**Java O&A Theory and Coding**

----------------------------------------Java Basics--------------------------------------------------------------

1. What is the difference between JDK and JRE?

2. What is Java Virtual Machine (JVM)?

3. What are the different types of memory areas allocated by JVM?

4. What is JIT compiler?

5. How Java platform is different from other platforms?

6. Why people say that Java is 'write once and run anywhere' language?

7. How does ClassLoader work in Java?

8. Do you think ‘main’ used for main method is a keyword in Java?

9. Can we write main method as public void static instead of public static

void?

10. In Java, if we do not specify any value for local variables, then what

will be the default value of the local variables?

11. Let say, we run a java class without passing any arguments. What will

be the value of String array of arguments in Main method?

12. What is the difference between byte and char data types in Java?

-------------------------------------------------------------------OOPS---------------------------------------------------------

13. What are the main principles of Object Oriented Programming?

14. What is the difference between Object Oriented Programming

language and Object Based Programming language?

15. In Java what is the default value of an object reference defined as an

instance variable in an Object?

16. Why do we need constructor in Java?

17. Why do we need default constructor in Java classes?

18. What is the value returned by Constructor in Java?

19. Can we inherit a Constructor?

20. Why constructors cannot be final, static, or abstract in Java?

---------------------------Inheritance---------------------------------

21. What is the purpose of ‘this’ keyword in java?

22. Explain the concept of Inheritance?

23. Which class in Java is superclass of every other class?

24. Why Java does not support multiple inheritance?

25. In OOPS, what is meant by composition?

26. How aggregation and composition are different concepts?

27. Why there are no pointers in Java?

28. If there are no pointers in Java, then why do we get

NullPointerException?

29. What is the purpose of ‘super’ keyword in java?

30. Is it possible to use this() and super() both in same constructor?

31. What is the meaning of object cloning in Java?

----------------------------Static---------------------------------------

32. In Java, why do we use static variable?

33. Why it is not a good practice to create static variables in Java?

34. What is the purpose of static method in Java?

35. Why do we mark main method as static in Java?

36. In what scenario do we use a static block?

37. Is it possible to execute a program without defining a main() method?

38. What happens when static modifier is not mentioned in the signature

of main method?

39. What is the difference between static method and instance method in

Java?

----------------------------Method Overloading and Overriding------------

40. What is the other name of Method Overloading?

41. How will you implement method overloading in Java?

42. What kinds of argument variations are allowed in Method

Overloading?

43. Why it is not possible to do method overloading by changing return

type of method in java?

44. Is it allowed to overload main() method in Java?

45. How do we implement method overriding in Java?

46. Are we allowed to override a static method in Java?

47. Why Java does not allow overriding a static method?

48. Is it allowed to override an overloaded method?

49. What is the difference between method overloading and method

overriding in Java?

50. Does Java allow virtual functions?

51. What is meant by covariant return type in Java?

------------------------Polymorphism----------------------------------------

52. What is Runtime Polymorphism?

53. Is it possible to achieve Runtime Polymorphism by data members in

Java?

54. Explain the difference between static and dynamic binding?

Abstraction

55. What is Abstraction in Object Oriented programming?

56. How is Abstraction different from Encapsulation?

57. What is an abstract class in Java?

58. Is it allowed to mark a method abstract method without marking the

class abstract?

59. Is it allowed to mark a method abstract as well as final?

60. Can we instantiate an abstract class in Java?

61. What is an interface in Java?

62. Is it allowed to mark an interface method as static?

63. Why an Interface cannot be marked as final in Java?

64. What is a marker interface?

65. What can we use instead of Marker interface?

66. How Annotations are better than Marker Interfaces?

67. What is the difference between abstract class and interface in Java?

68. Does Java allow us to use private and protected modifiers for

variables in interfaces?

69. How can we cast to an object reference to an interface reference?

------------------------------------Final---------------------------------------

70. How can you change the value of a final variable in Java?

71. Can a class be marked final in Java?

72. How can we create a final method in Java?

73. How can we prohibit inheritance in Java?

74. Why Integer class in final in Java?

75. What is a blank final variable in Java?

76. How can we initialize a blank final variable?

77. Is it allowed to declare main method as final?

------------------------------------Package-------------------------------------

78. What is the purpose of package in Java?

79. What is java.lang package?

80. Which is the most important class in Java?

81. Is it mandatory to import java.lang package every time?

82. Can you import same package or class twice in your class?

83. What is a static import in Java?

-----------------------------------------------------------------Collections --------------------------------------------------------

Q1. What is Java Collections Framework? List out some benefits of Collections framework?

