





# Danilo Paiva

Back End Developer

# Agenda

KOTLIN

DEMO

TDD

DEMO



2011



2012



2016

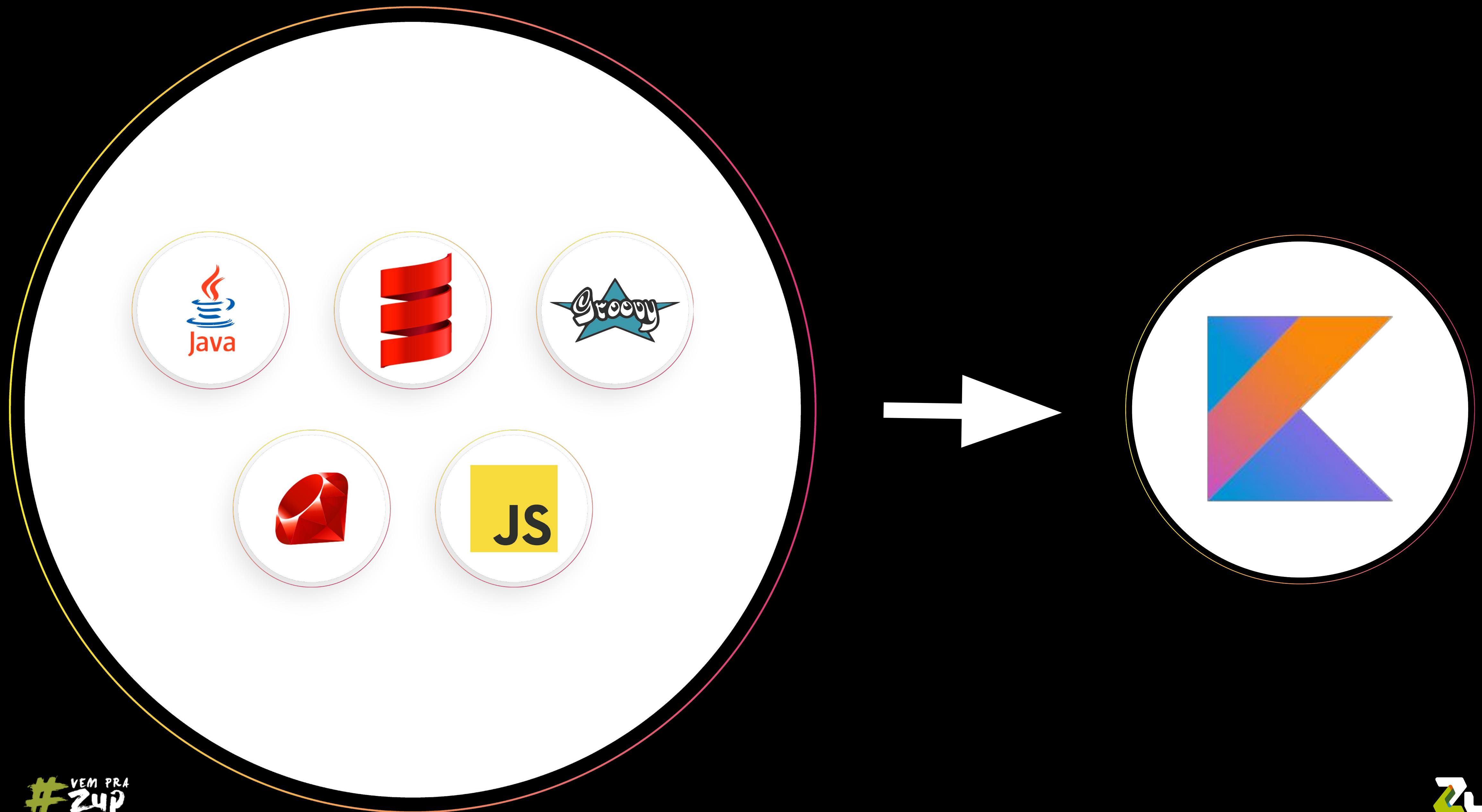


2017



2017

**WARRIOR**



# Curva de aprendizagem

- Verboso

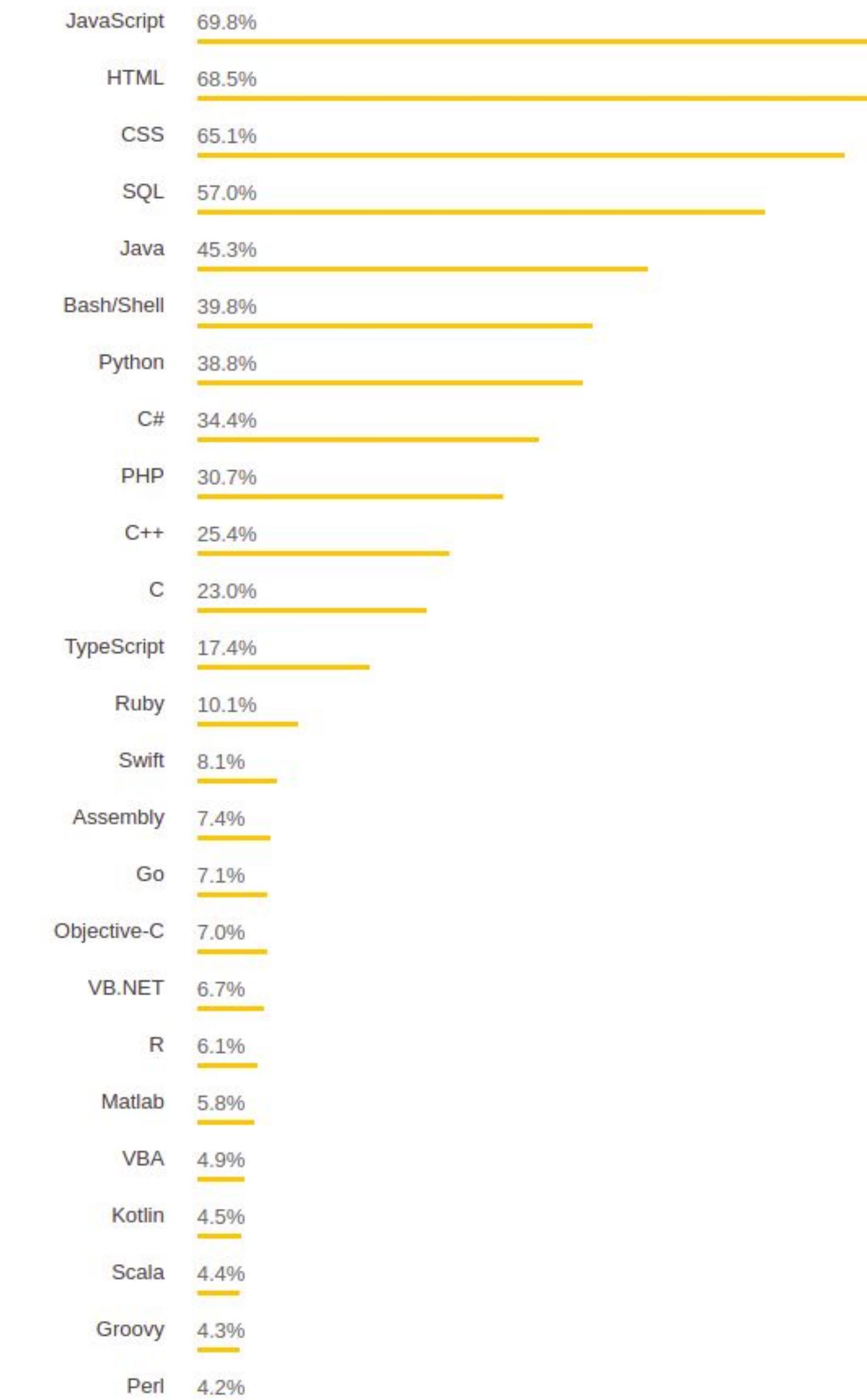
- Boilerplate

**Desenvolve + rápido**

# Null Safety

# Interoperabilidade

# Most Popular

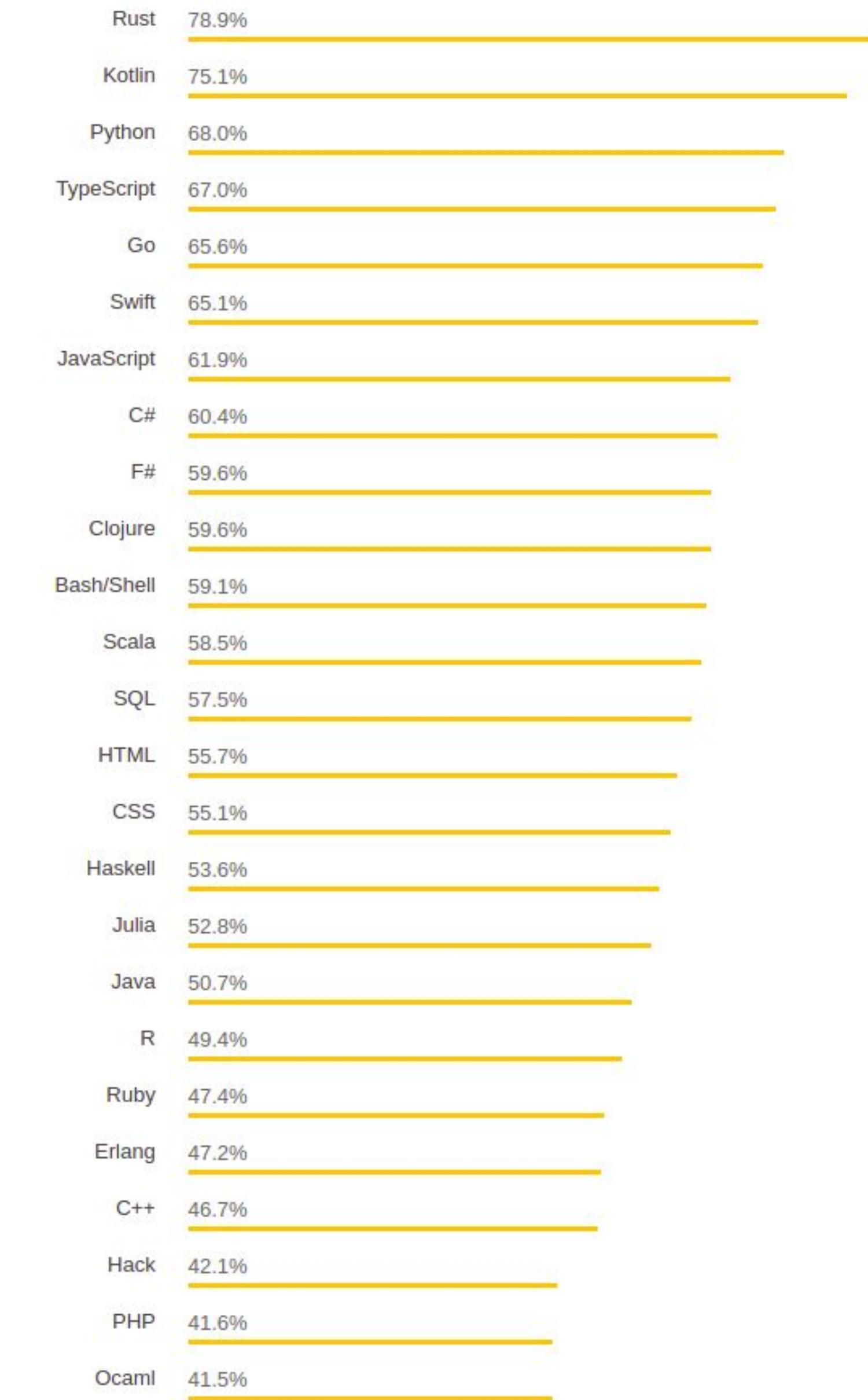




#**Zup**  
VEM PRA

**Zup**

# Loved



# Agenda

KOTLIN

DEMO

TDD

DEMO

# Agenda

KOTLIN

DEMO

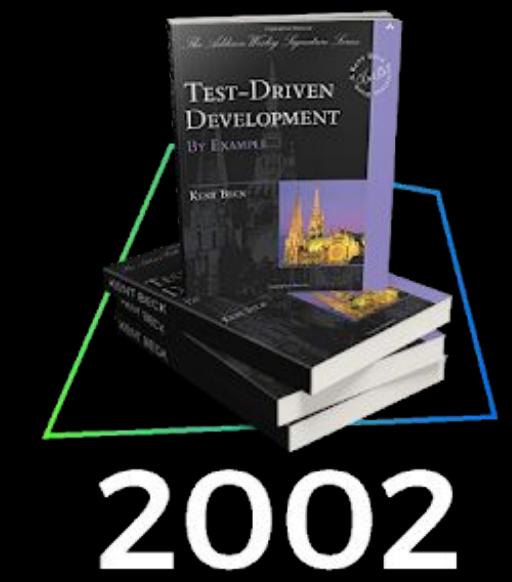
TDD

DEMO



