**河南工业大学 Java程序设计实验 实验报告**

专业班级：计科2101 学号： 211040100127 姓名： 翟向阳 指导老师： 赵晨阳 评分：

1. **实验题目**：实验（六）：实用类和输入输出流
2. **实验目的**：1. 熟悉Java 中实用类。2.熟悉Java中各种输入输出流。
3. **实验要求：1. 每个题目，写出实验思路；2. 代码实现；3. 运行结果截图。**
4. **实验内容**：

1．String类的public char charAt(int index)方法可以得到当前字符串index位置上的一个字符。试自己编写一个方法，以自己的方法重新实现charAt方法的功能。另外，以自己的方法编写一个reverse方法，可以将字符串逆序。（注：不要调用现有的类库中方法）

2.编写程序删除一个字符串中的全部非数字字符，例如，将形如“ab123you”的非数字字符全部删除，得到字符串“123”。

3.使用StringTokenizer类解析字符串："数学87分，物理76分，英语96分"中的考试成绩，并计算出总成绩以及平均分数。

4.在当前目录下创建一个文件，命名为学号.txt，判定文件的存在性，如果没有，创建物理文件，同时读取文件的各种属性信息。

5. 编写一个Java程序：（1）定义一个copyFile(String sFile, String dFile)的方法，将文件sFile中的内容复制到文件dFile中，要求使用try-catch进行异常处理，若捕获异常则通过调用IOException异常类的toString()方法显示异常的相关信息；（2）在main方法中指定源文件和目标文件分别为当前目录下的source.txt和backup.txt，然后调用copyFile完成文件复制操作。

6. 编写一个Java程序：使用Java的输入、输出流将一个文本文件的内容按行读出，每读出一行就顺序添加行号，并写入到另一个文件中，要求使用try-catch进行异常处理，若捕获异常则通过调用IOException异常类的toString()方法显示异常的相关信息。

1. **实验执行：**

1. String类的public char charAt(int index)方法可以得到当前字符串index位置上的一个字符。试自己编写一个方法，以自己的方法重新实现charAt方法的功能。另外，以自己的方法编写一个reverse方法，可以将字符串逆序。（注：不要调用现有的类库中方法）

（1）思路

先将字符串转化为char类型的数组，直接返回该位置的字符就好，第二个逆转，也是先将字符串转化为char类型的数组，再将字符从尾至头一个一个放到新定义的字符串b中，返回b数组即可

（2）代码实现

public class test1 {

public static void main(String[] args) {

string1 a = new string1("123456");

System.out.println(a.charAt(5));

System.out.println(a.reverse());

}

}

class string1{

private String a;

string1(String a){

this.a=a;

}

public char charAt(int index){

char[] a1 = a.toCharArray();

return a1[index];

}

public String reverse(){

String b = "";

char[] a1 = a.toCharArray();

for(int j=a.length()-1;j>=0;j--){

b = b + a1[j];

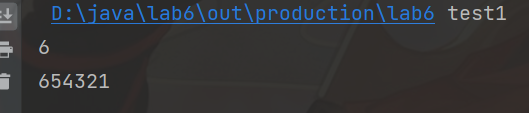
}

return b;

}

}

（3）运行结果截图



2.编写程序删除一个字符串中的全部非数字字符，例如，将形如“ab123you”的非数字字符全部删除，得到字符串“123”。

（1）

先将a转化为char数组，再进行判断，若其中字符大于1小于9 ，那么将该字符加到b中，返回b字符串。

（2）

public static String delete(String a){

String b = "";

char[] a1 = a.toCharArray();

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

if(a1[i]>='1'&&a1[i]<='9'){

b+=a1[i];

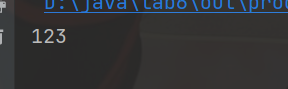
}

}

return b;

}

（3）



3.使用StringTokenizer类解析字符串："数学87分，物理76分，英语96分"中的考试成绩，并计算出总成绩以及平均分数。

（1）

（2）

String a = "数学87分,物理76分,英语96分";

String regex = "[^0123456789.]+";

a = a.replaceAll(regex,"#");

StringTokenizer fenxi = new StringTokenizer(a,"#");

double sum=0;

int amount = fenxi.countTokens();

while (fenxi.hasMoreTokens()){

String item = fenxi.nextToken();

double fenshu = Double.parseDouble(item);

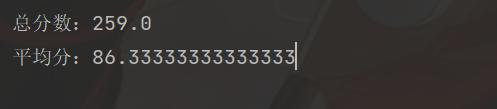
sum+=fenshu;

