

Tópicos Abordados

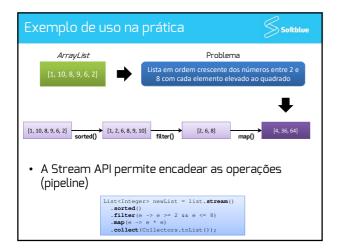


- O que é a Stream API
- Exemplo de uso na prática
- O tipo Stream<T>
- Operações
 - sorted()
 - limit()
 - filter()
 - distinct()
 - map()
 - collect()
- Referenciando construtores
- Streams paralelas

Stream AP



- API que permite combinar operações
 Usada principalmente em coleções do Java
- Introduzida no Java 8
- Aproveita o uso de expressões lambda
- Funciona bem para coleções pequenas e também para coleções muito grandes
 - Usa a abordagem de lazy evaluation



Obtendo um objeto *Stream<T>*



 As coleções List e Set possuem o método stream()

Stream<Integer> stream = list.stream();

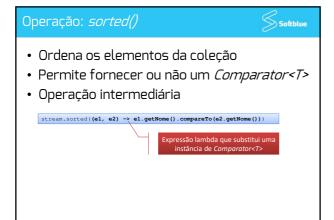
- O objeto *Stream<T>* é a porta de entrada para a Stream API
- O tipo parametrizado $\mathcal T$ depende do tipo da coleção
- Para arrays, o código muda

int[] array = new int[10];
Stream<int[]> stream = Stream.of(array);

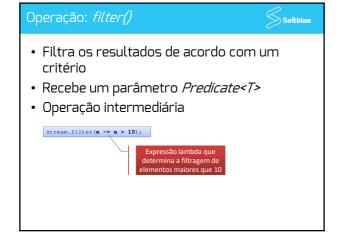
Operações da Stream API



- A Stream API possui uma série de operações para manipulação de dados
- 2 tipos
 - Intermediárias
 - Retornam um novo objeto *Stream<T>*
 - Possibilitam o pipeline
 - Ex: sorted(), limit(), filter(), map()
 - Terminais
 - Geram um resultado final (redução)
 - Finalizam o uso da stream
 - Ex: collect(), reduce(), count(), max()



Define um tamanho máximo para a coleção Os elementos que excedem o limite fornecido são removidos Operação intermediária | Stream.limit(10); | Só os 10 primeiros elementos são mantidos | São mantidos |



• Remove elementos duplicados • Operação intermediária stream.distinct(); Operação: map() Mapeia um elemento em outro elemento (transformação) • Recebe um parâmetro Function<T, R> • Operação intermediária stream.map(e -> e + 2); • Existem também mapeamentos especializados – mapToInt() -> IntStream – mapToDouble() -> DoubleStream - mapToLong-> LongStream • Finaliza o pipeline, gerando um resultado • Operação terminal • A classe *Collectors* tem métodos estáticos que auxiliam nesta operação • Exemplos: stream.collect(Collectors.toList()) stream.collect(Collectors.toSet()) stream.collect(Collectors.counting()) stream.collect(Collectors.summingInt(e -> e.getIdade()))

Operação: count() • Faz a contagem de elementos • Operação terminal long c = stream.count();

