# 实验八 Java IO

【实验目的】

1.了解输入输出流类的基本使用方法。

2.理解Java 系统数据流的构成。

【实验要求】

1.掌握字节输入输出流、字符输入输出流。

2.应用流完成输入输出操作。

【实验内容】

1. 利用字节流从键盘上输入一串字符串并输出到屏幕上。

import java.io.IOException;

public class Main {  
 public static void main(String[] args) throws IOException {  
 byte[] buffer = new byte[1024];  
 int bytesRead = System.*in*.read(buffer); *// 从键盘读取输入的字节并存储到buffer中* String inputString = new String(buffer, 0, bytesRead);  
 System.*out*.println(inputString);  
 }  
}

2.利用字符流从键盘上输入一串字符串并输出到屏幕上。

import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStreamReader;  
  
public class StreamRead {  
 public static void main(String[] args) throws IOException {  
 InputStreamReader inputStreamReader = new InputStreamReader(System.*in*);  
 BufferedReader bufferedReader = new BufferedReader(inputStreamReader);  
 String inputString = bufferedReader.readLine(); *// 从键盘读取输入的字符串* System.*out*.println(inputString);  
 bufferedReader.close();  
 inputStreamReader.close();  
 }  
}

3.利用字节流从磁盘上输入一个文件，并将其输出到另一个文件中，实现文件的复制。

import java.io.FileInputStream;  
import java.io.FileOutputStream;  
import java.io.IOException;  
  
public class FileByteRead {  
  
 private static final String *originalFilePath* = "C:\\Users\\lys\\Desktop\\smjb.txt";  
 private static final String *copyOfFilePath* = "C:\\Users\\lys\\Desktop\\wdnmd.txt";  
  
 public static void main(String[] args) {  
 try (FileInputStream fileInputStream = new FileInputStream(*originalFilePath*)) {  
 try (FileOutputStream outputStream = new FileOutputStream(*copyOfFilePath*)) {  
 byte[] buffer = new byte[1024];  
 int bytesRead;  
 while ((bytesRead = fileInputStream.read(buffer)) != -1) {  
 outputStream.write(buffer, 0, bytesRead);  
 }  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
}

4.利用字符流从磁盘上输入一个文件，并将其输出到另一个文件中，实现文件的复制。

import java.io.FileReader;  
import java.io.FileWriter;  
import java.io.IOException;  
  
public class FileCharRead {  
 private static final String *originalFilePath* = "C:\\Users\\lys\\Desktop\\smjb.txt";  
 private static final String *copyOfFilePath* = "C:\\Users\\lys\\Desktop\\wdnmd.txt";  
  
 public static void main(String[] args) {  
 try (FileReader fileReader = new FileReader(*originalFilePath*)) {  
 try (FileWriter fileWriter = new FileWriter(*copyOfFilePath*)) {  
 int character;  
 while ((character = fileReader.read()) != -1) {  
 fileWriter.write(character);  
 }  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
}

5.教材例10.1-10.5、10.8-10.10

10-1：

package Src.com.learn.IO.IOBook;

import java.io.File;

public class Demo\_01 {

public static void main(String[] args)

{

// 创FIle文件对象，使用绝对路径

File file\_01 = new File("D:" + File.separator + "learn.txt");

// 创建文件对象，使用相对路径

File file\_02 = new File("Src/com/learn/zs.txt");

System.out.println(file\_01);

System.out.println(file\_02);

}

}

10-2：

package Src.com.learn.IO.IOBook;

import java.io.File;

public class Demo\_02 {

public static void main(String[] args)

{

File f1 = new File("Src/com/learn/yyds.txt");

System.out.println("文件名：" + f1.getName());

System.out.println("文件是否存在：" + f1.exists());

System.out.println("文件大小:" + f1.length());

System.out.println("文件的相对路径" + f1.getPath());

System.out.println("文件的绝对路径：" + f1.getAbsolutePath());

System.out.println("文件的父级对象是否为文件：" + f1.isFile());

System.out.println("文件是否删除成功:" + f1.delete());

}

}

12-3：

package Src.com.learn.IO.IOBook;

import java.io.File;

import java.io.IOException;

public class Demo\_03 {

public static void main(String[] args) throws IOException {

// 创建临时文件

File file = File.createTempFile("yyds",".txt");

// java虚拟机退出时自动删除文件

file.deleteOnExit();

System.out.println("是否为文件：" + file.isFile());

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

}

}

10-4：

package Src.com.learn.IO.IOBook;

import java.io.File;

public class Demo\_04 {

public static void main(String[] args) {

File file = new File("Src/com/learn/");

if(file.isDirectory())

{

for (String s : file.list()) {

System.out.println(s);

}

}

}

}

10-5：

package Src.com.learn.IO.IOBook;

import java.io.File;

import java.io.FilenameFilter;

import java.util.Arrays;

public class Demo\_05 {

public static void main(String[] args) {

File file = new File("Src/com/learn/IO/IOBook");

FilenameFilter filenameFilter = (f,name) -> {

File file1 = new File(f, name);

if(file1.isFile() && name.endsWith(".java")) return true;

else return false;

};

String[] list = file.list(filenameFilter);

Arrays.stream(list).forEach(System.out::println);

}

}

10-8：

package Src.com.learn.IO.IOBook;

import java.io.File;

public class Demo\_08 {

public static void main(String[] args) {

File file = new File("Src/com/learn/StringDemo");

deleteDir(file);

System.out.println("删除成功");

}

public static void deleteDir(File file)

{

if(file.exists())

{

for (File f : file.listFiles()) {

if(f.isDirectory())

{

deleteDir(f);

}

else f.delete();

}

file.delete();

}

}

}

10-9

package Src.com.learn.IO.IOBook;

import java.io.BufferedInputStream;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.InputStream;

public class Demo\_09 {

public static void main(String[] args) {

try(

InputStream is = new FileInputStream("Src/com/learn/a.txt");

BufferedInputStream bufferedInputStream = new BufferedInputStream(is);

) {

int a;

while(true)

{

a = bufferedInputStream.read();

if(a == -1) break;

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

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

10-10

package Src.com.learn.IO.IOBook;

import java.io.BufferedOutputStream;

import java.io.FileOutputStream;

import java.io.OutputStream;

import java.nio.charset.StandardCharsets;

import java.util.Scanner;

public class Demo\_10 {

public static void main(String[] args) {

try(

Scanner sc = new Scanner(System.in);

OutputStream os = new FileOutputStream("Src/com/learn/a.txt");

BufferedOutputStream bo = new BufferedOutputStream(os);

){

System.out.println("请输入需要向文件中输入的数据");

String next = sc.next();

bo.write(next.getBytes(StandardCharsets.UTF\_8));

}catch(Exception e)

{

e.printStackTrace();

}

}

}

【实验成绩】