

IMPORTING java.util.Scanner;

IMPORTING java.io.\*;

DEFINE class LibraryCatalogue

DEFINE static int stringLength;

DEFINE public static boolean isString(String string) {

INITIALIZE boolean isEmpty = false;

INITIALIZE (int i = 0; i <= string.length(); i++) {

stringLength = string.replaceAll(" ", "").length();

INITIALIZE if (stringLength > 0) {

isEmpty = true;

} else {

isEmpty = false;}

} RETURNING isEmpty; }

DEFINE public static boolean isNumber(String number) {

INITIALIZE for (int i = 0; i <= number.length(); i++) {

stringLength = number.replaceAll(" ", "").length(); }

INITIALIZE try {

Float.parseFloat(number);

return true;

} catch (NumberFormatException e) {

return false; }

}

DEFINE class main

DEFINE new scanner scan

PRINT("Enter name of the data file to read : ");

DEFINE String filename=scan.nextLine();

DEFINE class main

DEFINE new scanner scan

PRINT("Enter name of the data file to read : ");

DEFINE String filename=scan.nextLine();

PRINT("Enter name of the data file to read : ");

DEFINE String filename=scan.nextLine();

DEFINE File fileobject;

DEFINE Scanner filereader=null;

DEFINE String fileline;

DEFINE int invalidBooks=0;

DEFINE String delimiter= null;

DEFINE int count=0;

DEFINE int authorCount= 0;

DEFINE int wrong\_count=0;

DEFINE int invalidcount=0;

DEFINE String[] wrong\_delimiter = {";", "\*", "+", "/","!","@","#","$","%","^","&"};

try to try and catch the error {DEFINE fileobject = new File(filename+".txt");

DEFINE filereader = new Scanner(fileobject);

PRINT ("File found.\n");

}

catch (FileNotFoundException e) to catch the error

{

PRINT ("File not found!!!!!!!!!!!!");

}

DEFINE String temp;

DEFINE int index=0;

DEFINE String[] title=new String[20];

DEFINE String[] author=new String[20];

DEFINE String[] publisher=new String[20];

DEFINE float[] price=new float[20];

DEFINE int[] pages=new int[20];

DEFINE String[] ISBN=new String[20];

PRINT("-----Error in file text-----");

INTIALIZE while(filereader.hasNext())

{

fileline = filereader.nextLine();

String[] splitline=fileline.split("-");

title[index]=splitline[0].trim();

author[index]=splitline[1].trim();

publisher[index]=splitline[2].trim();

temp=splitline[3].trim();

price[index]=Float.*parseFloat*(temp);

temp=splitline[4].trim();

pages[index]=Integer.*parseInt*(temp);

ISBN[index]=splitline[5].trim();

INITIALIZE try {

INITIALIZE for (int x=0; x<wrong\_delimiter.length;x++)

{

if (fileline.contains(wrong\_delimiter[x]) && (arraylength <6)) {

wrong\_count = wrong\_count + 1;

delimiter= wrong\_delimiter[x];

}

}

WHEN if (wrong\_count > 0 ) {

PRINT ("The invalid field delimiter is used: "+ delimiter);

invalidBooks++;

continue;

}

OR else if (arraylength<6) {

PRINT ("The field delimiter may be missing.");

invalidBooks++;

continue;

}

INITIALIZE if (!isString(splitline[0])) {

PRINT ("Book title may be missing.");

COUNT invalidcount++; }

INITIALIZE if (!isString(splitline[1])) {

PRINT ("Book author may be missing.");

COUNT invalidcount++;}

INITIALIZE if (!isString(splitline[2])) {

PRINT ("Book publisher may be missing.");

COUNT invalidcount++;}

INITIALIZE if (!isString(splitline[5])) {

PRINT ("ISBN may be missing.");

COUNT invalidcount++;

} }

INITIALIZE if (!isNumber(splitline[3])) {

PRINT ("Price may be missing.");

COUNT invalidcount++; }

INITIALIZE if (!isNumber(splitline[3])) {

PRINT ("Page may be missing.");

COUNT invalidcount++; }

INITIALIZE if (invalidcount > 0) {

invalidBooks++;}

}catch(NumberFormatException e) {}

Index= Index +1;

}

PRINT ("Number of invalid books: "+ invalidBooks);

PRINT ("---------Book Details---------\n");

