Overview of Java 8 Streams (Part 2)

Douglas C. Schmidt <u>d.schmidt@vanderbilt.edu</u> www.dre.vanderbilt.edu/~schmidt



Professor of Computer Science

Institute for Software Integrated Systems

Vanderbilt University Nashville, Tennessee, USA



- Understand the structure & functionality of Java 8 streams, e.g.,
 - Fundamentals of streams
 - Common stream aggregate operations



Terminal operation (Function h)

- Understand the structure & functionality of Java 8 streams, e.g.,
 - Fundamentals of streams
 - Common stream aggregate operations
 - These operations apply to both sequential & parallel streams



- Understand the structure & functionality of Java 8 streams, e.g.,
 - Fundamentals of streams
 - Common stream aggregate operations
 - These operations apply to both sequential & parallel streams



Being a good streams programmer makes you a better parallel streams programmer

- Understand the structure & functionality of Java 8 streams, e.g.,
 - Fundamentals of streams
 - Common stream aggregate operations
 - These operations apply to both sequential & parallel streams

We'll use a simple sequential stream example to explain common Java 8 aggregate operations



See github.com/douglascraigschmidt/LiveLessons/tree/master/SimpleSearchStream

• This example finds words in an input string



See github.com/douglascraigschmidt/LiveLessons/tree/master/SimpleSearchStream

• This example finds words in an input string



See github.com/douglascraigschmidt/LiveLessons/tree/master/SimpleSearchStream

• This example finds words in an input string



• This example finds words in an input string







Convert collection to a (sequential) stream

• This example finds words in an input string



• This example finds words in an input string



• This example finds words in an input string

List

<String>

Stream < String >





Search for the word in each input string

• This example finds words in an input string



• This example finds words in an input string



• This example finds words in an input string



Remove empty search results from the stream

• This example finds words in an input string



• This example finds words in an input string



• This example finds words in an input string



Trigger intermediate operation processing

• This example finds words in an input string



Search Words

• The "physical" processing of a stream is different from the "logical" model we discussed here..



See <u>www.ibm.com/developerworks/library/j-java-streams-3-brian-goetz</u>

• An aggregate operation processes all elements in a stream



See en.wikipedia.org/wiki/Higher-order_function

• An aggregate operation processes all elements in a stream





The focus is on the "what" (declarative), not the "how" (imperative)

• An aggregate operation processes all elements in a stream



Common aggregate operations include map(), filter(), & collect()

• An aggregate operation processes all elements in a stream



See docs.oracle.com/javase/8/docs/api/java/util/stream/Stream.html#map

• An aggregate operation processes all elements in a stream



The # of output stream elements matches the # of input stream elements

• An aggregate operation processes all elements in a stream



The map() aggregate operation may change the type of the elements it processes

• An aggregate operation processes all elements in a stream



The map() aggregate operation may change the type of elements it processes



• An aggregate operation processes all elements in a stream



See docs.oracle.com/javase/8/docs/api/java/util/stream/Stream.html#filter

• An aggregate operation processes all elements in a stream



The # of output stream elements may be less than the # of input stream elements

• An aggregate operation processes all elements in a stream



The filter() aggregate operation *can't* change the type of elements it processes







• An aggregate operation processes all elements in a stream





Intermediate operations are "lazy" & don't run until a terminal operator is reached

• An aggregate operation processes all elements in a stream



See docs.oracle.com/javase/8/docs/api/java/util/stream/Stream.html#collect





End of Overview of Java 8 Streams (Part 2)