**Lambda expressions**

**Aditya Yadav**

**1. Write an application to perform basic arithmetic operations like add, subtract, multiply & divide. You need to define a functional interface first.**

**Description:-**

Define Functional Interface and write a program to perform arithmetic operations like add, subtract, multiply and divide using functional interface.

**Examples:**

**Input:-**13       5

**Output:-**

18.0      //Addition of 13 and 5

                                       8.0        //Subtraction of 13 and 5

                                       65.0      // Multiplication of 13 and 5

                                       2.6        //Division of 13 and 5

**FunctionalInterface:-**

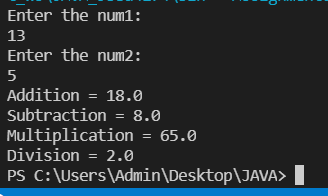
A functional interface is an interface that contains only one abstract method. They can have only one functionality to exhibit. From Java 8 onwards, lambda expressions can be used to represent the instance of a functional interface. A functional interface can have any number of default methods. Runnable, ActionListener, Comparable are some of the examples of functional interfaces.

**Specifications:**

public class Assignment4Q1 {  
    public double addition(int num1,int num2){}  
    public double subtraction(int num1,int num2){}  
    public double multiplication(int num1,int num2){}  
    public double division(int num1,int num2){}  
    public static void main(String[] args) {}  
}







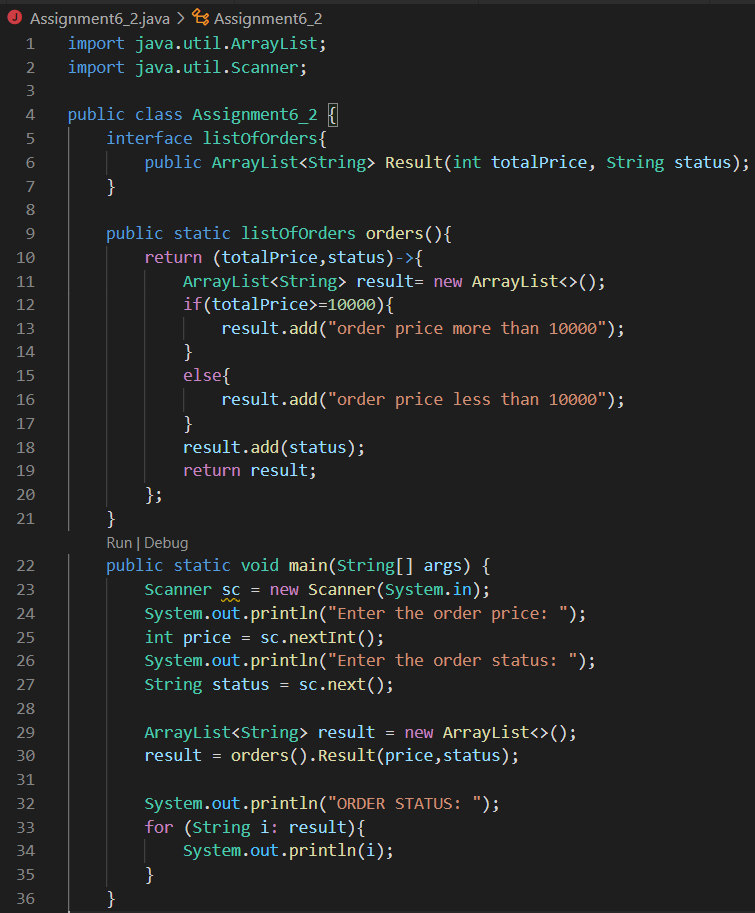
**Q2. Write an application using lambda expressions to print Orders having 2 criteria implemented: 1) order price more than 10000 2) order status is ACCEPTED or COMPLETED.**

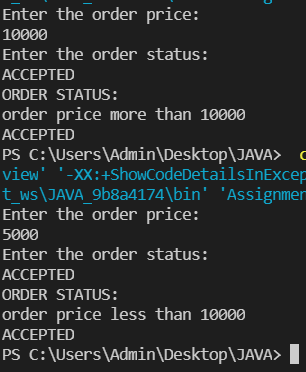
**Description:**

Write a program in such a way that it has a method which returns the list of orders satisfying the 2 conditions mentioned in the question.

