

Lambdas		Stream API - Aggregation	
<code>java.util.function</code>	Package	Those APIs consider the whole set	
<code>@FunctionalInterface</code>	Optional. Force a compiler error if the interface has more than one abstract method.	<code>.sorted()</code>	
<code>(String text) -> System.out.println(text)</code>	Anonymous function.	<code>.min()/max()</code>	
<code>(text) -> System.out.println(text)</code>		<code>.groupingBy()</code>	Gruppieren gleicher Elemente
<code>System.out::println</code>		<code>.distinct()</code>	make list unique based on lambda
Default/static method in interface		<code>.count()</code>	
Stream API - Generatoren		Stream API - Ergebnisse	
Iterate / Range	Automatische Erzeugung von IntStream durch Iteration	forEach / forEachOrdered	
Builder	Aufbau eines Streams mit Elementen ohne Collection	toArray(String[]::new)	
Of	Sammlung von Objekten als Argument	collect(Collectors.toList());	
Generate	Supplier Lambda zur Generierung von Einträgen	collect(Collectors.toCollection(ArrayList::new));	
Files (NIO), ZipFile, JarFile, Patterns (Regex)		Stream API	
Stream API - Transformation		<pre>List<Person> persons1, persons2; Stream.of(persons1, persons2) // stream of lists .flatMap(ps -> ps.stream()) // stream of obj .map(p -> p.firstName + " " + p.name) // stream of name .forEach(System.out::println); IntStream.iterate(0, i -> i + 1) .limit(10) .forEach(System.out::print); Stream.builder() .add("Wir ") .add("bauen ") .add("einen Satz.") .build() .forEachOrdered(...); Stream.generate(() -> Double.valueOf(Math.random() * 100).intValue()) .limit(5).forEach(...)</pre>	
map(lambda)	Objekte von Eingabetyp in Ausgabebetyp umwandeln		
flatMap(lambda)	Collection-Elemente in Stream-Elemente begradigen (Array[][] -> Array[])		
Stream API - Combining / Shortening		Java I/O, NIO	
<code>.filter(p -> p.name.startsWith("M"))</code>		Files.lines(Paths.get(this.class.getResource("file.txt")).toUR .count();	
<code>.reduce(x1, x2) -> x1 + x2</code>	$y = ((x1 + x2) + x3) + \dots$		
<code>.collect(Collectors.joining(", "));</code>	$e1 + ", " + e2 + \dots$		
<code>.limit(5)</code>	limit number of results		
<code>.skip(5)</code>	skip elements		



By **marb**
cheatography.com/marb/

Not published yet.
Last updated 9th December, 2016.
Page 1 of 2.

Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours
<https://apollopod.com>

Date/Time API

LocalTime

LocalDate

LocalDateTime

ZonedDateTime.of() / .now(Zoned.of("CET"));

Instant Maschinenzeit, genauer, ohne Zeitzone

TemporalAdjuster

Duration

DateTimeFormatter.ISO_OFFSET_DATE_TIME

```
// 2011-12-03T10:15:30+01:00
```

DateTimeFormatter.ofLocalizedDate(FormatStyle.LONG)

```
// June 20, 2014; 10:15:58 GMT+1
```

Optional

```
Optional<String> result = doSomething();
```

```
if ( result.isPresent()) result.get();
```

```
String result = doSomething().orElse("unknown");
```

```
doSomething().ifPresent(lambda);
```

Repeatable Annotation

```
@Repeatable(TableAnnos.class)
@Retention(RetentionPolicy.RUNTIME)
public @interface Table {
    String value();
}

@Retention(RetentionPolicy.RUNTIME)
public @interface TableAnnos {
    Table[] value();
}

@Table("Customer")
@Table("Employee")
public class Person {
}
```

C

By **marb**

cheatography.com/marb/

Not published yet.

Last updated 9th December, 2016.

Page 2 of 2.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopod.com>