7. Working with Streams

Demo1.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
import java.util.stream.Stream;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */

public class Demo1 {
public static void main(String[] args) {

List<Integer> numsList = Arrays.asList(1,2,3,4,5,6,7,8,9);
System.out.println(numsList);
System.out.println("-------------------");

Stream<Integer> mystream = numsList.stream(); //1
Stream<Integer> oddStream = mystream.filter(num -> num % 2 !=0);//2
Stream<Integer> squareStream = oddStream.map(num -> num * num);//2
squareStream.forEach(System.out::println); //3

System.out.println("-------------------");

numsList.stream().filter(num -> num % 2 ==0)
.map(num -> num * num)
.forEach(System.out::println);

System.out.println("-------------------");
System.out.println(numsList);
}
}
```

Demo2.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
import java.util.stream.Stream;
/*
```

```
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo2 {
public static void main(String[] args) {

List<Integer> numsList = Arrays.asList(1,2,3);

Stream<Integer> mystream1 = numsList.stream(); //1
mystream1.forEach(System.out::println); //3
//mystream1.forEach(System.out::println); //3 //Exception

System.out.println("-------------");
Stream<Integer> mystream2 = numsList.stream(); //1
mystream2.forEach(System.out::println); //3

System.out.println("-------------");
Stream<Integer> mystream3 = numsList.stream(); //1
mystream3.forEach(System.out::println); //3
}
}
```

Demo3.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo3 {
public static void main(String[] args) {

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

numsList.stream().forEach(System.out::println);
System.out.println("---------------------");
numsList.parallelStream().forEach(System.out::println);
}
}
```

Demo4.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.Collection;
import java.util.List;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo4 {
public static void main(String[] args) {

List<Integer> list1 = Arrays.asList(1,2,3);
List<Integer> list2 = Arrays.asList(10,20,30);

List<List<Integer>> mylist = Arrays.asList(list1,list2);

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

System.out.println("--------------------");

mylist.stream()
.map(Collection::stream)
.forEach(System.out::println);

System.out.println("--------------------");

mylist.stream()
.map(Collection::stream)
.flatMap(input -> input)
.forEach(System.out::println);
}
}
```

Demo5.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo5 {
public static void main(String[] args) {

List<Integer> mylist = Arrays.asList(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16);

mylist.stream().limit(10).forEach(System.out::println);

System.out.println("---------------------");
mylist.stream()
.limit(10)
.filter(num -> num%2!=0)
.map(num -> num * num)
.forEach(System.out::println); //3

}
}
```

Demo6.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo6 {
public static void main(String[] args) {

List<Integer> mylist = Arrays.asList(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16);

mylist.stream().skip(4).limit(9).forEach(System.out::println);

System.out.println("---------------------");
mylist.stream()   //1
.skip(4)
.limit(9)
.filter(num -> num%2!=0)
.map(num -> num * num)
.forEach(System.out::println); //3
}
}
```

Demo7.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo7 {
public static void main(String[] args) {

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

System.out.println("---------------------");
mylist.stream()    //1
.skip(3)
.limit(5)
.peek(System.out::println)
.filter(num -> num%2!=0)
.peek(System.out::println)
.map(num -> num * num)
.forEach(System.out::println); //3

}
}
```

Demo8.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo8 {
public static void main(String[] args) {

List<Integer> mylist = Arrays.asList(5,2,7,5,8,3,9,9,4,2,3,7,1,6,1);
mylist.stream()
.limit(15)
.filter(num -> num%2!=0)
.map(num -> num * num)
.sorted()
.forEach(System.out::println);

System.out.println("---------------------");
mylist.stream()
.limit(15)
.distinct()
.filter(num -> num%2!=0)
.map(num -> num * num)
.sorted()
.forEach(System.out::println);
}
}
```

Demo9.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
import java.util.stream.Stream;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo9 {
public static void main(String[] args) {