Q2. What will be the problem if you do not override hashcode() method?

Q3. What differences exist between HashMap and Hashtable?

Q4. What is the benefit of Generics in Collections Framework?

Q5. How do WeakHashMap works?

Q6. What is difference between Array and ArrayList?

Q7. What is difference between ArrayList and LinkedList?

Q8. How to remove duplicates from ArrayList?

Q9. What is Java Priority Queue?

Q10. What is LinkedHashMap in Java?

Q11. What is difference between HashMap and Hashtable?

Q12. What is EnumSet?

Q13. What is the difference between fail-fast and fail-safe iterator?

Q14. What are concurrent collection classes?

Q15. What is difference between Enumeration and Iterator interface?

Q16. What is difference between Iterator and ListIterator?

Q17. What is a default capacity of ArrayList, Vector, HashMap, Hashtable and Hashset?

Q18. What is the difference between Collection and Collections?

Q19. What is the difference between HashSet and TreeSet?

Q20. What is the difference between Set and Map?

Q21. What is the difference between HashSet and HashMap?

Q22. What is the difference between HashMap and TreeMap?

Q23. What is the Dictionary class?

Q24. What are all the Classes and Interfaces that are available in the collections?

Q25. What is the difference between HashMap and ConcurrentHashMap?

Q26. How to make an ArrayList read only in Java?

Q27. Why ConcurrentHashMap is faster than Hashtable in Java?

Q28. What is the difference between peek(), poll() and remove() method of the Queue interface?

Q29. How HashMap works in Java?

Q30. How does HashMap handle collisions in java?

Q31. What is difference between arrayList and linkedList?

Q32. How Set/HashSet implement unique values?

Q33. What is Comparable and Comparator Interface in java?

Q34. Difference between containsKey(), keySet() and values() in HashMap.

-----------------------------------------------------------------------------On JAVA 8 Stream ----------------------------------------------------

Q1. What Is a Stream? How Does It Differ From a Collection?

Q2 What Is the Difference Between Intermediate and Terminal Operations

Q3 What does Java’8 Stream pipelining mean?

Q4. How to Create Streams in Java?

Q5. What is the difference between Collections API and Streams API?

Q6. What does the Stream map() function do? why you use it?

Q7. What does the Stream filter() method do? when you use it?

Q8. What does the Stream flatmap() function do? why you need it?

Q9. Difference Between Stream map() and flatMap() Functions

Q10. What new features does Java 8 have?

Q11. What are the Lambda Function's primary characteristics?

Q12 .What function does Java 8's limit() method serve?

Q13. What are the main features that JDK has added?

----------------------------------------Java 1.7 Basic and intermediate Coding Questions-------------------------------------------------------------

1) print helloworld 10 times.

o/p= Hello world 1

Hello World 2 ....

Hello World 10

2) print the 3rd highest from array.

3) reverse a given string. i/p="Capgemini"

4) write a code to find missing number in Integer List

Test cases :

a) 1,2,3,4,5,7,8,9,10

b)2,4,6,10,12,14,16,18,20

c)1,3,5,9,11,13,15,17

5) Write a program to get the occurrence of a specific letter in a given String.

i/p="Hello"

i/p=l

o/p=2

6) Write a java program for palindrome in normal way as well as using Command line arguments.

i/p="MADAM"

o/p MADAM is a Pallendrome

7) Write a program to sort a given list of intergers in ascending order

i/p list={2,4,0,1,56,23,4}

o/p list={0,1,2,4,4,23,56}

8) Write a program to check a given number is prime or not

i/p n=34

9) write a program to a sort a Array of String in ascending oreder alphabetially

i/p String s=["Capgemini","Accenture","TCS","EPAM"]

o/p =[Accenture,Capgemini,EPAM,TCS]

10) Write a program to check a given number is even or odd

i/p n=34

o/p = 34 is even number

11) write a program for fibonacci series;

12) write a program to return 2nd largest number from given array

i/p=[2,3,5,6,1,0]

o/p=6

13) Write a program to implement the defalut method in a Fuctional interface and override it.