INITIALIZR for LOOP (int j=0;j<index;j++)

{

PRINT ("Title: "+title[j]);

PRINT ("Author: "+author[j]);

PRINT ("Publisher: "+publisher[j]);

PRINT ("Price: "+price[j]);

PRINT ("Pages: "+pages[j]);

PRINT ("ISBN: "+ISBN[j]);

}

DEINE String authorname;

PRINT ("Enter name of author for report of book by that author : ");

SCAN NEXTLINE TO GET authorname

PRINT ("Books Written by "+authorname+" are : ");

INITIALIZE for(int k=0; k<index; k++)

{

CREATE if CONDITION ((authorname).equals(author[k]))

{

PRINT ("Title: "+title[j]);

PRINT ("Author: "+author[j]);

PRINT ("Publisher: "+publisher[j]);

PRINT ("Price: "+price[j]);

PRINT ("Pages: "+pages[j]);

PRINT ("ISBN"+ISBN[j]);

}

else {

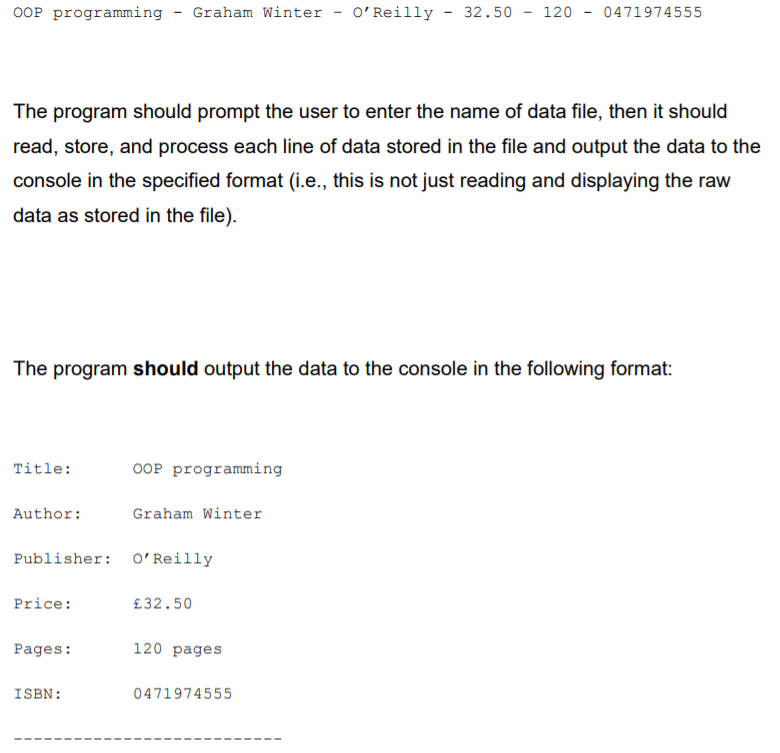
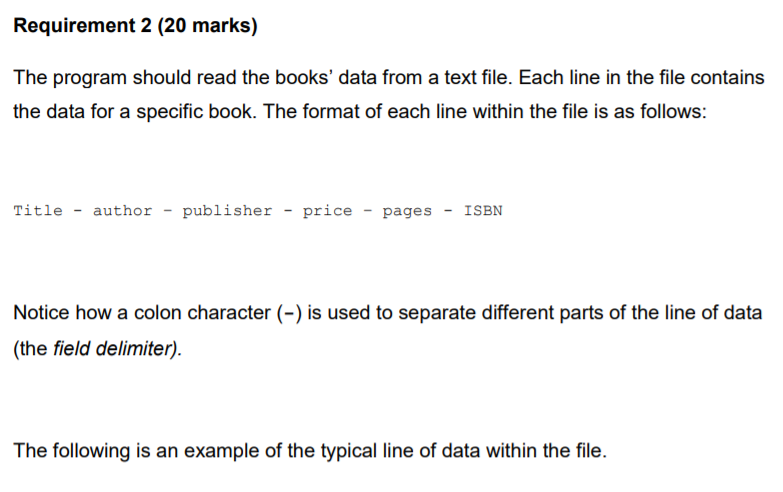
PRINT ("There is no author named "+ authorname );

