

Java Stream API - Complete Cheat Sheet

1. Stream Initialization

`Stream.of(T... values)` - Creates a stream from values

`Arrays.stream(T[] array)` - From array

`List.stream()` - From list

`Stream.empty()` - Creates empty stream

`Stream.generate(Supplier)` - Infinite stream

`Stream.iterate(seed, UnaryOperator)` - Infinite iterative stream

`IntStream.range(start, end)` - Int stream (exclusive)

`IntStream.rangeClosed(start, end)` - Int stream (inclusive)

`Files.lines(Path)` - Stream of lines from file

2. Intermediate Operations

`filter(Predicate)` - Filter elements

`map(Function)` - Transform elements

`flatMap(Function)` - Flatten nested structure

`distinct()` - Remove duplicates

`sorted()` - Sort naturally

`sorted(Comparator)` - Sort with comparator

`limit(long)` - Limit elements

`skip(long)` - Skip elements

`peek(Consumer)` - Inspect/debug

`mapToInt(ToIntFunction)` - Convert to `IntStream`

`mapToLong(ToLongFunction)` - Convert to `LongStream`

`mapToDouble(ToDoubleFunction)` - Convert to `DoubleStream`

`boxed()` - Convert primitive to boxed stream

`takeWhile(Predicate)` - Takes while condition true (Java 9+)

`dropWhile(Predicate)` - Drops while condition true (Java 9+)

3. Terminal Operations

`forEach(Consumer)` - Perform action for each

`toArray()` - Convert to array

`reduce(...)` - Reduce elements to single value

`collect(Collector)` - Collect results

min(Comparator) - Minimum element

max(Comparator) - Maximum element

count() - Count elements

anyMatch(Predicate) - If any match condition

allMatch(Predicate) - If all match condition

noneMatch(Predicate) - If none match condition

findFirst() - Return first element

findAny() - Return any element

4. Common Collectors

Collectors.toList() - Collect to List

Collectors.toSet() - Collect to Set

Collectors.toMap(keyMapper, valueMapper) - To Map

Collectors.joining() - Join strings

Collectors.joining(delimiter) - Join with delimiter

Collectors.joining(delimiter, prefix, suffix) - Join with affixes

Collectors.counting() - Count elements

Collectors.averagingInt(ToIntFunction) - Average (int)

Collectors.averagingLong(ToLongFunction) - Average (long)

Collectors.averagingDouble(ToDoubleFunction) - Average (double)

Collectors.summarizingInt(ToIntFunction) - Summary stats (int)

Collectors.groupingBy(Function) - Group by key

Collectors.groupingBy(Function, Collector) - Group + downstream

Collectors.partitioningBy(Predicate) - Split true/false

Collectors.mapping(Function, Collector) - Map before collect

Collectors.reducing(...) - Reduce with collector

5. Primitive Stream Methods (IntStream, etc.)

sum() - Sum of elements

average() - Average

min(), max() - Min/Max values

summaryStatistics() - Count, sum, min, max, average