

NOTE: For Detailed codes, please check on my repository.

ASSIGNMENT -1: Simple Name Sorter using Lambda Expression

```
1 package com.Java8features;
2
3 import java.util.Arrays;
4 import java.util.List;
5 import java.util.stream.Stream;
6
7 public class Assignment_1 {
8
9     public static void main(String[] args) {
10         // TODO Auto-generated method stub
11         List<String> names=Arrays.asList("rohit","tarun","ajay","mohana","harish");
12         Stream<String> data=names.stream()
13             .sorted();
14         data.forEach(n-> System.out.println(n));
15
16
17     }
18 }
19
20
```

Problems Javadoc Declaration Console × Progress SonarQube Rule Description Error L

<terminated> Assignment_1 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4

ajay
harish
mohana
rohit
tarun

ASSIGNMENT -2: Safe Divider

```
1 package com.Java8features;
2
3 import java.util.Optional;
4 import java.util.Scanner;
5
6
7
8 public class Assignment_2 {
9     public static Optional<String> divide(int dividend ,int divisor){
10         if(dividend/divisor ==0) {
11             return Optional.of("Not Allowed");
12         }
13         double result=(double)dividend/divisor;
14         return Optional.of(String.valueOf(result));
15     }
16
17     public static void main(String[] args) {
18         // TODO Auto-generated method stub
19         Scanner sc=new Scanner(System.in);
20         int a=sc.nextInt();
21         int b=sc.nextInt();
22         divide(a,b).ifPresent(System.out::println);
23     }
24 }
25 }
26
```

Problems Javadoc Declaration Console × Progress SonarQube Rule Description

<terminated> Assignment_2 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.F

5
10
Not Allowed

Problems Javadoc Declaration Console × Progress SonarQube Rule Description

<terminated> Assignment_2 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.F

18
6
3.0

ASSIGNMENT -3: Greetings Printer

```
1 package com.Java8features;
2
3 @FunctionalInterface
4 interface Greeting{
5     void greet();
6 }
7 public class Assignment_3 {
8
9     public static void main(String[] args) {
10         // TODO Auto-generated method stub
11         Greeting g=()->System.out.println("Welcome to my party");
12         g.greet();
13     }
14
15 }
16
```

Problems Javadoc Declaration Console X Progress SonarQube Rule

<terminated> Assignment_3 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-ec

Welcome to my party

ASSIGNMENT -4: Default Power

```
1 package com.Java8featues;
2
3 interface human{
4     void work();
5     default void eat() {
6         System.out.println("Human need to eat compulsory");
7     }
8 }
9 class man implements human{
10
11     @Override
12     public void work() {
13         // TODO Auto-generated method stub
14         System.out.println("Human need to work for growth");
15     }
16
17 }
18 public class Assignment_4 {
19
20     public static void main(String[] args) {
21         // TODO Auto-generated method stub
22         man m=new man();
23         m.eat();
24         m.work();
25     }
26
27 }
28
```

Problems Javadoc Declaration Console × Progress SonarQube Rule Description

<terminated> Assignment_4 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.0

Human need to eat compulsory
Human need to work for growth

ASSIGNMENT -5: A-Name Finder

```
1 package com.Java8features;  
2  
3 import java.util.Arrays;  
4 import java.util.List;  
5 import java.util.stream.Stream;  
6  
7 public class Assignment_5 {  
8  
9     public static void main(String[] args) {  
10         // TODO Auto-generated method stub  
11         List<String> names=Arrays.asList("jacob", "david", "liam", "Adam", "Alaister");  
12         Stream<String> data=names.stream();  
13         data.filter(n->n.charAt(0)=='A')  
14             .forEach(n->System.out.println(n));  
15  
16     }  
17  
18 }  
19
```

Problems Javadoc Declaration Console X Progress SonarQube Rule Description Error Log

<terminated> Assignment_5 [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-wi

Adam

Alaister

ASSIGNMENT -6: Word Counter

```
1 package com.Java8featuers;  
2  
3 import java.util.Arrays;  
4 import java.util.List;  
5 import java.util.stream.Stream;  
6  
7 public class Assignment_6 {  
8  
9     public static void main(String[] args) {  
10         // TODO Auto-generated method stub  
11         List<String> names=Arrays.asList("klinger", "david", "liamliving", "Adam", "Alaister");  
12         Stream<String> data=names.stream();  
13         Long total=data.filter(n->n.length()>5)  
14             .count();  
15         System.out.println("Names with length greater than five : " + total);  
16  
17     }  
18  
19 }  
20
```

Problems Javadoc Declaration Console X Progress SonarQube Rule Description Error Log Terminal 1

<terminated> Assignment_6 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win32.x86_64.exe
Names with length greater than five : 3

ASSIGNMENT -7: Even Number Collector

```
1 package com.Java8features;  
2  
3 import java.util.Arrays;  
4 import java.util.List;  
5 import java.util.stream.Collectors;  
6 import java.util.stream.Stream;  
7  
8 public class Assignment_7 {  
9  
10     public static void main(String[] args) {  
11         // TODO Auto-generated method stub  
12         List<Integer> names=Arrays.asList(5,9,6,3,4,2);  
13         Stream<Integer> data=names.stream();  
14         List<Integer> even=data.filter(n->n%2==0)  
15             .collect(Collectors.toList());  
16         System.out.println("Even numbers : "+even);  
17     }  
18  
19 }  
20
```

