# **Homework**

# **Grade 11 Review** 7 **– File Output and Input**

1. **StarTable.java**: Ask users for the dimension a table. Create a 2D array of characters with the specified dimensions, and initialize each element to be ‘\*’. Then print out the content of the array as a table to the file “star.txt”

| /\*  \* Program name: StarTable.java  \*  \* By: Lucas Chow (Last edited: 2022-09-30)  \*  \* ICS4U1 - 04\_Gr11Review  \*  \* This program ask the user or dimensions for a table, creating a 2d array of the characters  \* with dimensions given by the user. Each element in the 2d array will be  \* the character '\*', printing out the content as a table to the .txt file of star.txt  \*  \*  \*/    import java.io.\*;  import java.util.\*;  public class StarTable  {  public static void main(String[] args)  {  //initializing variables and objects  String fileName = "star.txt";  int starArrLength;  int starArrWidth;  try  {  //creating BufferedWriter  BufferedWriter out = new BufferedWriter(new FileWriter(fileName,false));  Scanner sc = new Scanner(System.in);    //prompting the user for the length of the star array  System.out.print("Enter the length of the star array: ");  starArrLength = sc.nextInt();  sc.nextLine();    //prompting the user for the width of the star array  System.out.print("Enter the width of the star array: ");  starArrWidth = sc.nextInt();  sc.nextLine();    //creating the array  String[][] starArr = new String[starArrWidth][starArrLength];      //looping through the array initializing all the values as '\*'  for (int i = 0; i < starArrWidth; i++)  {  for (int a = 0; a < starArrLength; a++)  {  starArr[i][a] = "\*";  }  }      //looping thorugh the array again, printing out all the values  // > This could have been done in one array, but this can be more understandable  for (int i = 0; i < starArrWidth; i++)  {  for (int a = 0; a < starArrLength; a++)  {  out.write(starArr[i][a]+" ");  }  out.write("\n");  }    //confirmation message  System.out.println("The file "+fileName+" has successfully been printed!");    //closing the bufferedwriter and scanner  out.close();  sc.close();  }    //catching file error exception  catch (IOException e)  {  //printing out the error  System.out.println(e + " Problem reading " + fileName);  }    }  } |
| --- |
|  |

1. **TwoPowerTable.java**: Write a program that creates a text file that contains a power of two table in HTML format. The size (maximum power) of the table should be inputted by user. The file could be called "twoPowerTable.html". When it is viewed with a browser you will see something like:

| **Exponent** | **Power of 2** |
| --- | --- |
| 0 | 1 |
| 1 | 2 |
| 2 | 4 |
| 3 | 8 |
| 4 | 16 |
| 5 | 32 |
| 6 | 64 |
| 7 | 128 |
| 8 | 256 |
| 9 | 512 |
| 10 | 1024 |

The file should start with something like:

<html><head>

<title>Powers of Two</title>

</head>

<body>

<table border cellpadding=5>

<tr><th>Power of 2</th><th>Value</th></tr>

And end with:

</table>

</body></html>

Each line of the table looks like:

<tr><td>0</td><td>1</td></tr>



| /\*  \* Program name: TwoPowerTable.java  \*  \* By: Lucas Chow (Last edited: 2022-09-30)  \*  \* ICS4U1 - 04\_Gr11Review  \*  \* This program prompts the user for the 'n' maximum power of two 2^x <= n,  \* and outputs an html file called "twoPowerTable.html" in html. This program  \* uses BufferedReader(new InputStreamReader(System.in)) and  \* BufferedWriter(new FileWriter(filename, false)) to take in user input  \* and output the html file respectively.  \*  \*/  //importing java.io.\* to use BufferedReader & BufferedWriter  import java.io.\*;  public class TwoPowerTable {    //start of main method  public static void main(String[] args)  {  try  {  //declaring variables and objects  //declaring the filename variable  String filename = "twoPowerTable.html";  int maximumPower;  //BufferedReader for user input  BufferedReader sc = new BufferedReader(new InputStreamReader(System.in));  //BufferedWriter for output to html file  BufferedWriter bw = new BufferedWriter(new FileWriter(filename, false));    System.out.print("Enter size/maximum power of the table: ");  maximumPower = Integer.parseInt(sc.readLine());    //start of the html coding  bw.write("<html><head>\n");  bw.write("<title>Powers of Two</title>\n");  bw.write("</head>\n");  bw.write("<body>\n");  bw.write("<table border cellpadding=5>\n");  bw.write("<tr><th>Power of 2</th><th>Value</th></tr>\n");    //printing out the powers of two in html format  for (int i = 0; Math.pow(2, i) <= maximumPower; i++ )  {  bw.write("<tr><td>"+i+"</td><td>"+Math.pow(2,i)+"</td></tr>");  }    //ending of the html code  bw.write("</table>");  bw.write("</body></html>");    //confirmation message  System.out.println("File exported succesfuly!");    //closing bufferedReader of inputStreamReader and Bufferedwriter  sc.close();  bw.close();  }  //if IOException from the BufferedWriter  catch (IOException e)  {  System.out.println("Error: "+e);  }  }  } |
| --- |
|  |