**Specifications:**

import java.util.ArrayList;  
  
public class Assignment4Q2 {  
  
    private int totalPrice;  
    private String status;  
  
    public static ArrayList<Assignment4Q2> listOfOrders(ArrayList<Assignment4Q2> orders) {}  
    public static void main(String[] args) {}  
}





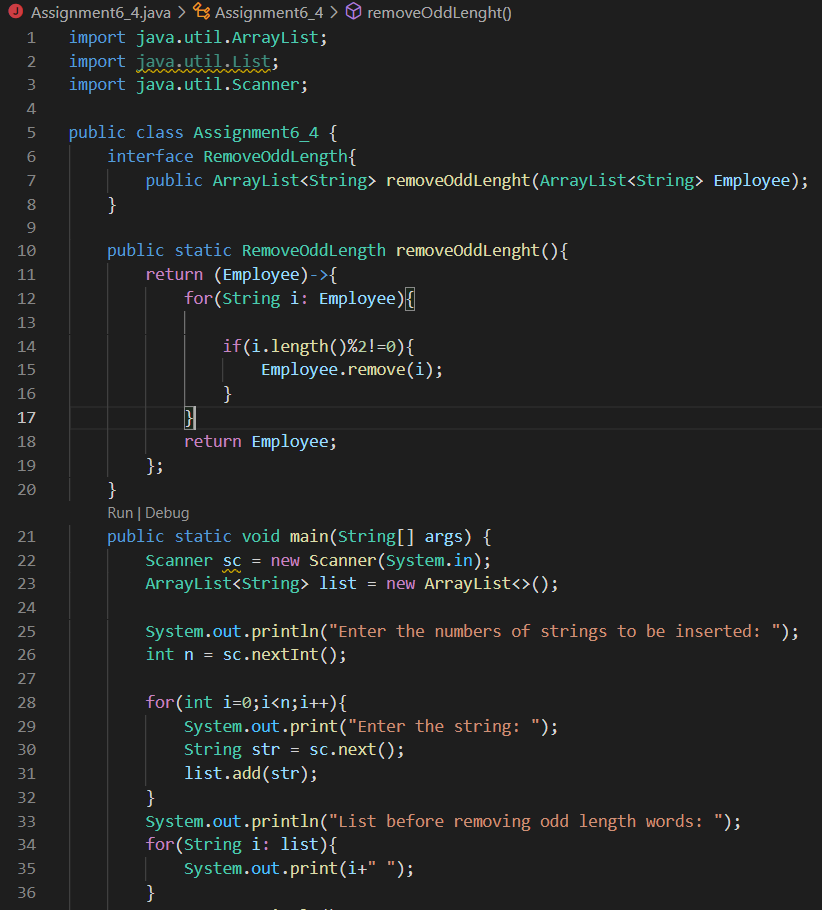
**Q4. Remove the words that have odd lengths from the list. HINT: Use one of the new methods from JDK 8. Use removeIf() method from Collection interface.**

