

Tópicos Abordados

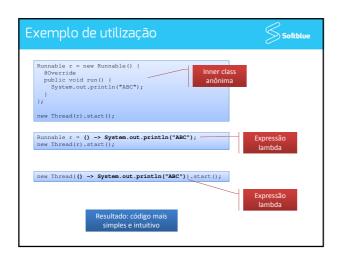


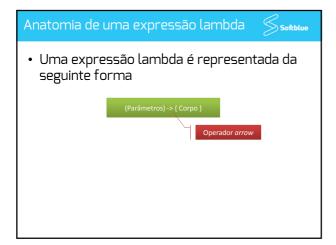
- O que são expressões lambda
- Exemplos de uso
- Anatomia
- Sintaxe
- Interfaces funcionais
 - Predicate<T>
 - Consumer<T>
 - Function<T, R>
- Novas funcionalidades em coleções
- Referências a métodos
- Closures

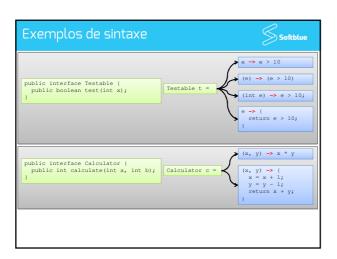
Expressões Lambda

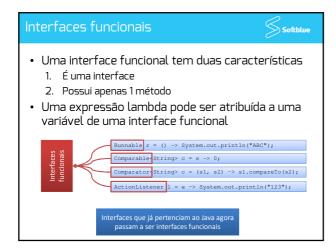


- Maior inovação da versão 8 do Java
- O nome lambda vem do conceito matemático de *cálculo lambda*
- Expressões lambda trazem o Java mais próximo do paradigma de programação funcional
- Uma expressão lambda pode ser considerada como um objeto
 - Pode ser referenciada por uma variável
 - Pode ser passada como parâmetro para métodos e utilizada como retorno
- Em Java, uma expressão lambda é utilizada em substituição a uma inner class anônima









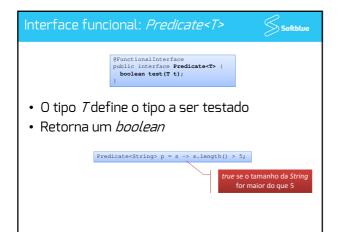


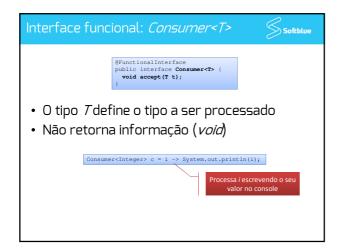


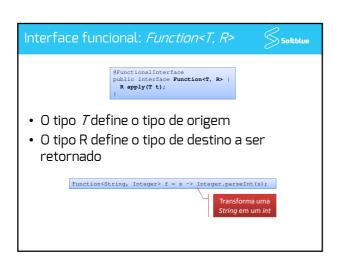
• Esta annotation define uma interface como interface funcional

@FunctionalInterface
public interface Generator {
 public String generate();
}

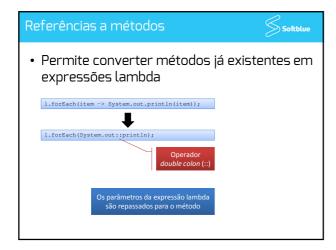
- Seu uso não é obrigatório
- Se @FunctionalInterface for utilizada, o compilador checa se a interface define apenas 1 método



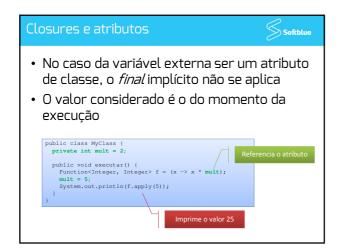














			-