

# Java 8 e Stream API

Java 8 foi uma das atualizações mais importantes da linguagem Java, lançada em março de 2014. Ela trouxe diversos recursos modernos que permitem escrever códigos mais concisos, legíveis e com foco em programação funcional.

Abaixo estão os principais recursos introduzidos no Java 8:

Recurso	O que faz?	Exemplo breve
Lambda Expressions	Funções mais curtas e funcionais	<code>(a, b) -&gt; a + b</code>
Stream API	Manipulação eficiente de listas e coleções com operações funcionais	<code>listOf(1, 2, 3, 4, 5).collect(...)</code>
Functional Interfaces	Interfaces com um único método, usadas com <code>Predicates</code>	<code>Predicate&lt;T&gt;</code> , <code>Consumer&lt;T&gt;</code>
Method References	Referência direta a métodos	<code>System.out::println</code>
Default Methods	Métodos com implementação em interfaces	<code>interface A { default void m() {} }</code>
Optional	Evita null com um container seguro	<code>Optional&lt;String&gt; nome = Optional.of("Maria")</code>
New Date/Time API	Nova API robusta para datas	<code>LocalDate.now()</code> , <code>LocalDateTime.of(...)</code>

## O que é Stream API?

A Stream API foi introduzida no Java 8 para permitir o processamento funcional de coleções de dados. Ela permite manipular listas e coleções usando operações como `filter`, `map`, `reduce`, `collect`, entre outras. Streams ajudam a escrever código mais limpo e eficiente, muitas vezes com menos linhas e de forma mais declarativa.