-common-modules-66b39d641617)
- Kotlin/JVM via Junit
- Kotlin/Native via ?



# /more

- `kotlinx.coroutines`
  - [github.com/Kotlin/kotlinx.coroutines](https://github.com/Kotlin/kotlinx.coroutines)
  - Native support? #246
- `kotlinx.serialization`
  - JSON, CBOR, Protobuf (de-)serialization
- `kotlin-frontend`
  - [github.com/Kotlin/kotlin-frontend-plugin](https://github.com/Kotlin/kotlin-frontend-plugin)
  - npm dependencies
- Setup
  - [github.com/JetBrains/kotlin-native/blob/master/MULTIPLATFORM.md](https://github.com/JetBrains/kotlin-native/blob/master/MULTIPLATFORM.md)

# /projects

- KotAcademy application
  - [github.com/MarcinMoskala/KotAcademyPortal](https://github.com/MarcinMoskala/KotAcademyPortal)
  - backend/web/android/wear/desktop
- KotlinConf App
  - [github.com/JetBrains/kotlinconf-app](https://github.com/JetBrains/kotlinconf-app)
  - backend/web/android/iOS