System.out.println("-----1------");
Stream<Integer> myStream1 =  Stream.empty();
myStream1.forEach(System.out::println);

System.out.println("------2------");
Stream<Integer> myStream2 = Stream.of(11,12,13);
myStream2.forEach(System.out::println);

List<Integer> numsList = Arrays.asList(11,12,13,14,15);

System.out.println("-----3------");
Stream<Integer> myStream3 = numsList.stream();
myStream3.forEach(System.out::println);
System.out.println("------4-------");
Stream<Integer> myStream4 =  numsList.parallelStream();
myStream4.forEach(System.out::println);
}
}
```

Demo10.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
import java.util.stream.Stream;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo10 {
public static void main(String[] args) {

Stream<Integer> myStream1 = Stream.of(11,12,13);

List<Integer> numsList = Arrays.asList(21,22,23);
Stream<Integer> myStream2 = numsList.stream();

Stream<Integer> myStream3 =Stream.concat(myStream2, myStream1);
myStream3.forEach(System.out::println);
}
}
```

Demo11.java

```
package com.jlcindia.demos;

import java.util.stream.Stream;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */

public class Demo11 {
public static void main(String[] args) {
```

```
Stream.iterate(11, n-> n+1)
.limit(10)
.forEach(System.out::println);

System.out.println("-------------------");
Stream.iterate(11, n-> n+1)
.limit(10)
.filter(num -> num %2==0)
.forEach(System.out::println);

System.out.println("-------------------");
Stream.iterate(11, n-> n+1)
.limit(10)
.filter(num -> num %2!=0)
.forEach(System.out::println);

}
}
```

Demo12.java

```
package com.jlcindia.demos;

import java.util.stream.Stream;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo12 {

public static void main(String[] args) {

Stream.iterate(101, n-> n+1)
.skip(25)
.limit(10)
.forEach(System.out::println);

System.out.println("-------------------");
Stream.iterate(101, n-> n+1)
.skip(25)
.limit(10)
.filter(num -> num %2==0)
.forEach(System.out::println);
System.out.println("-------------------");
Stream.iterate(101, n-> n+1)
.skip(25)
.limit(10)
.filter(num -> num %2!=0)
.forEach(System.out::println);
}
}
```

Demo13.java

```
package com.jlcindia.demos;

import java.util.Random;
import java.util.stream.Stream;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo13 {
public static void main(String[] args) {

Stream.generate(() -> (new Random()).nextInt(100))
.limit(10)
.forEach(System.out::println);

System.out.println("-------------------");
Stream.generate(() -> (new Random()).nextInt(100))
.limit(10)
.filter(num -> num % 2 != 0)
.map(num -> num * num)
.forEach(System.out::println);

System.out.println("-------------------");
Stream.generate(() -> (new Random()).nextInt(100))
.limit(10)
.filter(num -> num % 2 == 0)
.map(num -> num * num)
.forEach(System.out::println);
System.out.println("Done!!!");
}
}
```

Demo14.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo14 {
public static void main(String[] args) {

List<Integer> numsList = Arrays.asList(1, 2, 3, 4, 5);
int sumResult1 = numsList.stream().reduce(0, (a, b) -> a + b);
System.out.println(sumResult1);
System.out.println("-------------------");

int sumResult2 = numsList.stream().reduce(0, Integer::sum);
System.out.println(sumResult2);
System.out.println("-------------------");

int mulResult = numsList.stream().reduce(1, (a, b) -> a * b);
System.out.println(mulResult);
System.out.println("-------------------");

Integer lastElement = numsList.stream()
.reduce((num1, num2) -> num2)
.orElse(-1);

