



02. Que venham os lambdas!

📅 Date	@28/10/2022
📁 Categoria	Java
📖 Curso	Java 8: conheça as novidades dessa versão

Tópicos

- Que venham os lambdas!
- Métodos de uma interface funcional
- Escrevendo o forEach com lambda
- Método sort com lambda
- Entendendo o tipo de um lambda
- Threads com lambda!

Que venham os lambdas!

Tendo essas dificuldade e verbosidade da sintaxe das classes anônimas em vista, o Java 8 traz uma nova forma de implementar essas interfaces ainda mais sucinta. É a sintaxe do lambda. Em vez de escrever a classe anônima, deixamos de escrever alguns itens que podem ser inferidos.

Como essa interface só tem um método, não precisamos escrever o nome do método. Também não daremos new. Apenas declararemos os argumentos e o bloco a ser executado, separados por `->`:

```
palavras.forEach((String s) -> {  
    System.out.println(s);  
});
```

Métodos de uma interface funcional

Quantos métodos uma `interface funcional` pode ter?

Um único método abstrato. Além desse método ela pode ter outros métodos, contanto que sejam `default` ou 'static'. Essa estrutura é fundamental, pois assim o compilador sabe exatamente que o corpo da expressão lambda que escrevemos é a implementação de seu único método abstrato.

Threads com lambda!

Considere o seguinte código que executa um `Runnable` em uma `Thread`:

```
new Thread(new Runnable() {  
  
    @Override  
    public void run() {  
        System.out.println("Executando um Runnable");  
    }  
  
}).start();
```

Como podemos escrevê-lo usando uma expressão lambda?

```
new Thread(() -> System.out.println("Executando um Runnable")).start();
```