}

System.out.println("总分数："+sum);

System.out.println("平均分："+sum/amount);

（3）



4.在当前目录下创建一个文件，命名为学号.txt，判定文件的存在性，如果没有，创建物理文件，同时读取文件的各种属性信息。

（1）

（2）

File file = new File("学号.txt");

if(!file.exists()){

try {

file.createNewFile();

System.out.println("创建成功！");

} catch (IOException e) {

throw new RuntimeException(e);

}

}

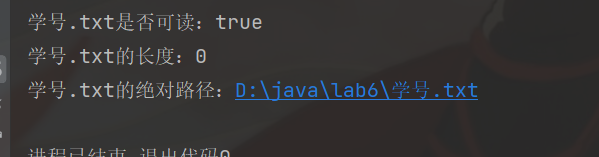
System.out.println(file.getName()+"是否可读："+file.canRead());

System.out.println(file.getName()+"的长度："+file.length());

System.out.println(file.getName()+"的绝对路径："+file.getAbsolutePath());

}

（3）



5. 编写一个Java程序：（1）定义一个copyFile(String sFile, String dFile)的方法，将文件sFile中的内容复制到文件dFile中，要求使用try-catch进行异常处理，若捕获异常则通过调用IOException异常类的toString()方法显示异常的相关信息；（2）在main方法中指定源文件和目标文件分别为当前目录下的source.txt和backup.txt，然后调用copyFile完成文件复制操作。

（1）

首先通过缓冲流定义sFile 和dFile 的BufferedReader 类，通过readLine 方法从sFile中读取其中的内容放入strLine中，最后通过缓冲输出流将内容写入d.txt文件中。

（2）

import java.io.\*;

public class test5 {

public static void main(String[] args) {

String s = ("s.txt");

String d = ("d.txt");

copyFile(s,d);

}

public static void copyFile(String sFile,String dFile){

try{

BufferedReader bs = new BufferedReader(new FileReader(sFile));

BufferedReader bd = new BufferedReader(new FileReader(dFile));

BufferedWriter bw = new BufferedWriter(new FileWriter(dFile));

String strLine = null;

while ((strLine=bs.readLine())!=null){

bw.write(strLine);

System.out.println(strLine);

bw.close();

System.out.println(bd.readLine());

}

bs.close();

bd.close();

} catch (IOException e) {

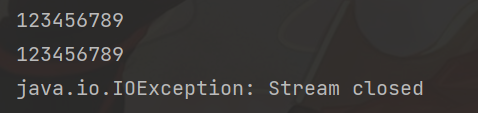
System.out.println(e.toString());

}

}

}

（3）



6. 编写一个Java程序：使用Java的输入、输出流将一个文本文件的内容按行读出，每读出一行就顺序添加行号，并写入到另一个文件中，要求使用try-catch进行异常处理，若捕获异常则通过调用IOException异常类的toString()方法显示异常的相关信息。

（1）

跟上一个实验差不多，但是这次在写入文件的时候需要写入行号，strLine = "第"+i+"行："+strLine;利用这个更改一下strLine的值,再将此值写入文件中。

（2）

import java.io.\*;

public class test6 {

public static void main(String[] args) {

try{

FileReader source = new FileReader("s.txt");

FileReader target = new FileReader("d.txt");

FileWriter targetWrite = new FileWriter("d.txt");

BufferedReader bs = new BufferedReader(source);

BufferedReader bt = new BufferedReader(target);

BufferedWriter btw = new BufferedWriter(targetWrite);

String strLine = null;

int i=1;

while ((strLine=bs.readLine())!=null){

strLine = "第"+i+"行："+strLine;

btw.write(strLine);

btw.newLine();

System.out.println(strLine);

i++;

}

btw.close();

} catch (IOException e) {

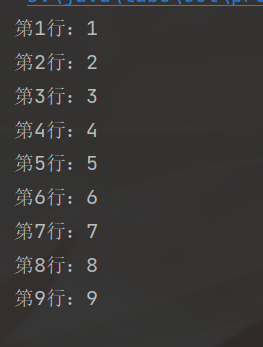
System.out.println(e.toString());

}

}

}

（3）



1. **实验测试中遇到的问题和自己的解决方案**

更加清楚的了解了文件类型的操作，感觉java中的文件操作比c 或者c++更加方便，采用像对象一样的操作，在缓冲流中如果最后不关闭该流，那么最后并不会被写入文件中。