System.out.println(lastElement);
}
}
```

Demo15.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
import java.util.Optional;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
```

```
public class Demo15 {
public static void main(String[] args) {

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


mylist.stream()
.filter(num -> num%2!=0)
.map(num -> num * num)
.forEach(System.out::println); //3
System.out.println("---------------------");

long count= mylist.stream()
.filter(num -> num%2!=0)
.map(num -> num * num)
.count();
System.out.println(count);
System.out.println("---------------------");

Optional<Integer> maxNum = mylist.stream()
.filter(num -> num%2!=0)
.map(num -> num * num)
.max((num1, num2) -> num1.compareTo(num2));
System.out.println(maxNum);
maxNum.ifPresent(System.out::println);

System.out.println("---------------------");
Optional<Integer> minNum = mylist.stream()
.filter(num -> num%2!=0)
.map(num -> num * num)
.min((num1, num2) -> num1.compareTo(num2));
System.out.println(minNum);
minNum.ifPresent(System.out::println);

System.out.println("---------------------");

System.out.println("Done!!!");
}
}
```

Demo16.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
import java.util.Optional;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */

public class Demo16 {
public static void main(String[] args) {

List<Integer> numList = Arrays.asList(11,22,33,44,55,66,77,88,99);

Optional<Integer>  mycourse1 =numList.stream().findAny();
mycourse1.ifPresent(System.out::println);
System.out.println("-------------");

Optional<Integer>  mycourse2 =numList.stream().findFirst();
mycourse2.ifPresent(System.out::println);
System.out.println("-------------");

Optional<Integer>  mycourse3 =numList.stream().parallel().findAny();
mycourse3.ifPresent(System.out::println);
System.out.println("-------------");

Optional<Integer>  mycourse4 =numList.stream().parallel().findFirst();
mycourse4.ifPresent(System.out::println);

System.out.println("-------------");

System.out.println("Done!!!");
}
}
```

Demo17.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.LinkedList;
import java.util.List;
import java.util.stream.Collectors;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */

public class Demo17 {
public static void main(String[] args) {

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

List<Integer> mylist1 = numsList.stream()
.filter(num -> num % 2 == 0)
.map(num -> num * num)
.sorted()
.collect(Collectors.toList());

System.out.println(mylist1);

System.out.println("-------------");
List<Integer> mylist2 = numsList.stream()
.filter(num -> num % 2 == 0)
.map(num -> num * num)
.sorted()
.collect(Collectors.toCollection(LinkedList::new));

System.out.println(mylist2);

System.out.println("-------------");

System.out.println("Done!!!");
}
}
```

Demo18.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.stream.Collectors;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */

public class Demo18 {
public static void main(String[] args) {

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

System.out.println("-------------");
Set<Integer> myset1 = numsList.stream()
.filter(num -> num % 2 == 0)
.map(num -> num * num)
.sorted()
.collect(Collectors.toSet());

System.out.println(myset1);

System.out.println("-------------");
Set<Integer> myset2 = numsList.stream()
.filter(num -> num % 2 == 0)
.map(num -> num * num)
.sorted()
.collect(Collectors.toCollection(TreeSet::new));

System.out.println(myset2);

System.out.println("-------------");
System.out.println("Done!!!");
}
}
```

Demo19.java

```
package com.jlcindia.demos;

import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;
/*
* @Author : Srinivas Dande
* @Company: Java Learning Center
* */
public class Demo19 {
public static void main(String[] args) {

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

long count = numsList.stream()
.filter(num -> num % 2 != 0)
.collect(Collectors.counting());

System.out.println(count);

long sumResult = numsList.stream()
.filter(num -> num % 2 != 0)
.collect(Collectors.reducing(0, (num1,num2)-> num1+num2));

System.out.println(sumResult);

long mulResult = numsList.stream()
.filter(num -> num % 2 != 0)
.collect(Collectors.reducing(1, (num1,num2)-> num1*num2));

System.out.println(mulResult);

List<String> courseList = Arrays.asList("Java","SpringBoot","DevOps");

String result  = courseList.stream()
.collect(Collectors.joining(" --- "));

System.out.println(result);
}
}
```