1. **AddNumbers.java**: Write a program that reads the number on each line in a file “numbers.txt”, then outputs the sum to the standard output. Test your program by creating a text file “numbers.txt” that contains a list of numbers (any type) on separate lines.

| /\*  \* Program name: AddNumbers.java  \*  \* By: Lucas Chow (Last edited: 2022-09-30)  \*  \* ICS4U1 - 04\_Gr11Review  \*  \* \*\*This program requires a file called "numbers.txt" filled with numbers  \*  \* This program uses BufferedReader to read the file "numbers.txt", finding the sum  \* of all the numbers in the file, and outputting the value  \*  \*/  //importing BufferedWriter from java.io.\*;  import java.io.\*;  public class AddNumbers {  //start of main method  public static void main(String[] args)  {  //declaring and initializing filename  String filename = "numbers.txt";    //declaring variables  String inputStream;  double sumOfNumbers;  try  {  //creating object BufferedReader, using fileReader, on the filename: "numbers.txt"  BufferedReader br = new BufferedReader(new FileReader(filename));    //initializing sumOfnumbers to 0  sumOfNumbers = 0;    //getting theinput form the text  do  {  //taking in text from the file  inputStream = br.readLine();    //checking if value is not blank (null)  if (inputStream != null)  {  //implying if not a blank line; parses to double and adds to total  sumOfNumbers += Double.parseDouble(inputStream);  }    //repeats until the program runs into a blank line  } while (inputStream != null);    //outputting result  System.out.println("the sum of the digits in file: \"" + filename + "\" is " + sumOfNumbers);    //closing BufferedReader br  br.close();    //catching error  } catch (IOException e)  {  //error message  System.out.println("Error: "+ e + " with file "+ filename);  }  }  } |
| --- |
|  |

1. **CountCases.java**: Write a program that reads in the file “passage.txt”, then outputs the number of upper and lower case letters in the content. Any non-alphabet character should be ignored. Test your program by creating a text file “passage.txt” that contains any random text.

| /\*  \* Program name: CountCases.java  \*  \* By: Lucas Chow (Last edited: 2022-09-30)  \*  \* ICS4U1 - 04\_Gr11Review  \*  \* This program reads the file "passage.txt" and outputs how many uppercase  \* and lowercase letters there are, ignoring any non-alphabetical characters  \*  \*/  //importing BufferedReader  import java.io.\*;  public class CountCases {    //start of the main method    public static void main(String[] args)  {    String filename = "passage.txt";  String inputStream;  int upperCaseCount;  int lowerCaseCount;  try  {  //creating BufferedReader object  BufferedReader br = new BufferedReader(new FileReader(filename));      //initializing upper and lower case count to 0  upperCaseCount = 0;  lowerCaseCount = 0;  do  {  inputStream = br.readLine();  if (inputStream != null)  {    //loops through the string, checking the ASCII value of each character to determine casing  for (int i = 0; i < inputStream.length(); i++)  {  //if uppercase (A -> Z)  if (inputStream.charAt(i) >= 65 && inputStream.charAt(i) <= 90)  {  //adding one to the number of upper case characters found  upperCaseCount++;  }    //if lowercase (a -> z)  else if (inputStream.charAt(i) >= 97 && inputStream.charAt(i) <= 122)  {  //adding one to the number of lower case characters found  lowerCaseCount++;  }  }  }    //ends when reaching a blank line  } while (inputStream != null);    //outputting the # of upper case characters and lower case characters in the file "passage.txt"  System.out.println("Number of upper case letters in the file: " + upperCaseCount);  System.out.println("Number of lower case letters in the file: " + lowerCaseCount);          //closing bufferedReader  br.close();    //catching IOException error thrown by BufferedReader  } catch (IOException e)  {  //outputting error message  System.out.println("Error "+ e + " reading file " + filename);  }  }  } |
| --- |
|  |