Java Streams: Visualizing
WordSearcher.printSlice()

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

Professor of Computer Science
Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA
Learning Objectives in this Part of the Lesson

- Visualize aggregate operations in SimpleSearchStream’s WordSearcher.printResults() method
- Visualize aggregate operations in SimpleSearch Stream’s WordSearcher.printSlice() method

```java
stream()
| collect(groupingBy(...))
| entrySet().stream()
| dropWhile(e -> notEqual(e, word))
| forEach(this::printResult)
```
Visualizing the Word Searcher.printSlice() Method
Visualizing the `WordSearcher.printSlice()` Method

- Prints a slice of the `listOfSearchResults` starting at a particular word

```java
wordSearcher.printSlice("La", results);
```

Word "La" appeared at indices [234|417|658|886|991|1207|1247|1269|1291|1339|1361|1742|1847|1863|1909|1949|2161|2254|2276|2283] with max index of 2283


This method shows the `collect(groupingBy())` & `dropWhile()` aggregate operations

*Prints a slice of the stream from “La” to “Ti”, while ignoring all entries from “Do” to “So”.*
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

**Input a list of search results**

List
<SearchResults>

stream()

The search results correspond to words to find: "do", "re", "mi", "fa", "so", "la", "ti".
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word.

List
<SearchResults>

Convert list to a (sequential) stream of search results.
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the list of search results starting at a particular word

Output a stream of search results

List
<SearchResults>

Stream
<SearchResults>

Print out results of each map entry (key = word & value = list of search results)
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

Input a stream of search results

List
<SearchResults>

Stream
<SearchResults>

stream()
collect(groupingBy(…))
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

List <SearchResults>

Stream <SearchResults>

stream()

collect(groupingBy(…))

Trigger “intermediate operation” processing
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

Create a map that groups words with the indices where each word was found.

```
List<SearchResults>
Stream<SearchResults>
Map<String, List<Result>>
```

```
stream()
collect(groupingBy(()))
```
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

Output a map of strings & lists of search results

List <SearchResults>
Stream <SearchResults>
Map<String, List<Result>>

\[\text{collect(groupingBy(\ldots))}\]
\[\text{stream(\ldots)}\]
• Prints a slice of the `listOfSearchResults` starting at a particular word

```
Input a map of strings & lists of search results
```

```
Map<String, List<Result>>
```

```
List<SearchResults>
```

```
Stream<SearchResults>
```

```
collect(groupingBy(...))
stream()
```

```
entrySet().stream()
```

Visualizing the `WordSearcher.printSlice()` Method
Visualizing the `WordSearcher.printSlice()` Method

- Prints a slice of the `listOfSearchResults` starting at a particular word

List `<SearchResults>`

Stream `<SearchResults>`

Map `<String, List <Result>>`

Obtain the entry set from the map & convert it into a stream
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

**Output a stream of map entries**

List
<SearchResults>

Stream
<SearchResults>

Map<String, List<Result>>

Stream<Entry<String, List<Result>>>

List<SearchResults>

stream()

collect(groupingBy(…))

entrySet().stream()
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

```java
Stream<List<SearchResults>>
List<Map<String, List<Result>>>
Map<String, List<Result>>
```

```
Input a stream of map entries
```

```
stream()
collect(groupingBy(...))
entrySet().stream()
dropWhile(e -> notEqual(e, word))
```
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

```
List<SearchResults>
Stream<SearchResults>
Map<String, List<Result>>
Stream<Entry<String, List<Result>>>
```

Slice the stream to contain remaining elements after dropping subset of elements that don't match `word`.

See [docs.oracle.com/javase/9/docs/api/java/util/stream/Stream.html#dropWhile](docs.oracle.com/javase/9/docs/api/java/util/stream/Stream.html#dropWhile)
The `notEqual()` method is simple:

```java
return !entry.getKey().equals(word)
```
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

```
List <SearchResults>
Stream <SearchResults>
Map<String, List <Result>>
Stream<Entry<String, List<Result>>>
```

- `dropWhile()` differs from `filter()` since it needn’t examine the entire input stream.

See [itnext.io/streams-api-new-features-after-java-8-f2df81bed5ac](http://itnext.io/streams-api-new-features-after-java-8-f2df81bed5ac)
Prints a slice of the `listOfSearchResults` starting at a particular word

```
Output a (sub)stream starting at "La".
```

```
List
<SearchResults>

Stream
<SearchResults>

Map<String, List<Result>>

Stream<Entry<String, List<Result>>>

Stream<Entry<String, List<Result>>>
```

```
visualizing the WordSearcher.printSlice() method
```

```
stream()}

collect(groupingBy(…))

entrySet().stream()

dropWhile(e -> notEqual(e, word))
```
Prints a slice of the `listOfSearchResults` starting at a particular word.

Input a (sub)stream starting at "La".

```
List<SearchResults>
Stream<SearchResults>
Map<String, List<Result>>
Stream<Entry<String, List<Result>>>
Stream<Entry<String, List<Result>>>
```

Visualizing the `WordSearcher.printSlice()` Method

- `stream()`
- `collect(groupingBy(...))`
- `entrySet().stream()`
- `dropWhile(e -> notEqual(e, word))`
- `forEach(this::printResult)`
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the listOfSearchResults starting at a particular word

List
\(<\text{SearchResults}>\)

Stream
\(<\text{SearchResults}>\)

Map\(<\text{String, List}<\text{Result}>\>)

Stream\(<\text{Entry}<\text{String, List}<\text{Result}>\>)

Stream\(<\text{Entry}<\text{String, List}<\text{Result}>\>)

Iterate thru each element in (sub)stream

See [docs.oracle.com/javase/8/docs/api/java/util/stream/Stream.html#forEach](docs.oracle.com/javase/8/docs/api/java/util/stream/Stream.html#forEach)
Visualizing the WordSearcher.printSlice() Method

- Prints a slice of the list of search results starting at a particular word

Stream <SearchResults>

Map<String, List<Result>>

Stream<Entry<String, List<Result>>>

List <SearchResults>

Print out results of each map entry (key = word & value = list of search results)
End of Java Streams: Visualizing Word Searcher.printSlice()