



Grupo:

Diogo Furtado

Phellipe Perin

Stephan Hebeda

# INTRODUÇÃO

- Scala é uma linguagem de programação de propósito geral projetado para expressar padrões de programação comuns de uma forma concisa.
- Foi desenvolvido como um “Java melhor”, rodando em cima da JVM ele mantém uma forte interoperabilidade com Java enquanto que ao mesmo tempo integra programação funcional e orientada a objeto.

# INTRODUÇÃO

- Como Scala roda na JVM ele é compatível com programas Java já existentes. Existe também uma implementação alternativa para a plataforma .NET .
- Scala é tipado estaticamente e fortemente orientado a objeto, seu código é compilado em Java bytecode o que permite que ele use bibliotecas do Java sem a necessidade de uma “camada de cola”.

# TIPAGEM ESTÁTICA

- O Scala é equipado com um sistema de tipos expressivo, que reforça estaticamente que abstrações sejam usadas de maneira segura e coerente. Em particular, o sistema de tipos suporta:
  - classes genéricas;
  - anotações de variância;
  - tipos de limite superior e inferior ;
  - classes e enumeração (tipo de dado);
  - tipo composto;
  - auto referencia;
  - métodos polimórficos.

# CARACTERÍSTICAS DA ORIENTAÇÃO A OBJETO

- Scala é puramente orientada-a-objetos no sentido de todo o valor ser um Objeto. Tipos e comportamento dos objetos são descritos pelas classes e traços. Abstrações de classes são estendidas por "subclassing" e um flexível mecanismo de "mixin-based composition" como uma forma "limpa" de substituição a heranças múltiplas.

# PROGRAMAÇÃO FUNCIONAL

- Scala é também uma linguagem funcional, no sentido de cada função ser um valor. Scala fornece uma sintaxe "leve" para definição de funções anônimas, ela suporta higher-order functions, ela permite que funções sejam aninhadas, e suporta currying. Classes Scala e seu suporte interno para pattern matching modela tipos algébricos usados em muitas linguagens de programação funcionais. Além disso, sua noção de "pattern matching" natural é estendida para o processamento de arquivos XML com a ajuda de expressões regulares. Neste contexto, a linguagem se torna de grande valia para o processamento de sintaxes futuras. Estas características, fazem com que o Scala seja ideal para o desenvolvimento de aplicações como web services.

# CÓDIGO

- Scala é um software de código aberto.
- O nome Scala vem da mistura de “Scalable” e “Language”, significando que ele foi desenvolvido para crescer de acordo com as demandas dos usuários.

# HISTÓRIA

- Scala foi inventada na École Polytechnique Fédérale de Lausanne (EPFL), Lausana na Suíça, em 2001 por Martin Odersky (ex-integrante do Java Generics). Foi liberada publicamente na plataforma Java em janeiro de 2004, e na plataforma .NET em junho do mesmo ano. Uma segunda versão da linguagem foi liberada em Março de 2006.
- Em 17 de Janeiro de 2011 o Time Scala ganhou uma bolsa de pesquisa de 5 anos no valor de 2,3 milhões de Euros do European Reserach Council. E em maio do mesmo ano, Odersky e alguns colaboradores fundaram a Typesafe, uma empresa para dar suporte comercial, treinamento e serviços para Scala.



# PLATAFORMAS

- Scala roda em 3 diferentes plataformas, são elas:
  - JVM (Um dos motivos do sucesso da linguagem).
  - Android
  - .NET
-

# IDE

- Existem plugins de desenvolvimento Scala para as IDEs mais conhecidas atualmente. São elas:
  - Eclipse
  - NetBeans (mais completo).
  - IntelliJ (menos completo).
  
  - Principais funções: Highlight, hyperlink, auto-complete, marcadores de erro, visualização hierárquicas das classes, pacotes. Build, run e debug.
  - Debugger só nas versões Eclipse e NetBeans.
-

# DOCUMENTAÇÃO E BIBLIOTECAS

- Documentação online completa. (Igual ao JAVA)
- Códigos exemplos e guias para iniciantes na página oficial da linguagem (<http://www.scala-lang.org/>), inclusive para Android.
- Assim como JAVA, existem diversas bibliotecas que podem ser usadas pelo desenvolvedor (apenas importando-as) que foram feitas tanto pelos criadores da linguagem, como por outros desenvolvedores.
- Essas bibliotecas vão desde conexão com banco de dados, teste de software até bibliotecas de processamento de algebra.