14)write a program to Count the number of alphabets in a given sentence

i/p="Capgemini"

o/p = {C-1,a-1,p-1,g-1,e-1,m-1,i-2,n-1}

15)Write a program to store the List of prime numbers in a arrayList and print it.

16)Write a program to create POJO class of Product (Product Id, ProductName) and Store all values in any collection and Sort that values by ProductName in ascending order

17)write a program To count number of words in a given sentence.

i/p "Hello world"

o/p = 2 words

18)Write a program to Reverse a string="Capgemini Training"

19)Write a program count Each vowel frequency in a String(String="capgemini training")

20)Create a class having Fields id,name ,salary and sort it based on id and name using comparator and comparable

21) Given a string sentence, count the number of words in it and display count. The words are separated by the following characters: space (‘ ‘) or new line (‘\n’) or tab (‘\t’) or a combination of these.

22) write program to check whether a number is palindrome or not and

1) n=101

2) n=22022

23).Design a Prime Number between 2 to 30.

24).Swap 2 numbers without using 3rd variable.

25).Design a program to display magic number, accept a number and check it is a magic number or not.

26).Display a L shape \* symbol.

27).Accept a number , find the reverse of it.

28).Write a program to display Pyramid symbol.

29).Declare a TreeSet and store the 2 Employee Objects.

30).Accept a string line and check how many vowels are there.

31).How can you find first non-repeated character in a word.

32).How can you check a string can only have alphabets and not digits.

33).How to get matching characters in a string.

34).How to get non-matching character in a string.

35).How do you prove 2 strings are anagrams.44).Find out the count of occurance of particular character in a string.

36).How to verify if 2 strings are rotation mutually.

37).How to find out first character of string , which is not repeated.

38).How to search a missing numbers in an array, which contains integer from 1 to 100.

39).Determine largest and smallest elements of an array , which is not sorted.

40).Explain the Bubble Sort algorithm.

41).Explain the Merge Sort algorithm.

42).How to remove special character in a string that is lowercase.

43).what is process of deleting matched element from a LinkedList which is not sorted.

44).Find out the Fibonacci series between 1 to 30.

45).Find out factors of given numbers. n=34

46).Write a java program to count number of words in a string using HashMap.

47) given a Char array is there {'a','b','a','c'} Find out not duplicate character i.e. 'a'

48) Write a program to remove the duplicate elements from a list =[2,3,3,3,65,4,3,3,2]

49)Write a Java program to illustrate multiple catch block

50)Write a Java program to get the character at the given index within the String

51)Write a Java program to get the character (Unicode code point) at the given index within the String

52) Write a Java program that checks the letters of the second string are present in the first string. Return true otherwise false.

Input the first string:

Java

Input the second string:

Ja

Check first string contains letters from the second string:

true

53) Write a Java program that removes a specified word from a given text. Return the new string.

("Exercises Practice Solution", " Solution") -> "Exercises Practice"

("Red Green Blue", "Green") -> "Red Blue"

("Java Number Exercises”, “Java”) -> "Number Exercises"

54) Write a Java program to remove a specified character from a given string.

Original string:

abcdefabcdeabcdaaa

remove char from given string : "a"

Second string:

bcdefbcdebcd

55) Write a Java program to sort in ascending and descending order by length of the given array of strings.

Original unsorted colors: [Green, White, Black, Pink, Orange, Blue, Champagne, Indigo, Ivory]

Sorted color (descending order): [Champagne, Orange, Indigo, Green, White, Black, Ivory, Pink, Blue]

Sorted color (ascending order): [Pink, Blue, Green, White, Black, Ivory, Orange, Indigo, Champagne]

56) Write a Java program to create a new string repeating every character twice of a given string.

Sample Output:

The given string is: welcome

The new string is: wweellccoommee

57) Write a Java program to reverse words in a given string.

Sample Input / Output:

The given string is:= Reverse words in a given string

The new string after reversed the words:= string given a in words Reverse

58) Write a Java program to find the maximum occurring character in a string.

Sample Output:

The given string is: test string

Max occurring character in the given string is: t

59) Write a Java program to test if a given string contains the specified sequence of char values.