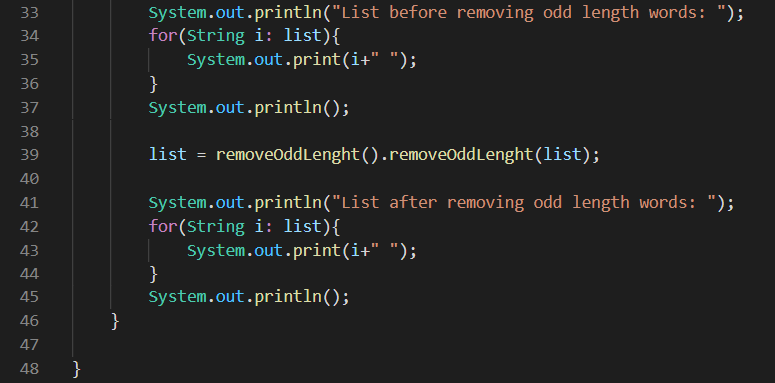
**Description:-**

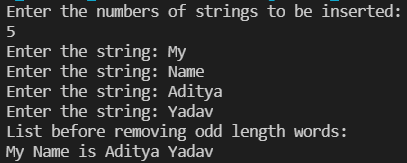
Write a program using java 8 features which can remove the odd length words from the list.

**Specifications:**

public class Assignment4Q4 {  
    public ArrayList<String> removeOddLength(ArrayList<String> employeeList){}  
    public static void main(String[] args) { }  
}







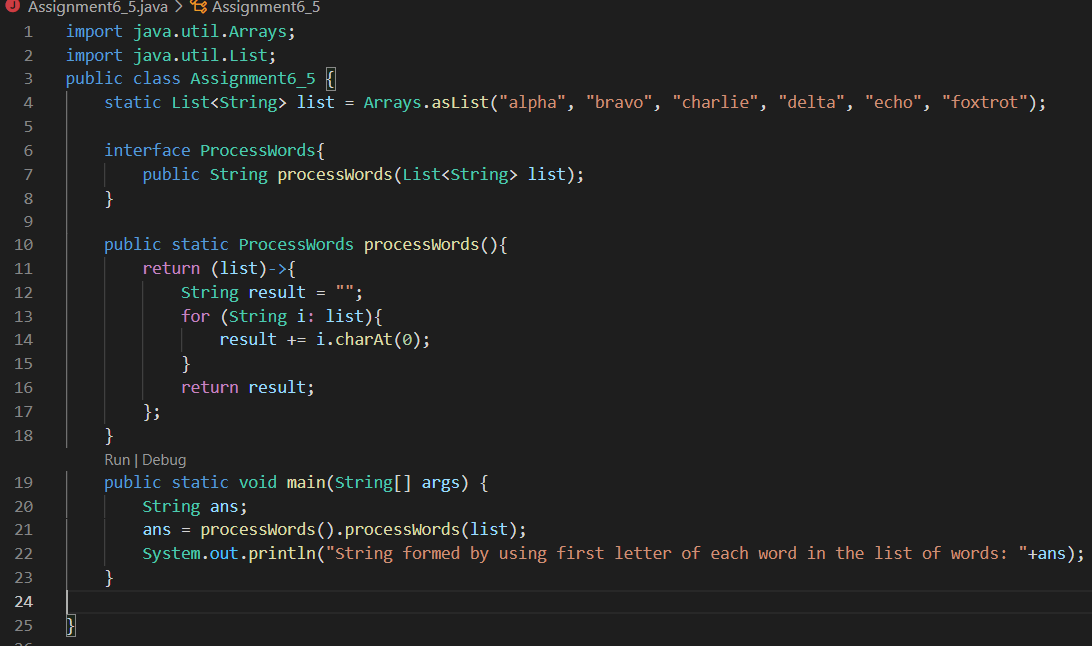
**Q5. Create a string that consists of the first letter of each word in the list of Strings provided. HINT: Use Consumer interface & a String Builder to construct the result.**

**Description:**

Write a java program using StringBuilder and Consumer interface which will return a string. The returned string should consistes of the first let of each word in the list of words.