Problems Javadoc Declaration Console × Progress SonarQube Rule
<terminated> Assignment_7 [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclips
Even numbers : [6, 4, 2]

ASSIGNMENT -8: Pretty Joiner

```
1 package com.Java8features;  
2  
3 import java.util.Arrays;  
4 import java.util.List;  
5 import java.util.stream.Collectors;  
6 import java.util.stream.Stream;  
7  
8 public class Assignment_8 {  
9  
10     public static void main(String[] args) {  
11         // TODO Auto-generated method stub  
12         List<String> names=Arrays.asList("jacob", "david", "liam", "Adam", "Alaister");  
13         Stream<String> data=names.stream();  
14         String pretty_joiner=data.collect(Collectors.joining(","));  
15         System.out.println(pretty_joiner);  
16     }  
17  
18 }  
19
```

Problems Javadoc Declaration Console × Progress SonarQube Rule Description Error Log

<terminated> Assignment_8 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-v
jacob,david,liam,Adam,Alaister

ASSIGNMENT -9: Parallel Square Printer

```
1 package com.Java8featues;  
2  
3 import java.util.Arrays;  
4 import java.util.List;  
5  
6 public class Assignment_9 {  
7  
8     public static void main(String[] args) {  
9         List<Integer> names=Arrays.asList(5,9,6,3,4,2);  
10        names.parallelStream()  
11            .forEach(n-> System.out.println("Squares of "+n+" :"+n*n));  
12    }  
13  
14 }  
15
```

Problems Javadoc Declaration Console × Progress SonarQube Rule Description

<terminated> Assignment_9 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.

Squares of 9 :81
Squares of 4 :16
Squares of 2 :4
Squares of 6 :36
Squares of 5 :25
Squares of 3 :9

ASSIGNMENT -10: Frequency Mapper

```
1 package com.Java8featues;  
2  
3 import java.util.Arrays;  
4 import java.util.List;  
5 import java.util.stream.Stream;  
6  
7 import java.util.*;  
8 import java.util.stream.Collectors;  
9  
10 public class Assignment_10 {  
11     public static void main(String[] args) {  
12         List<String> names = Arrays.asList("Anna", "Anna", "Brian", "Bella", "Ben", "Cathy");  
13  
14         Map<String, Long> nameCounts = names.stream()  
15             .collect(Collectors.groupingBy(name -> name, Collectors.counting()));  
16  
17         System.out.println(nameCounts);  
18     }  
19 }  
20
```

Problems Javadoc Declaration Console X Progress SonarQube Rule Description Error Log Terminal

<terminated> Assignment_10 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win
{Brian=1, Cathy=1, Ben=1, Bella=1, Anna=2}

ASSIGNMENT -11: Custom Filter Chain

```
1 package com.Java8features;  
2  
3 import java.util.Arrays;  
4 import java.util.List;  
5  
6 public class Assignment_12 {  
7     public static void main(String[] args) {  
8         List<String> names=Arrays.asList("jacob", "davey", "david", "Adam", "Alaister" );  
9         names.stream()  
10             .sorted()  
11             .forEach(System.out::println);  
12  
13     }  
14  
15 }  
16
```

Problems Javadoc Declaration Console X Progress SonarQube Rule Description Error Log Terminal 1

<terminated> Assignment_12 (1) [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win32.x86_64.exe

Adam
Alaister
davey
david
jacob

ASSIGNMENT -12: Method Reference Sorter

```
1 package com.Java8features;
2
3 import java.util.Arrays;
4 import java.util.List;
5 import java.util.function.Predicate;
6 import java.util.stream.Stream;
7
8 public class Assignment_11 {
9
10     public static void main(String[] args) {
11         // TODO Auto-generated method stub
12         List<String> names=Arrays.asList("Alaska", "Alex", "David", "Sara", "Alaister", "Amanda", "Tara" );
13         Predicate<String> isStart=n->n.charAt(0) == 'A';
14         Predicate<String> isEnd=n->n.charAt(n.length()-1) == 'a';
15         System.out.println("Names with start with a and ends with a");
16         names.stream()
17             .filter(isStart.and(isEnd))
18             .forEach(n->System.out.print(n+" "));
19         System.out.println();
20         System.out.println();
21         System.out.println("Names with start with a or ends with a");
22         names.stream()
23             .filter(isStart.or(isEnd))
24             .forEach(n->System.out.print(n+" "));
25         System.out.println();
26         System.out.println();
27         System.out.println("Names with not start with a" );
28         names.stream()
```

<terminated> Assignment_11 [Java Application] C:\Users\91798\Downloads\spring-tools-for-eclipse-4.31.0.RELEASE-e4.36.0-win32.win32.x86_64 (1)\sts

Names with start with a and ends with a
Alaska Amanda

Names with start with a or ends with a
Alaska Alex Sara Alaister Amanda Tara

Names with not start with a
David Sara Tara

JAVA 8 FEATURES ASSIGNMENTS OUTPUTS

BY KOMMOJU NAGA RAJESWAR ROHIT