Sample Output:

Original String: PHP Exercises and Python Exercises

Specified sequence of char values: "and"

output : true

-------------------------------------------------------------------------------------

60) Given an "out" string length 4, such as "<<>>", and a word, return a new string where the word is in the middle of the out string, e.g. "<<word>>". Note: use str.substring(i, j) to extract the String starting at index i and going up to but not including index j.

makeOutWord("<<>>", "Yay") → "<<Yay>>"

makeOutWord("<<>>", "WooHoo") → "<<WooHoo>>"

makeOutWord("[[]]", "word") → "[[word]]"

public String makeOutWord(String out, String word) {

return out.substring(0,2)+word+out.substring(2,4);

}

--------------------------------------------------------------------------------

61) Given a string of even length, return the first half. So the string "WooHoo" yields "Woo".

firstHalf("WooHoo") → "Woo"

firstHalf("HelloThere") → "Hello"

firstHalf("abcdef") → "abc"

--------------------------------------------JAVA 1.8 (Intermediate) coding Questions ----------------------------------------------------------------------

Q1) Given a list of integers, find out all the even numbers exist in the list using Stream functions? list=[2,34,678,3]

Q2) Given a list of integers, find out all the numbers starting with 1 using Stream functions?

Q3) How to find duplicate elements in a given integers list in java using Stream functions?

Q4) Given the list of integers, find the first element of the list using Stream functions?

Q5) Given a list of integers, find the total number of elements present in the list using Stream functions?

Q6) Given a list of integers, find the maximum value element present in it using Stream functions?

Q7) Given a String, find the first non-repeated character in it using Stream functions?

Q8) Given a String, find the first repeated character in it using Stream functions?

Q9) Given a list of integers, sort all the values present in it using Stream functions?

Q8) Given a String, find the first repeated character in it using Stream functions?

Q9) Given a list of integers, sort all the values present in it using Stream functions?

Q10) Given a list of integers, sort all the values present in it in descending order using Stream functions?

Q11) Write a Java 8 program to concatenate two Streams?

Q12 ) Write a Java 8 program to find the number of Strings in a list whose length is greater

than 5?

Q13) Explain StringJoiner Class in Java 8? How can we achieve joining multiple Strings

using StringJoiner Class?

Q14)Find the Word/number got repeated by using Stream api

input:2,3,34,,5,5,4,3,3,3

output:2-1,3-4,34-1,5-2,4-1

15) Write a Java Lambda Expression to Create Thread

16) How Lambda Expression and Functional Interfaces are Related?

17)Given a list of Strings .sConvert String to uppercase and Join them with coma (,) using Stream api

List=["USA", "Japan", "France", "Germany", "Italy", "U.K.","Canada"]

18)Given a list of Integer find square of all distinct numbers By using stream api

list=[9, 10, 3, 4, 7, 3, 4]

19)Given a List of Integers find total count, min, max, sum, and the average for numbers by using Stream api

input :list=[2, 3, 5, 7, 11, 13, 17, 19, 23, 29]

output: count=10, sum=129, min=2, average=12.900000, max=29

------------------------------------------------------------------QUESTIONS--------------------------------------------------------------------------

Q1) Given a list of integers, find out all the even numbers exist in the list using Stream functions?

Q2) Given a list of integers, find out all the numbers starting with 1 using Stream functions?

Q3) How to find duplicate elements in a given integers list in java using Stream functions?

Q4) Given the list of integers, find the first element of the list using Stream functions?

Q5) Given a list of integers, find the total number of elements present in the list using Stream functions?

Q6) Given a list of integers, find the maximum value element present in it using Stream functions?

Q7) Given a String, find the first non-repeated character in it using Stream functions?

Q8) Given a String, find the first repeated character in it using Stream functions?

Q9) Given a list of integers, sort all the values present in it using Stream functions?

Q8) Given a String, find the first repeated character in it using Stream functions?

Q9) Given a list of integers, sort all the values present in it using Stream functions?

Q10) Given a list of integers, sort all the values present in it in descending order using Stream functions?

Q11) Write a Java 8 program to concatenate two Streams?

Q12 ) Write a Java 8 program to find the number of Strings in a list whose length is greater

than 5?

Q13) Explain StringJoiner Class in Java 8? How can we achieve joining multiple Strings

using StringJoiner Class?

Q14)Find the Word/number got repeated by using Stream api

input:2,3,34,,5,5,4,3,3,3

output:2-1,3-4,34-1,5-2,4-1

15) Write a Java Lambda Expression to Create Thread

16) How Lambda Expression and Functional Interfaces are Related?