SUnit  
1994

JUnit  
2000





TDD  
NÃO  
É DDT

CÓDIGO CÓDIGO CÓDIGO CÓDIGO TESTES

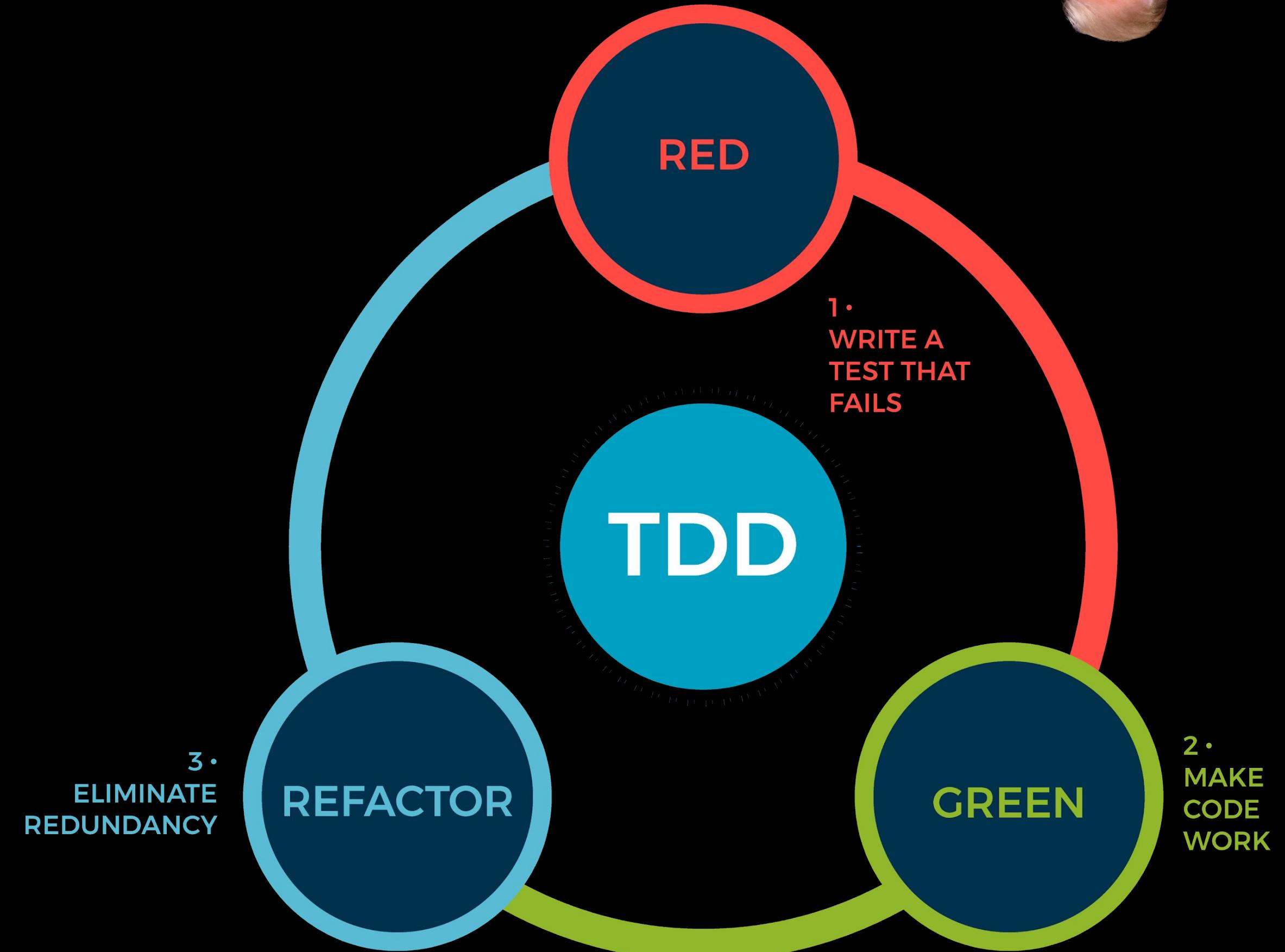
CÓDIGO CÓDIGO CÓDIGO CÓDIGO

DDT

TDD



# Ciclo



# TDD

# Benefícios

O código nasce testado :-)

# Feedbacks

# Design da classe :-)

- + Coesão
- Acoplamento

# Segurança

TDD

Dificuldades



“ Software baseado em testes  
que são escritos antes do  
nossa código de produção. ”



**Não tenho tempo para  
testar... blá blá blá**



**Testar é caro porque  
aumenta o tempo de  
desenvolvimento.**