**Specifications:**

public class Assignment4Q5 {  
    List<String> list = Arrays.*asList*("alpha", "bravo", "charlie", "delta", "echo", "foxtrot");  
  
    public static void main(String[] args) { }  
  
    public static String processWords(List<String> list) {}  
}



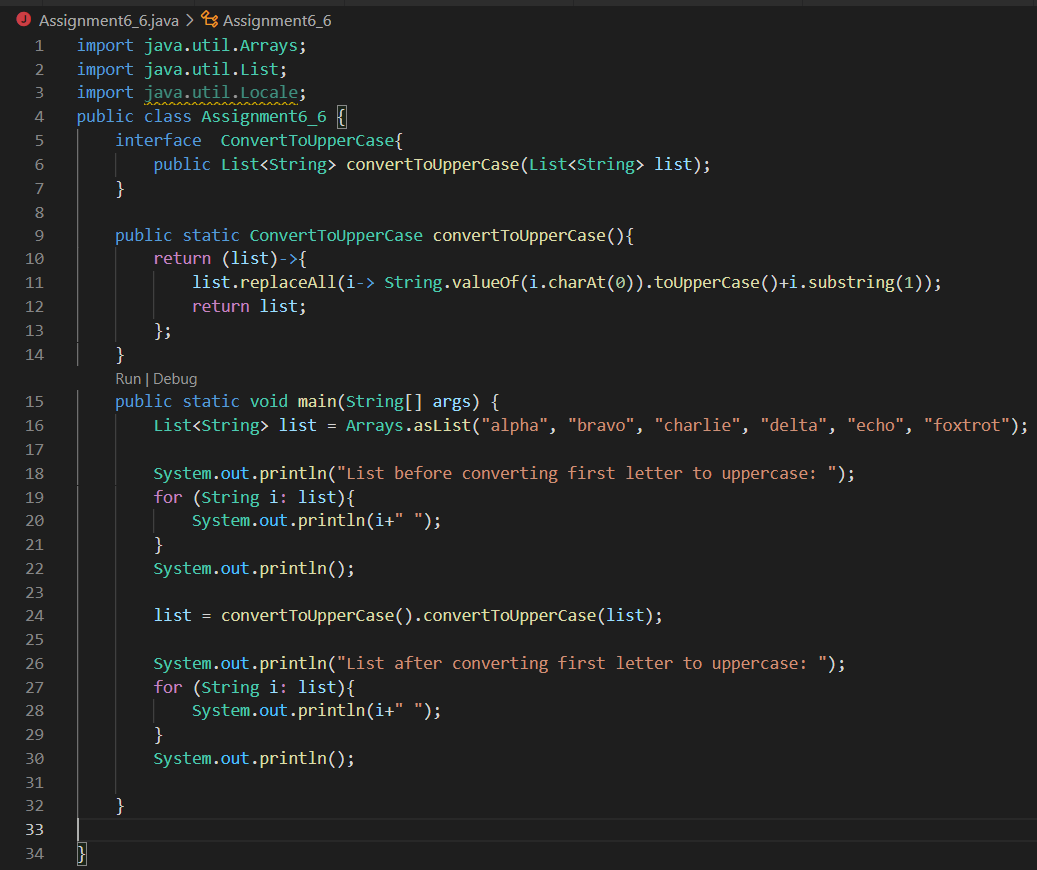


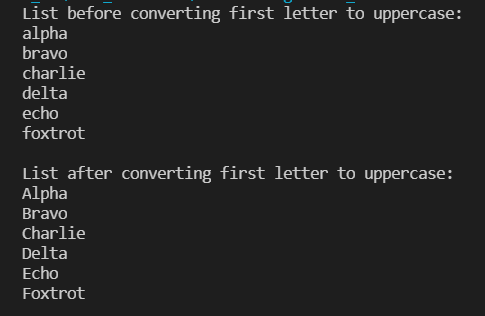
**6. Replace every word in the list with its upper case equivalent. Use replaceAll() method & Unary Operator interface.**

Using replaceAll() method and Unary Operator interface write a java program which replaces evry word in the list with its upper case equivalent.