17)Given a list of Strings .sConvert String to uppercase and Join them with coma (,) using Stream api

List=["USA", "Japan", "France", "Germany", "Italy", "U.K.","Canada"]

18)Given a list of Integer find square of all distinct numbers By using stream api

list=[9, 10, 3, 4, 7, 3, 4]

19)Given a List of Integers find total count, min, max, sum, and the average for numbers by using Stream api

input :list=[2, 3, 5, 7, 11, 13, 17, 19, 23, 29]

output: count=10, sum=129, min=2, average=12.900000, max=29

20) Given String str="telCo" Toggle it to str=TELcO

------------------------------------------------------------ANSWERS---------------------------------------------------------------------------------------

Q1 Given a list of integers, find out all the even numbers exist in the list using Stream functions?

import java.util.\*;

import java.util.stream.\*;

public class JavaHungry {

public static void main(String args[]) {

List<Integer> myList = Arrays.asList(10,15,8,49,25,98,32);

myList.stream()

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

.forEach(System.out::println);

}

}

Output:

10, 8, 98, 32

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q2 Given a list of integers, find out all the numbers starting with 1 using Stream functions?

import java.util.\*;

import java.util.stream.\*;

public class JavaHungry {

public static void main(String args[]) {

List<Integer> myList = Arrays.asList(10,15,8,49,25,98,32);

myList.stream()

.map(s -> s + "") // Convert integer to String

.filter(s -> s.startsWith("1"))

.forEach(System.out::println);

}

}

Output:

10, 15

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q3 How to find duplicate elements in a given integers list in java using Stream functions?

import java.util.\*;

import java.util.stream.\*;

public class JavaHungry {

public static void main(String args[]) {

List<Integer> myList = Arrays.asList(10,15,8,49,25,98,98,32,15);

Set<Integer> set = new HashSet();

myList.stream()

.filter(n -> !set.add(n))

.forEach(System.out::println);

}

}

Output:

98, 15

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q4 Given the list of integers, find the first element of the list using Stream functions?

import java.util.\*;

import java.util.stream.\*;

public class JavaHungry {

public static void main(String args[]) {

List<Integer> myList = Arrays.asList(10,15,8,49,25,98,98,32,15);

myList.stream()

.findFirst()

.ifPresent(System.out::println);

}

}

Output:

10

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q5 Given a list of integers, find the total number of elements present in the list using Stream functions?

import java.util.\*;

import java.util.stream.\*;

public class JavaHungry {

public static void main(String args[]) {

List<Integer> myList = Arrays.asList(10,15,8,49,25,98,98,32,15);

long count = myList.stream()

.count();

System.out.println(count);

}

}

Output:

9

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q6 Given a list of integers, find the maximum value element present in it using Stream functions?

import java.util.\*;

import java.util.stream.\*;

public class JavaHungry {

public static void main(String args[]) {

List<Integer> myList = Arrays.asList(10,15,8,49,25,98,98,32,15);

int max = myList.stream()

.max(Integer::compare)

.get();

System.out.println(max);

}

}

Output:

98

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q7 Given a String, find the first non-repeated character in it using Stream functions?

import java.util.\*;

import java.util.stream.\*;

import java.util.function.Function;

public class JavaHungry {

public static void main(String args[]) {

String input = "Java Hungry Blog Alive is Awesome";

Character result = input.chars() // Stream of String

.mapToObj(s -> Character.toLowerCase(Character.valueOf((char) s))) // First convert to Character object and then to lowercase

.collect(Collectors.groupingBy(Function.identity(), LinkedHashMap::new, Collectors.counting())) //Store the chars in map with count

.entrySet()

.stream()

.filter(entry -> entry.getValue() == 1L)

.map(entry -> entry.getKey())

.findFirst()

.get();

System.out.println(result);

}

}

Output:

j

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q8 Given a String, find the first repeated character in it using Stream functions?

import java.util.\*;

import java.util.stream.\*;

import java.util.function.Function;

public class JavaHungry {

public static void main(String args[]) {

String input = "Java Hungry Blog Alive is Awesome";

Character result = input.chars() // Stream of String

.mapToObj(s -> Character.toLowerCase(Character.valueOf((char) s))) // First convert to Character object and then to lowercase

.collect(Collectors.groupingBy(Function.identity(), LinkedHashMap::new, Collectors.counting())) //Store the chars in map with count

.entrySet()

