EEM 480 Homework 2 - Halil İbrahim ÖZTÜRK

A JAVA program using object orienting design technique of name Absurd and package is halil_ıbrahim_ozturk_hw2.

This program purpose learning and applications some string functions and read/write txt. Files.

It get commands from the user as a some letters and give user results.

Here the user can use the program by giving same commans as below:

The commands are :

s or S The user can enter the string
d/D n the n'th character is deleted from the string and the result is shown
p/P Program check whetheer the sring is palindrome or not
r/R The string will be reversed and result is shown
i/I n <String> The <String> will be inserted after n'th position of the string and the result is shown
m/M The string will be capitilized and result is shown
o/O output string to screen
t/T<Pathname\FileName> The string is read from the file
w/W<Pathname\FileName> The string is written to the file
f/F Program reverse the words and result is shown
x/X Terminates the program

When I do my this homework , I face to many compile errors and lack of information.

At this time , Internet was my best supplier.

I used private generally because ı did not public because of easily getting datas without any problem and there is no other script.

I tried complete my program without main class and ı just write Absurd(); in the main class.

My classes ;

Absurd()

Delete(int k)

Exit()

GetString()

InFile()

Insert(String n, int x)

MakeCapitilize()

OutFile()

OutReverse()

Output()

Palindrome()(boolean)

WordReverse()

Main(String[]args)

My variables ;

Command: String // user can entered this commands

İndex : int // index of string user entered

İnput : String // all of user entered

k : // index number of string user entered

n : String // inserted string

sc : Scanner // scanner object

str : String // string get by s command from user

x : int // index of string user entered

Example:

```
>>s Here is the String
>>Here is the String
>>d 4
>>Her is the String
>>p
>> The String is not a palindrome
>>r
>>gnirtS eht si reH
>> i 5 e
>>gnirteS eht si reH
>>o
>> gnirteS eht si reH
>>m
>>GNIRTES EHT SI REH
>>r
>>HER IS THE SETRING
>>f
>>SETRING THE IS HER
>>t c:\Myfile.txt
>> The file has been opened and the string has been read
>>w c:\Yourfile.txt
>> The file has been opened and the string has been written
>>x
C:\>The Program has terminated
```

MY CODE

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package halil_ıbrahim_ozturk_hw2;
import java.util.*; //import script for scanner functions
import java.lang.*;// Java program to ReverseString using ByteArray.and // Java program to insert a string into another string
// without using any pre-defined method
import java.io.*;// Java program to ReverseString using ByteArray.
import java.util.*;// Java program to ReverseString using ByteArray.
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.File;  // Import the File class
import java.io.FileNotFoundException;  // Import this class to handle errors
import java.util.Scanner; // Import the Scanner class to read text files


import java.util.regex.Pattern; // Java Program to reverse a String
// without using inbuilt String function
/**
 *
 * @author halilibrahim
 */
public class Halil_Ibrahim_Ozturk_HW2 {

   /**
    * @param args the command line arguments
    */
   private static String command;//definition of command
   private static String input;//definition string from user
   private static String str;//here is the string
   private static int k; // deleted number
   private static Scanner sc= new Scanner(System.in); //System.in is a standard input stream
   private static String n ;// index number for Insert()
   private static int x ; // word for Insert()
   private static int index;
   public static void main(String[] args) {



       // TODO code application logic here


       Absurd();
   }


   private static void Absurd(){
       System.out.print( "\n" + "Enter a command: ");
       input= sc.nextLine(); //getting something from user
       command  =input.substring(0, 1);//get first character as a command
       if("s".equalsIgnoreCase(command)) GetString();//go to Get String
       if("d".equalsIgnoreCase(command)) Delete(k);//delete the k'th letter in string
       if("p".equalsIgnoreCase(command)) Palindrome();//test palindrome
       if("r".equalsIgnoreCase(command)) OutReverse();//reversed string
       if("i".equalsIgnoreCase(command)) Insert(n,x);//insert string
       if("m".equalsIgnoreCase(command)) MakeCapitilize();//The string will be capitilized and result is
shown
       if("o".equalsIgnoreCase(command)) Output();//put the string into the output stream out
       if("t".equalsIgnoreCase(command)) InFile();//The file has been opened and the string has been
written
       if("w".equalsIgnoreCase(command)) OutFile();//put the string into the output stream outas a file
       if("f".equalsIgnoreCase(command)) WordReverse();
       if("x".equalsIgnoreCase(command)) Exit();//The file has been opened and the string has been
written
       else{
           System.out.print( "\n" + "Please enter a valid command ");
           Absurd();
       }
   }

   private static void GetString(){        //user can enter his/her string
     //number of characters in this string
     str= input.substring(2);//deleted command from the string
     System.out.print(str);//write the string
     Absurd();