# THEY DIDN'T USE

38

# TDD

**Faz o simples!**



A dificuldade está em pensar  
nos **cenários**

Quando o **DDT** já foi aplicado

:-)

PQ Não?



**4000**  
deploys em  
produção  
por dia

**QCon**  
SÃO PAULO



Fonte: <https://www.infoq.com/br/presentations/um-dia-na-vida-de-um-arquiteto-de-performance-do-netflix>



**TDD**

**Cuidados**

# Quando usar?

**Os testes agora são códigos**



#  
**VEM PRA**  
**ZUP**

**ZUP**

# JUnit

```
class ValidaMelhorTimeDoBrasilTest {  
    @Test  
    fun `deve falhar pois sao paulo nao eh igual ao curintia`() {  
        Assert.assertEquals("São Paulo", "Curintia")  
    }  
    @Test  
    fun `deve ser verdadeiro pois sao paulo eh o melhor time do brasil`() {  
        Assert.assertTrue(true)  
    }  
    @Test  
    fun `deve ser falso pois curintia nao eh o melhor time do brasil`() {  
        Assert.assertFalse(false)  
    }  
    @Test(expected = NullPointerException::class)  
    fun `deve lancar excecao pois palmeiras nao eh o melhor time do brasil`() {  
        throw NullPointerException("Não tem mundial!")  
    }  
}
```

Run: ValidaMelhorTimeDoBrasilTest

Tests failed: 1, passed: 3 of 4 tests – 79 ms

ValidaMelhorTimeDoBrasilTest (br.com.zup.realwave.swapper.domain) 79 ms

- deve falhar pois sao paulo nao eh igual ao curintia 74 ms
- deve lancar excecao pois palmeiras nao eh o melhor time do brasil 1 ms
- deve ser falso pois curintia nao eh o melhor time do brasil 4 ms
- deve ser verdadeiro pois sao paulo eh o melhor time do brasil 0 ms

org.junit.ComparisonFailure:  
Expected :São Paulo  
Actual :Curintia  
[Click to see difference](#)