.stream()

.filter(entry -> entry.getValue() > 1L)

.map(entry -> entry.getKey())

.findFirst()

.get();

System.out.println(result);

}

}

Output:

a

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q9 Given a list of integers, sort all the values present in it using Stream functions?

import java.util.\*;

import java.util.stream.\*;

import java.util.function.Function;

public class JavaHungry {

public static void main(String args[]) {

List<Integer> myList = Arrays.asList(10,15,8,49,25,98,98,32,15);

myList.stream()

.sorted()

.forEach(System.out::println);

}

}

Output:

8

10

15

15

25

32

49

98

98

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q10 Given a list of integers, sort all the values present in it in descending order using Stream functions?

import java.util.\*;

import java.util.stream.\*;

import java.util.function.Function;

public class JavaHungry {

public static void main(String args[]) {

List<Integer> myList = Arrays.asList(10,15,8,49,25,98,98,32,15);

myList.stream()

.sorted(Collections.reverseOrder())

.forEach(System.out::println);

}

}

Output:

98

98

49

32

25

15

15

10

8

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q11) Write a Java 8 program to concatenate two Streams?

import java.util.Arrays;

import java.util.List;

import java.util.stream.Stream;

public class Java8 {

public static void main(String[] args) {

List<String> list1 = Arrays.asList("Java", "8");

List<String> list2 = Arrays.asList("explained", "through", "programs");

Stream<String> concatStream = Stream.concat(list1.stream(), list2.stream());

// Concatenated the list1 and list2 by converting them into Stream

concatStream.forEach(str -> System.out.print(str + " "));

// Printed the Concatenated Stream

}

}

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q12 ) Write a Java 8 program to find the number of Strings in a list whose length is greater

than 5?

import java.util.ArrayList;

import java.util.List;

public class Java8 {

public static void main(String[] args) {

List<String> list = new ArrayList<String>();

list.add("Saket");

list.add("Saurav");

list.add("Softwaretestinghelp");

list.add("Steve");

// Added elements into the List

long count = list.stream().filter(str -> str.length() > 5).count();

/\* Converted the list into Stream and filtering out

the Strings whose length more than 5

and counted the length

\*/

System.out.println("We have " + count + " strings with length greater than 5"); }

}

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q13) Explain StringJoiner Class in Java 8? How can we achieve joining multiple Strings

using StringJoiner Class?

Answer: In Java 8, a new class was introduced in the package java.util which was known as

StringJoiner. Through this class, we can join multiple strings separated by delimiters along

with providing prefix and suffix to them.

In the below program, we will learn about joining multiple Strings using StringJoiner Class.

Here, we have “,” as the delimiter between two different strings. Then we have joined five

different strings by adding them with the help of the add() method. Finally, printed the

String Joiner.

import java.util.StringJoiner;

public class Java8 {

public static void main(String[] args) {

StringJoiner stj = new StringJoiner(",");

// Separated the elements with a comma in between.

stj.add("Saket");

stj.add("John");

stj.add("Franklin");

stj.add("Ricky");

stj.add("Trevor");

// Added elements into StringJoiner “stj”

System.out.println(stj);

}

}

output : Saket,John,Franklin,Ricky,Trevor

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q14)Find the Word/number got repeated by using Stream api

import java.util.Arrays;

import java.util.List;

import java.util.Map;

import java.util.function.Function;

import java.util.stream.Collectors;

public class Testing {

public static void main(String[] args) {

List<Integer> list1= Arrays.asList(2,432,467683,12,1,2,3,4,5,65,67,23);

List<String> list2=Arrays.asList("Devraj","Poojaree","Akash","Devraj");

Map<Integer, Long> e =list1.stream().collect(Collectors.groupingBy(Function.identity(),Collectors.counting()));

System.out.println(e);

list2.stream().collect(Collectors.groupingBy(Function.identity(),Collectors.counting())).forEach((a,b)->{

System.out.println(a+"="+b);

});

}

}

OutPut:

{432=1, 65=1, 1=1, 2=2, 67=1, 3=1, 4=1, 467683=1, 5=1, 23=1, 12=1}

Poojaree=1

Devraj=2

Akash=1

----------------------------------------------------------------------------------------------------------------------------------------

list i/p= ["Himanshu", "Devraj12", "rajeesh", "pavan", "sachine"]