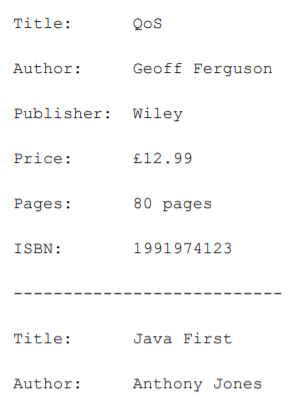
}

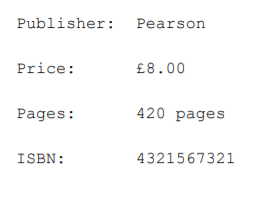
}

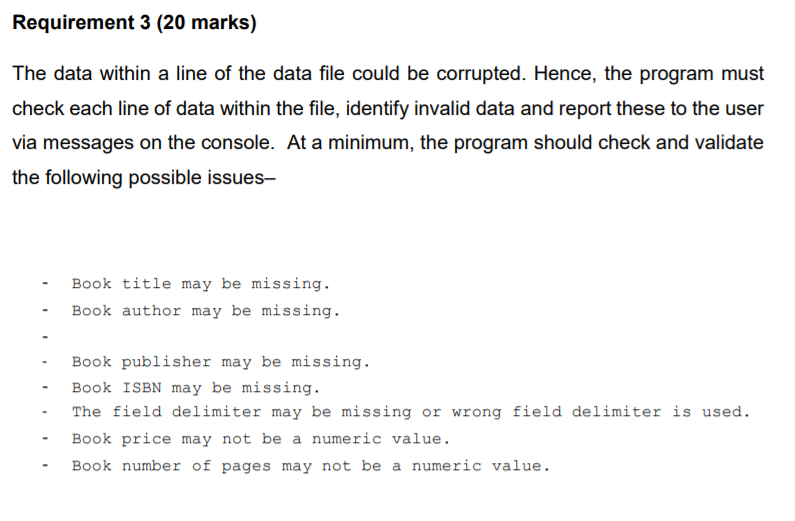
}

}











/\* Assignment 2, file handling \*/

**import** java.util.Scanner;

**import** java.io.\*;

//importing the necessary library

**public** **class** LibraryCatalogue { //creating a class

**static** **int** *stringLength*; //declaring a static variable

**public** **static** **boolean** isString(String string) { //creating a function to validate the strings

**boolean** isEmpty = **false**;

**for** (**int** i = 0; i <= string.length(); i++) { //initializing a for loop

*stringLength* = string.replaceAll(" ", "").length();

**if** (*stringLength* > 0) {

isEmpty = **true**;

} **else** {

isEmpty = **false**;

}

}

**return** isEmpty; //returing a value

}

// Function to perform validation of price

**public** **static** **boolean** isNumber(String number) {

**for** (**int** i = 0; i <= number.length(); i++) {

*stringLength* = number.replaceAll(" ", "").length();

}

**try** {

Float.*parseFloat*(number);

**return** **true**;

} **catch** (NumberFormatException e) {

**return** **false**;

}

}

**public** **static** **void** main(String[] args) //creating the main function

{

Scanner scan= **new** Scanner(System.***in***); //creating a scanner object scan

System.***out***.print("Enter name of the data file to read : "); //to ask the user name of the .txt file

String filename=scan.nextLine(); //scanning every line

File fileobject;

Scanner filereader=**null**;

String fileline;

String delimiter= **null**; //declaring variables

**int** invalidBooks=0;

**int** count=0;

**int** authorCount= 0;

**int** wrong\_count=0;

**int** invalidcount=0;

String[] wrong\_delimiter = {";", "\*", "+", "/","!","@","#","$","%","^","&"};

**try** {

fileobject = **new** File(filename+".txt"); //reading the .txt file

filereader = **new** Scanner(fileobject);

System.***out***.println("File found.\n");

}

**catch** (FileNotFoundException e)

{

System.***out***.println("File not found!!!!!!!!!!!!"); //if the file can't be found

}

String temp,Temp;

**int** index=0;

String[] title=**new** String[20]; //declaring array containing string values

String[] author=**new** String[20];

String[] publisher=**new** String[20];

String[] price=**new** String[20]; //declaring array containing double values

String[] pages=**new** String[20];

String[] ISBN=**new** String[20];

System.***out***.println("-----Error in file text-----"); //to list out all the errors

**while**(filereader.hasNext())

{

fileline = filereader.nextLine();

String[] splitline=fileline.split("-"); //splitting the line using the split()