# DIFICULDADE NO APRENDIZADO

- Scala é uma linguagem bem mais recente que JAVA, por isso não tem tanto conteúdo e fórum destinados a sua programação. Assim, fica mais difícil de tirar dúvidas.
- Pelo motivo acima, até no próprio site da Scala diz que é uma linguagem que seria bom aprender depois de JAVA, para já ter todos os conceitos e conhecimentos de orientação a objeto.
- Porém nada impede de algum programador começar usando Scala, apenas terá um pouco mais de dificuldade do que um programador JAVA no início.

# PORQUÊ SCALA?

- Evita repetição de código.
  - Consegue-se escrever menos para expressar a mesma coisa.
  - Interoperabilidade (proposta).
  - Roda na JVM, ou seja, pode obter bibliotecas já implementadas em Java.
  - Trabalha muito melhor que JAVA para fazer maior aproveitamento da programação concorrente.
  - Tuplas.
-

# PORQUÊ SCALA?

```
//Java
```

```
boolean nameHasUpperCase=false;
```

```
for(int i=0; i < name.length(); ++i) {
```

```
    if( Character.isUpperCase(name.charAt(i)) ) {
```

```
        nameHasUpperCase=true;
```

```
        break;
```

```
    }
```

```
}
```

```
//Scala
```

```
val nameHasUpperCase = name.exists( .isUpper)
```

# PORQUÊ SCALA?

```
//Java
```

```
conta.deposito(1000);
```

```
//Scala
```

```
conta deposito 1000
```

Como não existe diferenciação entre valores primitivos, conseguimos criar listas com diferentes tipos:

```
var a = {2, "Casa"}
```

```
var b = { 1, {5, "Texto", true}, false }
```

# VANTAGENS

- Não existe tipo primitivo, tudo é objeto.
  - Importa bibliotecas e possui interoperabilidade em Java e C#.
  - Alterações inclusive a níveis de compilação.
  - Portabilidade (projeto scala-llvm).
  - Muito eficiente para aplicações web-services.
-



# DESVANTAGENS

- Complexidade no que diz respeito a liberdade originado pela linguagem de alguns códigos e parametrizações
- Problemas com Java devido algumas bibliotecas não estarem preparadas com tecnologia Scala

# MERCADO

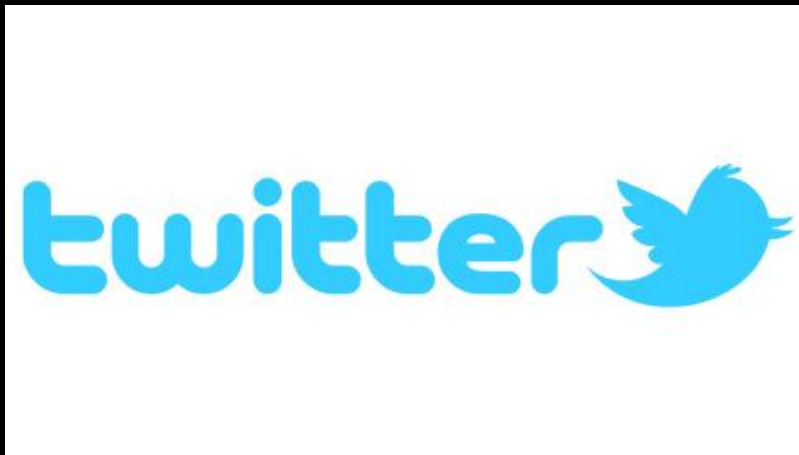
- Para profissionais, como toda área de tecnologia é altamente deficiente no que diz respeito a número de pessoas comparado a número de vagas.
  - Assim como Java é uma potência no meio tecnológico, a linguagem Scala possui potencial para tal.
  - A linguagem Scala já conseguiu grandes clientes ao longo de sua existência.
-

# SOFTWARES CONHECIDOS

- LinkedIn



- Twitter



# EMPRESAS QUE JÁ USARAM SCALA

- Sony (deixou todo seu código aberto, qualquer um pode pegá-lo).

The Sony logo is displayed in a bold, black, sans-serif font. The letters are thick and closely spaced, with a classic, slightly rounded design. The logo is centered within a white rectangular background.

- Siemens

The Siemens logo is displayed in a bold, teal-colored, sans-serif font. The letters are thick and closely spaced, with a modern, clean design. The logo is centered within a white rectangular background.