Map output= {5=[pavan], 7=[rajeesh, sachine], 8=[Himanshu, Devraj12]}

public class Main {

public static void main(String[] args) {

List<String> list = Arrays.asList("Himanshu", "Devraj12", "rajeesh", "pavan", "sachine");

final Map<Integer, List<String>> lengthToWords = list.stream()

.collect(Collectors.groupingBy(String::length, TreeMap::new, Collectors.toList()));

System.out.println(lengthToWords);

}

}

----------------------------------------------------------------------------------------------------------------------------------------------------------

15) Write a Java Lambda Expression to Create Thread

Runnable withLambda = () -> System.out.println(" Runnable example with lambda exp.");

Thread thread1 = new Thread(withLambda);

thread1.start();

----------------------------------------------------------------------------------------------------------------------------------------------------------

17)Convert String to uppercase and Join them with coma

List<String> G7 = Arrays.asList("USA", "Japan", "France", "Germany", "Italy", "U.K.","Canada");

String G7Countries = G7.stream() .map(x -> x.toUpperCase()) .collect(Collectors.joining(", "));

----------------------------------------------------------------------------------------------------------------------------------------------------------

18)Given a list of Integer find square of all distinct numbers By using stream api

List<Integer> numbers = Arrays.asList(9, 10, 3, 4, 7, 3, 4);

List<Integer> distinct = numbers.stream() .map( i -> i\*i) .distinct() .collect(Collectors.toList());

----------------------------------------------------------------------------------------------------------------------------------------------------------

19)Given a List of Integers find total count, min, max, sum, and the average for numbers by using Stream api

input :list=[2, 3, 5, 7, 11, 13, 17, 19, 23, 29]

output: count=10, sum=129, min=2, average=12.900000, max=29

List<Integer> primes = Arrays.asList(2, 3, 5, 7, 11, 13, 17, 19, 23, 29);

IntSummaryStatistics status = primes.stream() .mapToInt((x) -> x) .summaryStatistics();

System.out.println(status);

---------------------------------------------------------------------------------------------------------------------------------------

20)

String a = "tELco";

StringBuffer b = new StringBuffer();

List<Character> li = a.chars().mapToObj(c -> (char) c).collect(Collectors.toList());

li.stream().forEach(a1 -> {

if (Character.isLowerCase(a1)) {

b.append(Character.toUpperCase(a1));

} else {

b.append(Character.toLowerCase(a1));

}

});

System.out.println(b);

-----------------------------------------------------------------------------------------------------------------------------------------

public class Filter\_1 {

public static void main(String[] args) {

List<String> locations = Arrays.asList(new String[]

{"Bidar\_1234","Bidar","Bangaluru","Chennai","Mumbai"});

Stream <String> stream= locations.stream();

List<String> result =stream.filter((location)-> location.length()>5).

distinct().sorted().collect(Collectors.toList());

System.out.println("............ printing each cities...............");

result.stream().forEach((city)-> System.out.println(city));

System.out.println("............. printing only first two cities................. ");

result.stream().limit(2).forEach(System.out::println);

System.out.println("............skepped elements ..............");

result.stream().skip(2).forEach(System.out::println);

long i =result.stream().count();

System.out.println("Number of counts : "+ i);

System.out.println(result);

System.out.println(locations);

}

}

---------------------------------------------------------------------------------------------------------------------------------------------------

final List<Employee> list = new ArrayList<>(Arrays.asList(new Employee(2, "Devraj"), new Employee(3, "Dev"),

new Employee(1, "raj"), new Employee(8, "Suresh")));

Map<Object, List<String>> map = list.stream().collect(

Collectors.groupingBy(e-> e.startsWith(e.getName()), Collectors.mapping(Employee::getName, Collectors.toList())));

OutPut:

{r=[raj], S=[Suresh], D=[Devraj, Dev]}

---------------------------------------------------------------------------------------------------------------------------------------------------

final List<Employee> list = new ArrayList<>(Arrays.asList(new Employee(2, "Devraj"), new Employee(3, "Dev"),

new Employee(1, "raj"), new Employee(8, "Suresh")));

Map<Character, List<Employee>> m1 = list.stream()

.collect(Collectors.groupingBy(a -> a.startsWith(a.getName())));

OutPut:

{r=[Employee [id=1, name=raj]], S=[Employee [id=8, name=Suresh]], D=[Employee [id=2, name=Devraj], Employee [id=3, name=Dev]]}