**Specifications:**

public class Assignment4Q6 {  
    public static void main(String[] args) {}  
    public List<String> convertToUpperCase(List<String> list) {}    
}





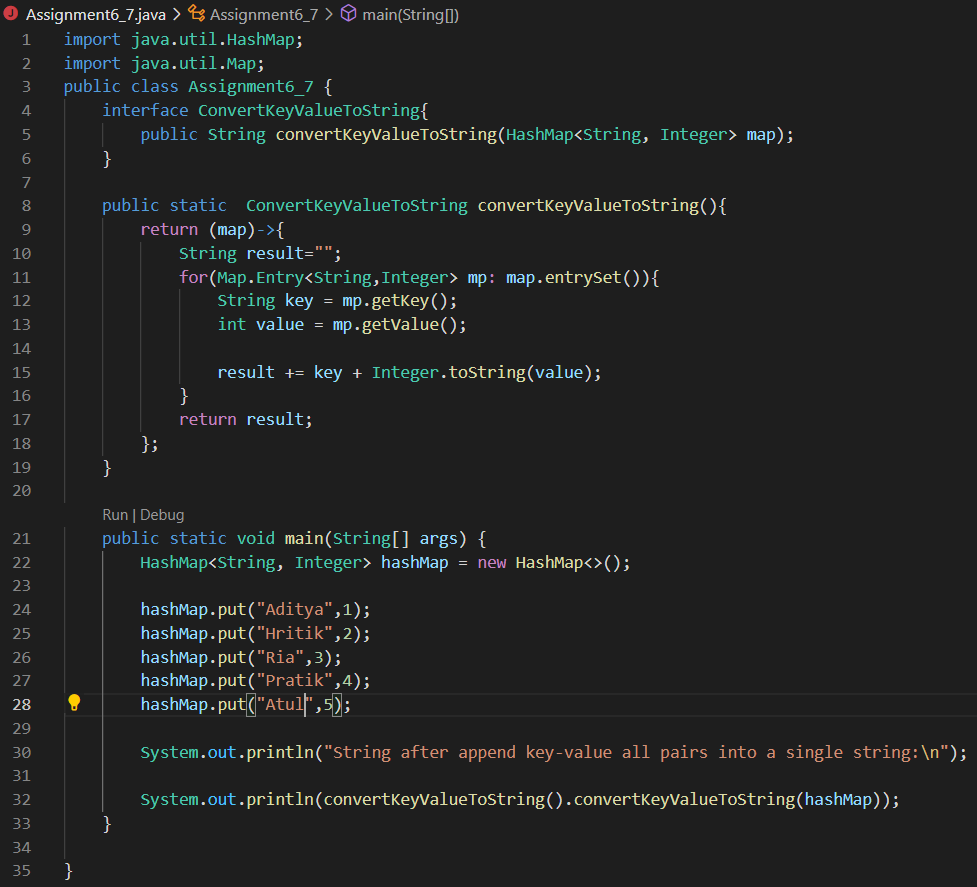
**7. Convert every key-value pair of the map into a string and append them all into a single string, in iteration order. HINT: Use Map.entrySet() method & a StringBuilder to construct the result String.**

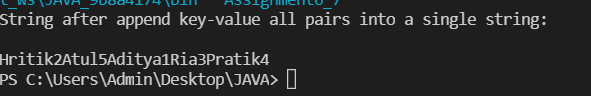
**Description:-**

Write a java program using Map.entrySet() method & a StringBuilder which will return a string by appending all the key value pairs of a map into a single string ,in insertion order.

**Specifications:**

public class Assignment4Q7 {  
    public static void main(String[] args) {}  
    public String convertKeyValueToString(HashMap<String, Integer> map) {}  
}





**Q8. Create a new thread that prints the numbers from the list. Use class Thread & interface Consumer.**

**Description:-**

Write a java program which will print the list of number using Thread and interface Consumer.

**Specifications:**

public class Assignment4Q8 {}



