|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 5 |

**Дисциплина: Языки программирования для работы с большими данными.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-21М |  |  | В.В. Ерискин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2020

**Задание**

Задача 1.

Выполнить задания на основе варианта 1 лабораторной работы 4, контролируя состояние потоков ввода/вывода. При возникновении ошибок, связанных с корректностью выполнения математических операций, генерировать и обрабатывать исключительные ситуации. Предусмотреть обработку исключений, возникающих при нехватке памяти, отсутствии требуемой записи (объекта) в файле, недопустимом значении поля и т.д.

Создать класс Cinema (кино) с внутренним классом, с помощью объектов которого можно хранить информацию об адресах кинотеатров, фильмах и времени сеансов.

Задача 2.

Выполнить задания из варианта 2 лабораторной работы 4, реализуя собственные обработчики исключений и исключения ввода/вывода.

Реализовать абстрактные классы или интерфейсы, а также наследование и полиморфизм для следующих классов interface Врач <- class Хирург <- class Нейрохирург.

Задача 3.

В каждом слове стихотворения Николая Заболоцкого заменить первую букву слова на прописную.

Задача 4.

Прочитать текст Java-программы и удалить из него все “лишние” пробелы и табуляции, оставив только необходимые для разделения операторов.

**Решение**:

Программа 1.

public class Main {  
 public static void main(String[] args) {  
 //'Cinema' class testing.  
  
 Cinema cinema = new Cinema();  
 showAddStatus(cinema.add("Фильм", 2001, "Жанр1",  
 "проспект Мира, 17", 100, 5000000));  
 showAddStatus(cinema.add("Фильм2", 2002, "Жанр2",  
 "улица Строителей, 3", 50, 1000000));  
 showAddStatus(cinema.add("Фильм3", 2003, "Жанр3",  
 "улица Новая, 15", 75, 3000000));  
 showDeleteStatus(cinema.delete());  
 cinema.printAll();  
 showDeleteStatus(cinema.delete());  
 cinema.printAll();  
 showDeleteStatus(cinema.delete());  
 cinema.printAll();  
  
 showAddStatus(cinema.add("Фильм4", 2004, "Жанр4",  
 "улица Белинского, 16", 90, 3000000));  
 showAddStatus(cinema.add("Фильм5", 2005, "Жанр5",  
 "проспект Руставелли, 1", 120, 7000000));  
  
 cinema.printAll();  
 }  
  
  
 static void showAddStatus(boolean value) {  
 System.out.println("Adding operation status: " + value);  
 }  
  
  
 static void showDeleteStatus(boolean value) {  
 try { //добавим исключение - проверку  
 System.out.println("Deleting operation status: " + value);  
 } catch (IndexOutOfBoundsException e1) {  
 System.out.println("неверное значение");  
 }  
 }  
}

Программа 2.

Программа 3.

import java.io.\*;  
public class Main {  
 public static void main(String[] args){  
 try {  
 File f = new File("/Users/vitaly/Desktop/BadFile.txt");  
 FileInputStream fis=new FileInputStream(f);//читающий поток  
 DataInputStream ods = new DataInputStream(fis);  
 int size=fis.available();  
  
 char Array[]=new char[size];  
 for(int i=0;i<size;i++){  
 if(fis.read()!=-1){  
 Array[i]=ods.readChar();  
 System.out.println(Array[i]);  
 }}  
 fis.close();  
  
 FileOutputStream os = new FileOutputStream("/Users/vitaly/Desktop/NewFile.txt");  
 DataOutputStream od = new DataOutputStream(os);  
 for(int i=0;i<size;i++){  
 if(Array[i]==' '){  
 Array[i+1]=Character.toUpperCase(Array[i+1]);  
 }  
 od.writeChar(Array[i]);  
 }  
  
 os.close();  
 //закрытие потока  
 }  
  
 catch (IOException e){  
 System.out.println("ошибка файла: " + e);  
 }  
 }  
}

Программа 2.