**int** arraylength= splitline.length; //declaring a variable containing the length of the array

**try** {

**for** (**int** x=0; x<wrong\_delimiter.length;x++)

//checking wheather the lines contains invalid delimiter and missing delimiter

{

**if** (fileline.contains(wrong\_delimiter[x]) && (arraylength <6)) {

wrong\_count = wrong\_count + 1; //counting the number of wrong delimiter

delimiter= wrong\_delimiter[x];

}

}

**if** (wrong\_count > 0 ) { //if there is one or more invalid delimiter

System.***out***.println("The invalid field delimiter is used: "+ delimiter);

invalidBooks++; //counting the invalid books

**continue**; //to skip the null values }

**if**(splitline.length == 6 ) {

title[index]=splitline[0].trim(); //trimming the space from the words

author[index]=splitline[1].trim();

publisher[index]=splitline[2].trim();

price[index]=splitline[3].trim();

pages[index]=splitline[4].trim();

ISBN[index]=splitline[5].trim();

**if** (!*isString*(splitline[0])) { //for the missing title

System.***out***.println("Book title may be missing."); //isString is called from the function above

invalidcount++; //counting the invalid counts

}

**if** (!*isString*(splitline[1])) { //for the missing author

System.***out***.println("Book author may be missing.");

invalidcount++;

}

**if** (!*isString*(splitline[2])) { //for the missing publisher

System.***out***.println("Book publisher may be missing.");

invalidcount++;

}

**if** (!*isString*(splitline[5])) { //for the missing ISBN

System.***out***.println("ISBN may be missing.");

invalidcount++;

}

**if** (!*isNumber*(splitline[3])) { //for the invalid price

System.***out***.println("Book price maynot be numeric.");

invalidcount++;

}

**if** (!*isNumber*(splitline[4])) { //for the invalid page

System.***out***.println("Book page may be numeric.");

invalidcount++;

}

**if** (invalidcount > 0) //if the invalid count is one or more then increasing the value of invalidBooks

{

invalidBooks++; //increasing the number by one

}

}

}**catch**(NumberFormatException e) {}

index+=1; //increasing the index by one

}

System.***out***.println("------------Book Details------------\n"); //printing the book details

**for**(**int** j=0;j<index;j++) //using for loop

{

System.***out***.println("Title \t \t: "+title[j]);

System.***out***.println("Author \t \t: "+author[j]);

System.***out***.println("Publisher \t: "+publisher[j]);

System.***out***.println("Price \t \t: "+price[j]);

System.***out***.println("Pages \t \t: "+pages[j]);

System.***out***.println("ISBN \t \t: "+ISBN[j]);

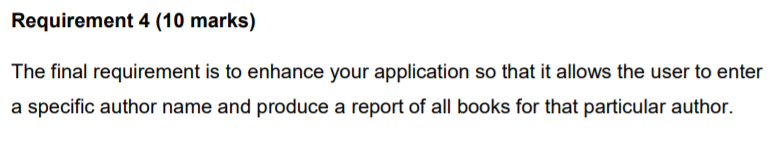
System.***out***.println("---------------------------------");

System.***out***.println("");

}

System.***out***.println("Number of invalid books: "+ invalidBooks); //printing the number of invalid books

System.***out***.println("\n");



String authorname;

System.***out***.print("Enter name of author for report of book by that author : "); //asking the user about the authername

authorname=scan.nextLine();

//authorname=authorname.toUpperCase().toString(); to change it into upper case

System.***out***.println("Books Written by "+authorname+" are : ");

**for**(**int** k=0; k<index; k++)

{

**if**((authorname).equals(author[k])) //comparing the authorname given by the user to the one in the data

{

System.***out***.println("---------------------------------"); //printing all the book details

System.***out***.println("Title \t \t: "+title[k]);

System.***out***.println("Publisher \t: "+publisher[k]);

System.***out***.println("Price \t \t: "+price[k]);

System.***out***.println("Pages \t \t: "+pages[k]);

System.***out***.println("ISBN \t \t: "+ISBN[k]);

System.***out***.println("---------------------------------");

authorCount++;

}

**else** {

**continue**;

}

**if** (authorCount == 0) {

System.***out***.println("Author not found!"); //In case there is no author in the data which has been given by the user.

}

}

}

}

**Submitted by: Garima Dhakal**

**Section: C**

**College ID: 6284**

**University ID: 77261088**