   }

   private static void Delete(int k){    //delete the k'th letter in string
   k=Integer.parseInt(input.replaceAll("[\\D]", ""));//get int from string
   str= str.substring(0,k-1)+str.substring(k);//delete string
   System.out.print(str);//write the string
   Absurd();
   }

   private static boolean Palindrome(){
       StringBuilder sb= new StringBuilder(str);
       sb.reverse(); //get reversed form
       String rev=sb.toString();
       if(str.equals(rev)){  //palindrome check
       System.out.print("Yes, this is palindrome");
       Absurd();
       return true;  //boolean is true




   }
       else{
       System.out.print("No, this is not palindrome");
       Absurd();

       return false;  //boolean is false
   }


   }

    private static void OutReverse(){// // reverse the string and put the new reversed form to screen using
//PrintString function

       // getBytes() method to convert string
       // into bytes[].
       byte[] strAsByteArray = str.getBytes();

       byte[] result = new byte[strAsByteArray.length];

       // Store result in reverse order into the
       // result byte[]
       for (int i = 0; i < strAsByteArray.length; i++)
           result[i] = strAsByteArray[strAsByteArray.length - i - 1];
       str= new String(result);
       System.out.print(str);
```

```
      Absurd();
   }

   private static void Insert(String n,int x){
      String index=input.substring(2, 3);
      x=Integer.parseInt(index.replaceAll("[\\D]", ""));//make x string to int
      n= input.substring(4);//selected word inserted

      str = str.substring(0, x) + n + str.substring(x);
      System.out.print(str);
      Absurd();

   }

   private static void MakeCapitilize(){//change every letter in the string to its capitilized form

      str = str.toUpperCase();//basic function for uppercase
      System.out.print(str);
      Absurd();

   }

   private static void Output(){
      try (PrintWriter p = new PrintWriter(new FileOutputStream("Yourfile.txt", true))) {//fileoutput
stream to print output
   p.println(str);
    } catch (FileNotFoundException e1) {
   e1.printStackTrace();
     }
      System.out.print(str);
      Absurd();
```

```
}

private static void OutFile(){
    input.substring(2);
   FileOutputStream fos = null;
  File file;
  try {
      //Specify the file path here
            file = new File(input.substring(2));
            fos = new FileOutputStream(file);

      /* This logic will check whether the file
             * exists or not. If the file is not found
             * at the specified location it would create
             * a new file*/
            if (!file.exists()) {
               file.createNewFile();
            }

            /*String content cannot be directly written into
             * a file. It needs to be converted into bytes
             */
            byte[] bytesArray = str.getBytes();

            fos.write(bytesArray);
            fos.flush();
            System.out.println("File Written Successfully");
   }
   catch (IOException ioe) {
            ioe.printStackTrace();
   }
  finally {
         try {
           if (fos != null)
           {
                   fos.close();
           }
    }
         catch (IOException ioe) {
           System.out.println("Error in closing the Stream");
         }
  }
 Absurd();
}
private static void InFile(){
   //scanner functions to scan the txt file
   //as a string
   try {
 File myObj = new File(input.substring(2));//file path here
 Scanner myReader = new Scanner(myObj);
 while (myReader.hasNextLine()) {
   String data = myReader.nextLine();
   System.out.println(data);
 }
 myReader.close();
} catch (FileNotFoundException e) {
 System.out.println("An error occurred.");
 e.printStackTrace();
}
       Absurd();


}
```

```
    private static void WordReverse(){



String[] split = str.split(" ");

String result = "";

for (int i = split.length - 1; i >= 0; i--) {

  result += (split[i] + " ");

}

str=result.trim();

System.out.print(str);

Absurd();


    }

    private static void Exit(){

        System.exit(0);


    }


}
```