|  |
| --- |
| import java.util.ArrayList; |
|  | import java.util.List; |
|  |  |
|  | public class Main { |
|  |  |
|  | public static void main(String[] args) { |
|  |  |
|  | Neurosurgeon neurosurgeon = new Neurosurgeon("Ringo Star"); |
|  |  |
|  | List<Surgeon> surgeons = new ArrayList<>(); |
|  | surgeons.add(neurosurgeon); |
|  |  |
|  | for (Surgeon surgeon: surgeons){ |
|  | surgeon.setDiagnos("Ann", 11); |
|  | surgeon.treat();//добавим исключение |
|  | ((Neurosurgeon)surgeon).sayName(); |
|  | try { |
|  | System.out.println("---------------------------------"); |
|  | }catch(IndexOutOfBoundsException e1){ |
|  | System.out.println("произошла ошибка"); |
|  | } |
|  |  |
|  | } |
|  |  |
|  | } |
|  |  |
|  | } |

Программа 3.

|  |  |
| --- | --- |
|  | import java.io.File; |
|  | import java.io.FileReader; |
|  | import java.io.FileWriter; |
|  | import java.io.IOException; |
|  |  |
|  | public class Main{ |
|  | public static void main (String[] args){ |
|  | File fileIN = new File("/Users/vitaly/Desktop/testFile.txt"); |
|  | File fileOUT = new File("/Users/vitaly/Desktop/testFile1.txt"); |
|  | int b; |
|  | char[] tempChar = new char[10000]; |
|  | try{ |
|  | FileReader fileReader = new FileReader(fileIN); |
|  | FileWriter fileWriter = new FileWriter(fileOUT); |
|  | for(int i = 0;(b = fileReader.read()) != -1; i++){tempChar[i] = (char)b;} |
|  | String tempString = new String(tempChar); |
|  | System.out.println("Программа перед редактированием: \n" + tempString); |
|  | tempString =(tempString.replaceAll("[\\s&&[^\r?\n]]{2,}", " ") |
|  | .replaceAll("( ?(\r\n)){2,}", "\r\n").replaceAll("( ?\n){2,}", "\n") |
|  | .replaceAll("\n ", "\n").trim());; |
|  | System.out.println("Программа после редактирования: \n" + tempString); |
|  | char[] resultText = tempString.toCharArray(); |
|  | fileWriter.write(resultText); |
|  | fileReader.close(); |
|  | fileWriter.close(); |
|  | }catch (IOException e) { |
|  | System.err.println("Ошибка файла: " + e); |
|  | } |
|  | } |
|  | } |

Программа 4.

import java.io.File;  
import java.io.FileReader;  
import java.io.FileWriter;  
import java.io.IOException;  
  
public class Main{  
 public static void main (String[] args){  
 File fileIN = new File("/Users/vitaly/Desktop/testFile.txt");  
 File fileOUT = new File("/Users/vitaly/Desktop/testFile1.txt");  
 int b;  
 char[] tempChar = new char[10000];  
 try{  
 FileReader fileReader = new FileReader(fileIN);  
 FileWriter fileWriter = new FileWriter(fileOUT);  
 for(int i = 0;(b = fileReader.read()) != -1; i++){tempChar[i] = (char)b;}  
 String tempString = new String(tempChar);  
 System.out.println("Программа перед редактированием: \n" + tempString);  
 tempString =(tempString.replaceAll("[\\s&&[^\r?\n]]{2,}", " ")  
 .replaceAll("( ?(\r\n)){2,}", "\r\n").replaceAll("( ?\n){2,}", "\n")  
 .replaceAll("\n ", "\n").trim());;  
 System.out.println("Программа после редактирования: \n" + tempString);  
 char[] resultText = tempString.toCharArray();  
 fileWriter.write(resultText);  
 fileReader.close();  
 fileWriter.close();  
 }catch (IOException e) {  
 System.err.println("Ошибка файла: " + e);  
 }  
 }  
}

**Вывод:** }

Задание выполнено, программы компилируются и успешно выполняются.