Apa itu Java Stream?

Java Stream adalah API baru yang diperkenalkan dalam Java 8 untuk memanipulasi koleksi data seperti array atau list dengan cara yang lebih fungsional dan ekspresif. Stream memungkinkan Anda untuk melakukan operasi seperti filter, map, reduce, dan lainnya pada data koleksi dengan lebih mudah dan ringkas.

Contoh 1: Membuat Stream

Anda dapat membuat Stream dari berbagai jenis koleksi, seperti List, Set, dan array. Berikut ini adalah contoh membuat Stream dari sebuah List:

**import java.util.Arrays;**

**import java.util.List;**

**import java.util.stream.Stream;**

**public class CreateStreamExample {**

**public static void main(String[] args) {**

**List<String> fruits = Arrays.asList("Apel", "Jeruk", "Mangga", "Pisang");**

**// Membuat Stream dari List**

**Stream<String> stream = fruits.stream();**

**// Menggunakan Stream**

**stream.forEach(System.out::println);**

**}**

**}**

Contoh 2: Filter dan Map pada Stream

Stream memungkinkan Anda untuk dengan mudah melakukan operasi seperti filter dan map. Berikut adalah contoh penggunaannya:

**import java.util.Arrays;**

**import java.util.List;**

**import java.util.stream.Collectors;**

**public class FilterAndMapExample {**

**public static void main(String[] args) {**

**List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);**

**// Menggunakan Stream untuk memfilter angka genap dan menggandakannya**

**List<Integer> evenDoubled = numbers.stream()**

**.filter(n -> n % 2 == 0)**

**.map(n -> n \* 2)**

**.collect(Collectors.toList());**

**System.out.println(evenDoubled);**

**}**

**}**

Contoh 3: Operasi Reduce pada Stream

Operasi `reduce` digunakan untuk menggabungkan semua elemen dalam Stream menjadi satu nilai. Berikut adalah contoh penggunaannya untuk menjumlahkan semua elemen dalam Stream:

**import java.util.Arrays;**

**import java.util.List;**

**public class ReduceExample {**

**public static void main(String[] args) {**

**List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5);**

**// Menggunakan Stream untuk menjumlahkan semua elemen**

**int sum = numbers.stream()**

**.reduce(0, (a, b) -> a + b);**

**System.out.println("Jumlah: " + sum);**

**}**

**}**

Contoh 4: Operasi Collect pada Stream

Operasi `collect` digunakan untuk mengumpulkan elemen-elemen dari Stream ke dalam sebuah koleksi seperti List, Set, atau Map. Berikut adalah contoh penggunaannya untuk mengumpulkan elemen-elemen yang lebih besar dari 3 ke dalam sebuah List:

**import java.util.Arrays;**

**import java.util.List;**

**import java.util.stream.Collectors;**

**public class CollectExample {**

**public static void main(String[] args) {**

**List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5);**

**// Menggunakan Stream untuk mengumpulkan elemen yang lebih besar dari 3 ke dalam List**

**List<Integer> filteredList = numbers.stream()**

**.filter(n -> n > 3)**

**.collect(Collectors.toList());**

**System.out.println(filteredList);**

**}**

**}**