<1 internal call>

```
at kotlin.test.junit.JUnitAss
at kotlin.test.AssertionsKt_
at kotlin.test.AssertionsKt.a
at kotlin.test.AssertionsKt_
at kotlin.test.AssertionsKt.a
at br.com.zup.realwave.swappe
```

# Mock

**CONTROLLER**



**DOMAIN**



**REPOSITORY**

```
class ValidaMelhorTimeDoBrasilTest {  
  
    private val validaMelhorTimeDoBrasil: ValidaMelhorTimeDoBrasil = mock()  
  
    @Test  
    fun `deve chamar a validacao uma vez`() {  
        whenever(validaMelhorTimeDoBrasil.valida(Time.SAO_PAULO)).thenReturn(true)  
        validaMelhorTimeDoBrasil.valida(Time.SAO_PAULO)  
        verify(validaMelhorTimeDoBrasil, times(1)).valida(Time.SAO_PAULO)  
    }  
  
    @Test(expected = Exception::class)  
    fun `deve estourar excecao pois vasco nunca sera o melhor time do brasil`() {  
        whenever(validaMelhorTimeDoBrasil.valida(Time.VASCO)).thenThrow(Exception::class.java)  
        validaMelhorTimeDoBrasil.valida(Time.VASCO)  
    }  
}
```

# **Spring Boot Starter Test**

**MockMvc**

**Rest Docs :-D**

# Setup

## Create

POST /v1/setup

Create tenant.

### Example request

```
POST /v1/setup HTTP/1.1
Content-Type: application/json; charset=UTF-8
Host: localhost:8080
Content-Length: 70

{
  "organizationSlug" : "zup",
  "applicationId" : "sample_segment"
}
```

### Request fields

| Path             | Type   | Optional | Description  |
|------------------|--------|----------|--|
| organizationSlug | String | false    | Identify your organization slug.<br><br>Size must be between 1 and 62 inclusive. |
| applicationId    | String | false    | Identify your application id.<br><br>Size must be between 1 and 62 inclusive.    |

## Example response

```
HTTP/1.1 201 Created
tracking: a3fb7e75edae4d75bcf001a7a60171a7
Content-Type: application/json; charset=UTF-8
Content-Length: 70

{
  "organizationSlug" : "zup",
  "applicationId" : "sample_segment"
}
```

## Response fields

| Path             | Type   | Optional | Description                |
|------------------|--------|----------|----------------------------|
| organizationSlug | String | true     | Organization slug created. |
| applicationId    | String | true     | Application Id created.    |