------------------------------------------------------------------------------------------------------------- Sorting the collections

public class Student {

private int age;

private String name;

public int getAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public Student(int age, String name) {

super();

this.age = age;

this.name = name;

}

@Override

public String toString() {

return "Student [age=" + age + ", name=" + name + "]";

}

public Student () {}

// @Override

// public int compareTo(Student o) {

// if(age>o.getAge())

// {

// return -1;

// }

// return 1;

// }

}

List<Student> list = new ArrayList<Student>(

Arrays.asList(new Student(23, "Ram"), new Student(22, "SaiRam"), new Student(21, "SitaRam")));

List<Student> list1 = list.stream().sorted().collect(Collectors.toList());

list1.forEach(System.out::println);

---------------------------------------------------------------------------------------------------

public class REDUCE\_1 {

public static void main(String[] args) {

List<Integer> intList= Arrays.asList(5,7,13,9,-1);

Optional<Integer> result = intList.stream().filter((a)-> a>0).reduce((a,b)->a>b?a:b);

if(result.isPresent())

{

System.out.println("Result : "+result.get());

}

}

}

-----------------------------------------------------------------------------------------------------------

package com.xyz;

public class Department {

private int id;

private String dName;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getdName() {

return dName;

}

public void setdName(String dName) {

this.dName = dName;

}

public Department(int id, String dName) {

super();

this.id = id;

this.dName = dName;

}

public Department() {

super();

}

@Override

public String toString() {

return "Department [id=" + id + ", dName=" + dName + "]";

}

public Character getChar(String str) {

return str.charAt(0);

}

}

package com.xyz;

import java.util.List;

import java.util.Objects;

public class Employee1 {

private int eid;

private String ename;

private String address;

private Double salary;

private List<Department> departments;

public int getEid() {

return eid;

}

public void setEid(int eid) {

this.eid = eid;

}

public String getEname() {

return ename;

}

public void setEname(String ename) {

this.ename = ename;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public Double getSalary() {

return salary;

}

public void setSalary(Double salary) {

this.salary = salary;

}

public List<Department> getDepartments() {

return departments;

}

public void setDepartments(List<Department> departments) {

this.departments = departments;

}

public Employee1(int eid, String ename, String address, Double salary, List<Department> departments) {

super();

this.eid = eid;

this.ename = ename;

this.address = address;

this.salary = salary;

this.departments = departments;

}

public Employee1() {

super();

}

@Override

public String toString() {

return "Employee1 [eid=" + eid + ", ename=" + ename + ", address=" + address + ", salary=" + salary

+ ", departments=" + departments + "]";

}

}

List<Department> dList = new ArrayList<>(Arrays.asList(new Department(2, "ECE"), new Department(5, "CSE"),

new Department(6, "Civil"), new Department(8, "ELE")));

int arr[] = { 3, 4, 4, 3, 2 };

List<Integer> li1 = Arrays.stream(arr).mapToObj(a -> a).toList();

System.out.println(li1);

String str = "DevrajDevraj";

List<Character> ce = str.chars().mapToObj(a -> (char) a).collect(Collectors.toList());

Map<Character, Long> map = ce.stream()

.collect(Collectors.groupingBy(Function.identity(), Collectors.counting()));

System.out.println(map);

List<Employee1> eList = new ArrayList<>(Arrays.asList(new Employee1(34, "Devraj", "xyz", 3435.345, dList)));

Map<Object, List<Department>> li = eList.stream().flatMap(a -> a.getDepartments().stream())

.collect(Collectors.groupingBy(d -> d.getChar(d.getdName())));

System.out.println(li);

-------------------------------------------------------------------------------------------Map Interface --------------------------------------

Map<String, Integer> map = new HashMap<>();

map.put("apple", 5);

map.put("banana", 3);

map.put("cherry", 7);

map.put("date", 5);

Map<Integer, List<String>> groupedMap = map.entrySet().stream()

.collect(Collectors.groupingBy(Map.Entry::getValue,

Collectors.mapping(Map.Entry::getKey, Collectors.toList())));

System.out.println(groupedMap);

groupedMap.entrySet().stream()

.forEach(entry -> System.out.println("Value: " + entry.getKey() + ", Keys: " + entry.getValue()));

output:

{3=[banana], 5=[date, apple], 7=[cherry]}