**《面向对象程序设计》课程实验报告**

**实验完成日期**： - -

|  |  |  |  |
| --- | --- | --- | --- |
| **课程名称**：**面向对象程序设计** | | **专业：**软件工程 | **教学班级：** |
| **姓名**：贠若轩 | | **学号**：2410250062 | **指导教师：** |
| **实验名称**：**实验7 IO流** | | | **实验学时：4学时** |
| **实验类型** | ☑验证性 □演示性 ☑设计性 ☑综合性 | | |
| **实验重点** | File类的使用；FileInputStream和FileOutputStream类的使用：BufferedReader和BufferedWriter类的使用。 | | |
| **实验难点** | OutputStreamWriter和InputStreamReader的使用。 | | |
| **实验环境** | jdk8.0以上版本，IDEA集成开发环境。 | | |

**一、实验目的**

1、理解字节流和字符流的含义，掌握IO流的分类；

2、掌握File类的使用；

3、掌握文件字节流FileInputStream、FileOutputStream和文件字符流FileReader、FileWriter的使用；

4、掌握字符缓冲流BufferedReader、BufferedWriter的使用；

5、了解转换流OutputStreamWriter和InputStreamReader的使用；

6、能够灵活使用IO流完成数据的处理。

**二、实验内容**

1、判断D盘指定目录下是否有后缀名为.exe的文件，如果有就输出此文件名称。

**源代码：**

package qimo.shiyan7;  
  
import java.io.File;  
  
public class ExeFileFinder {  
 public static void main(String[] args) {  
 File dir = new File("D:\\指定目录"); // 替换为实际目录路径  
  
 if (dir.exists() && dir.isDirectory()) {  
 File[] files = dir.listFiles((d, name) -> name.toLowerCase().endsWith(".exe"));  
  
 if (files != null && files.length > 0) {  
 System.*out*.println("找到的.exe文件:");  
 for (File file : files) {  
 System.*out*.println(file.getName());  
 }  
 } else {  
 System.*out*.println("没有找到.exe文件");  
 }  
 } else {  
 System.*out*.println("目录不存在或不是有效目录");  
 }  
 }  
}

**列出测试数据和实验结果截图：**



2、分别使用字节流和字节缓冲流的两种读取方式实现对音频文件的复制操作并比较两种方式在复制时间上的效率。

**源代码：**

package qimo.shiyan7;  
  
import java.io.\*;  
  
public class AudioCopyComparison {  
 public static void main(String[] args) {  
 String sourceFile = "test.mp3"; // 替换为实际音频文件  
 String destFile1 = "copy1.mp3";  
 String destFile2 = "copy2.mp3";  
  
 // 普通字节流  
 long start1 = System.*currentTimeMillis*();  
 try (InputStream is = new FileInputStream(sourceFile);  
 OutputStream os = new FileOutputStream(destFile1)) {  
 byte[] buffer = new byte[1024];  
 int length;  
 while ((length = is.read(buffer)) > 0) {  
 os.write(buffer, 0, length);  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 long end1 = System.*currentTimeMillis*();  
  
 // 缓冲字节流  
 long start2 = System.*currentTimeMillis*();  
 try (InputStream is = new BufferedInputStream(new FileInputStream(sourceFile));  
 OutputStream os = new BufferedOutputStream(new FileOutputStream(destFile2))) {  
 byte[] buffer = new byte[1024];  
 int length;  
 while ((length = is.read(buffer)) > 0) {  
 os.write(buffer, 0, length);  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 long end2 = System.*currentTimeMillis*();  
  
 System.*out*.println("普通字节流耗时: " + (end1 - start1) + "ms");  
 System.*out*.println("缓冲字节流耗时: " + (end2 - start2) + "ms");  
 }  
}

**列出测试数据和实验结果截图：**



3、编写一个程序，分别使用转换流、字符流和缓冲字符流拷贝一个文本文件。要求：

• 分别使用InputStreamReader、OutputStreamWriter类和FileReader、FileWriter类用两种方式（字符和字符数组）进行拷贝。

• 使用BufferedReader、BufferedWriter类的特殊方法进行拷贝。

**源代码：**

package qimo.shiyan7;  
import java.io.\*;  
  
public class TextFileCopy {  
 public static void main(String[] args) {  
 String source = "source.txt";  
 String dest1 = "dest1.txt";  
 String dest2 = "dest2.txt";  
 String dest3 = "dest3.txt";  
  
 // 1. 使用转换流  
 try (Reader isr = new InputStreamReader(new FileInputStream(source));  
 Writer osw = new OutputStreamWriter(new FileOutputStream(dest1))) {  
 int c;  
 while ((c = isr.read()) != -1) {  
 osw.write(c);  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
  
 // 2. 使用字符流 - 字符方式  
 try (FileReader fr = new FileReader(source);  
 FileWriter fw = new FileWriter(dest2)) {  
 int c;  
 while ((c = fr.read()) != -1) {  
 fw.write(c);  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
  
 // 3. 使用字符流 - 字符数组方式  
 try (FileReader fr = new FileReader(source);  
 FileWriter fw = new FileWriter(dest3)) {  
 char[] buffer = new char[1024];  
 int length;  
 while ((length = fr.read(buffer)) > 0) {  
 fw.write(buffer, 0, length);  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
  
 // 4. 使用缓冲字符流  
 String dest4 = "dest4.txt";  
 try (BufferedReader br = new BufferedReader(new FileReader(source));  
 BufferedWriter bw = new BufferedWriter(new FileWriter(dest4))) {  
 String line;  
 while ((line = br.readLine()) != null) {  
 bw.write(line);  
 bw.newLine();  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
  
 System.*out*.println("文件拷贝完成");  
 }  
}

**列出测试数据和实验结果截图：**

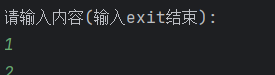
4、编程序实现下列功能：

• 向指定的txt文件中写入键盘输入的内容，然后再重新读取该文件的内容，显示到控制台上。

• 键盘录入5个学生信息(姓名, 成绩)，按照成绩从高到低追加存入上述的文本文件中。

**源代码：**

**列出测试数据和实验结果截图：**



5、复制指定目录中的指定类型（如.java）的文件到另一个目录中。

**源代码：**

**列出测试数据和实验结果截图：**

6、已知s.txt文件中有这样的一个字符串：“hcexfgijkamdnoqrzstuvwybpl”，请编写程序读取数据内容，把数据排序后写入ss.txt中。

**源代码：**

**列出测试数据和实验结果截图：**

**三、实验总结**

**对实验结果进行分析，列出错误及解决办法，回答问题，总结实验的心得体会，以及提出改进意见。**

**注：**

**1、实验报告完成后，修改文件名为：实验7-实验报告。**

**2、将java源程序文件按照题号命名，如S7\_1.java、S7\_2.java、S7\_3.java，…，并将相关源代码、输入输出文件和实验报告文件，一起打包，命名为班级号-学号-姓名-实验7。注：文件路径建议使用相对路径。**