# Sonar

Quality Gate Passed

Bugs & Vulnerabilities

Leak Period: since previous version  
started a month ago

0 A

 Bugs

0 A

 Vulnerabilities

0 A

 New Bugs

0 A

 New Vulnerabilities

Code Smells

1d A

started a month ago

40

 Code Smells

35min A

New Debt

3

 New Code Smells

Coverage



85.6%

Coverage

96.4%

Coverage on  
88 New Lines to Cover

Duplications



0.0%

Duplications

0

Duplicated Blocks

0.0%

Duplications on  
732 New Lines

Quality Gate Passed

Bugs & Vulnerabilities

0 A

 Bugs

0 A

 Vulnerabilities

Leak Period: since previous version  
started 17 days ago

0 A

 New Bugs

0 A

 New Vulnerabilities

Code Smells

2d A

started 17 days ago

100

 Code Smells

1d A

New Debt

51

 New Code Smells

Coverage



72.8%

Coverage

78.9%

Coverage on  
299 New Lines to Cover

Duplications



0.0%

Duplications



0

Duplicated Blocks

0.0%

Duplications on  
2.4k New Lines

# Agenda

KOTLIN

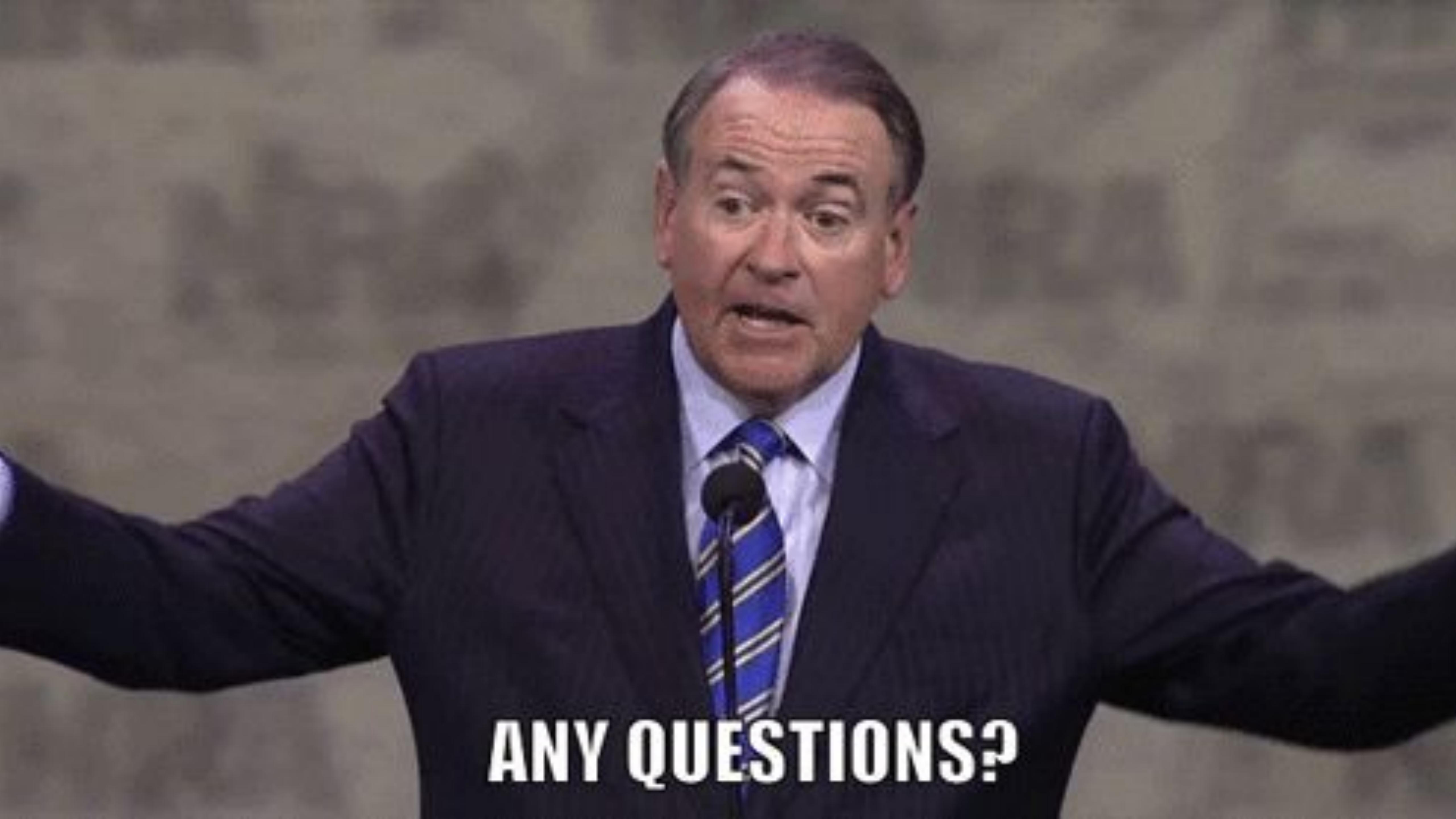
DEMO

TDD

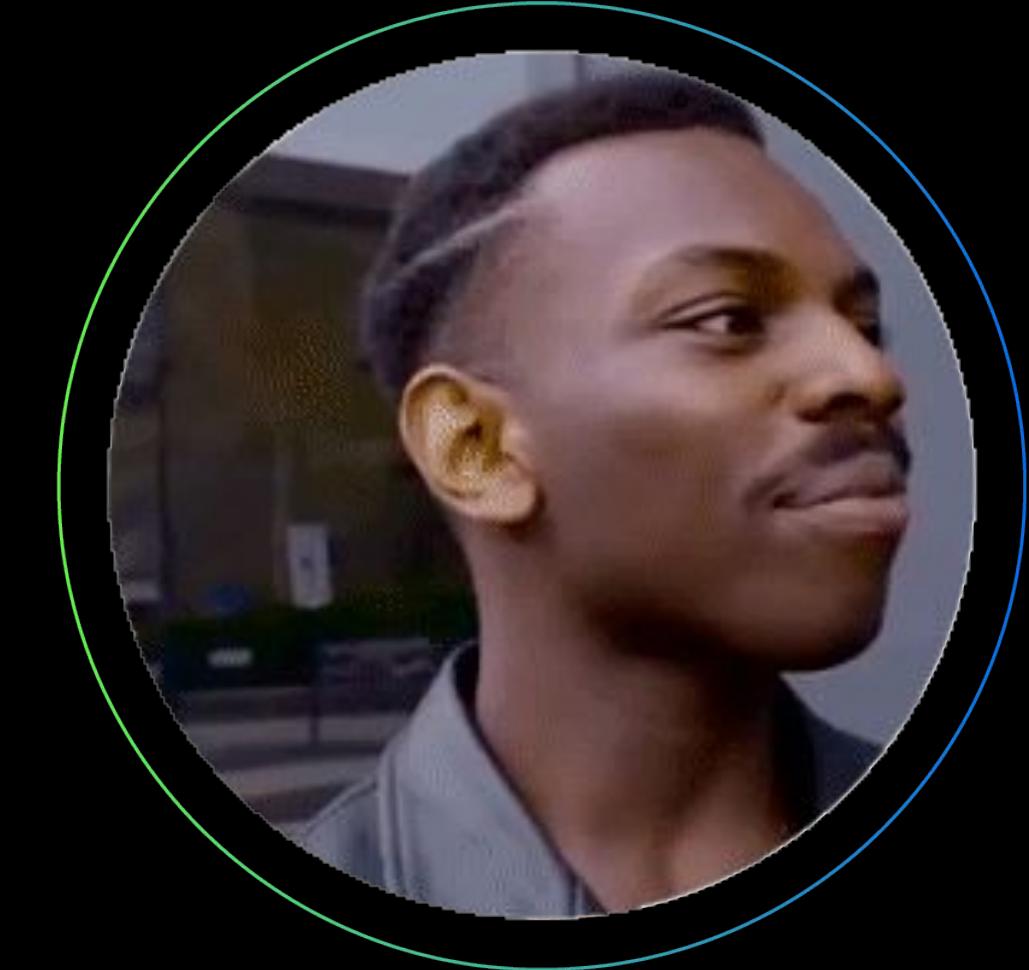
DEMO

**Blz, mas vamos fazer oq?**

**Partiu!**

A medium shot of a man from the chest up. He has short, dark hair and is wearing a dark suit jacket over a light blue dress shirt and a blue tie with thin white diagonal stripes. He is looking slightly to his left with a neutral to slightly questioning expression. A black microphone is positioned in front of him, pointing towards his mouth.

**ANY QUESTIONS?**



I'M NOT A GREAT PROGRAMMER;  
I'M JUST A GOOD PROGRAMMER WITH GREAT HABITS.

KENT BECK

# Obrigado!



[github.com/danilopaiva](https://github.com/danilopaiva)



[linkedin.com/in/danilopaivasilva](https://linkedin.com/in/danilopaivasilva)



[@danilopaiv4](https://www.instagram.com/danilopaiv4)



[danilo.silva@zup.com.br](mailto:danilo.silva@zup.com.br)