河南工业大学

《JAVA 程序设计》实验报告

专业班级：物联网 1603　　学号：201616070320　　姓名：郭治洪

实验单元三　Java 的输入、输出机制

实验四 Java 的输入机制 & 实验五 Java 的输出机制

实验时间：2018.11

【实验目的】

1、Java 如何操作文件。

2、了解 Java 中的输入机制：如何从控制台输入，如何用 InputStream 和 Reader 显示文件的内容。

3、掌握 Java 中的输出机制，会使用 OutputStream、Writer 输出。

4、能够结合输入、输出复制文件内容。

【实验环境】

JDK、Eclipse

【实验内容】

教材 P285：实验内容

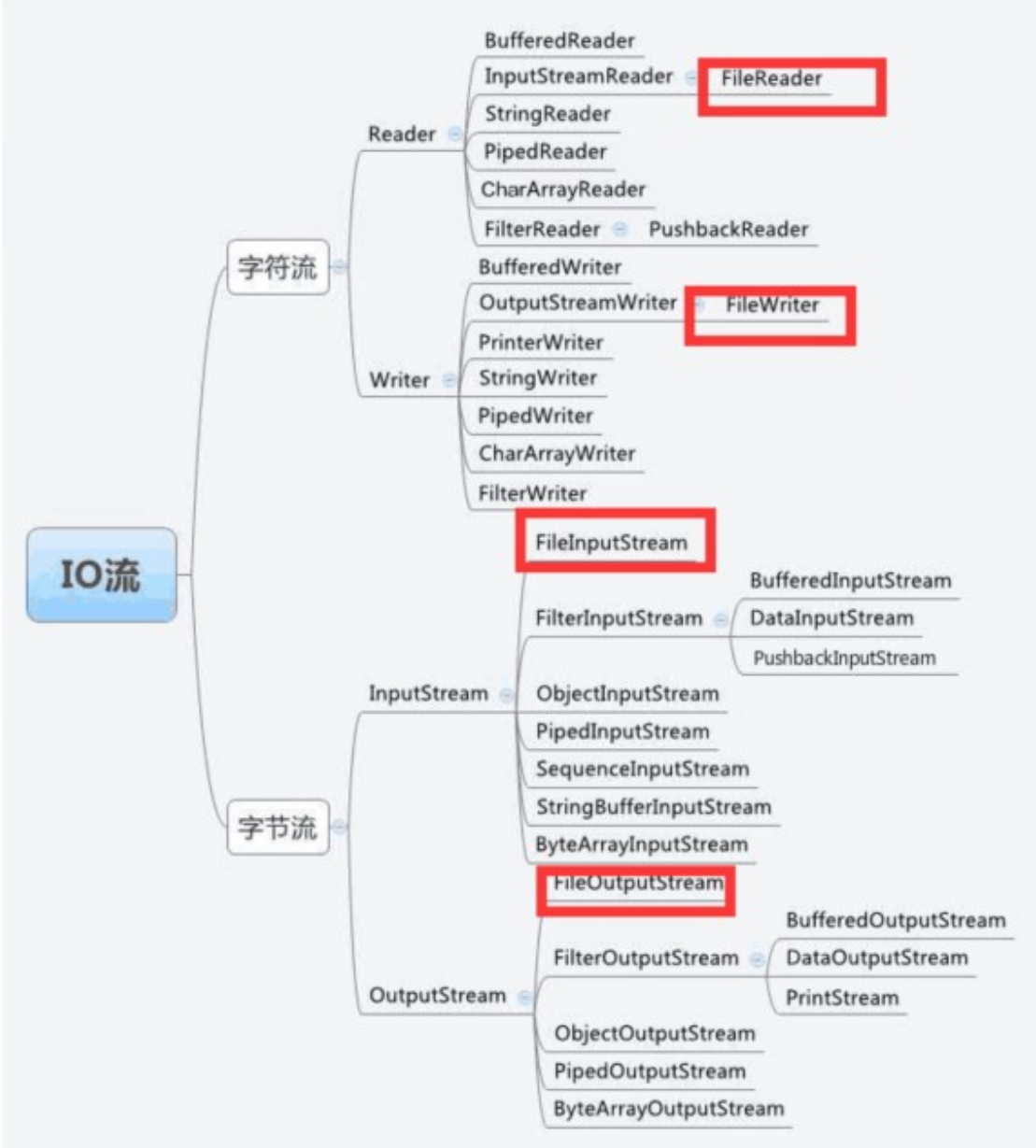
【详细分析】

首先弄明白 JAVA 的 IO 流有两种，分别是字符流和字节流，因为 JAVA 是 UNICODE 编码支持的,而一些语言可能不支持使用一个字节（8Bit/1Byte），因此我们需要根据不同编码集（GBK,GB2312,UTF8,UTTF16,BIG5）从一些字节流翻译出来的字符，编码集可以手工指定，也可以让编译器根据系统环境自动编译。但是所有的 IO 操作都是基于字节流做的，需要我们根据情况进行选择。

另外还可以是缓冲区（缓冲字符/缓存字节）加快读取和写入操作。

例如 **abstract int read(); 和 abstract int read(byte[] b);**

这是实验我使用了 4 个 IO 类，以及一个 Scanner 类来完成。



参考链接：

https://www.zhihu.com/question/39262026

http://www.runoob.com/java/java-scanner-class.html

https://www.ibm.com/developerworks/cn/java/j-lo-chinesecoding/

http://www.ruanyifeng.com/blog/2007/10/ascii_unicode_and_utf-8.html

http://www.runoob.com/java/java-files-io.html

https://www.geeksforgeeks.org/java-io-reader-class-java/

http://tutorials.jenkov.com/java-io/readers-writers.html

【实验源码】

A. Experiment10.java

```
/*
 * Java Experiment 10:  I/O Stream
 * Data:2018.9
 * Collage:Internet of Things
 * Class:IoT 1603
 * Name:GuoZhiHong
 * Student ID:201616070320
 */

package Experiment10;

public class Experiment10 {
    public static void main(String[] args) {
        IOStream io= new IOStream();
        IOStream.InputClass InputClassTest=io.new InputClass();
        //ListFile Test

System.out.println("----------------------------------------");
        System.out.println("The ListFile Test output is:");

System.out.println("----------------------------------------");
        InputClassTest.ListAllFileInTheDirectory("D:\\");
        //Reader Read by Char Test

System.out.println("----------------------------------------");
        System.out.println("Reader Read by Char Test output is:");

System.out.println("----------------------------------------");
        InputClassTest.CharReader("D:\\test.txt");
        //Reader Read by Buffer Test

System.out.println("----------------------------------------");
        System.out.println("Reader Read by Buffer Test output is:");

System.out.println("----------------------------------------");
        InputClassTest.BufferReader("D:\\test.txt");
        //InputStream Read by Char Test

System.out.println("----------------------------------------");
        System.out.println("InputStream Read by Char Test output is:");
```

```
System.out.println("----------------------------------------");
        InputClassTest.CharInputStream("D:\\test.txt");
        //InputStream Read by Buffer Test

System.out.println("----------------------------------------");
        System.out.println("InputStream Read by Buffer Test output is:");

System.out.println("----------------------------------------");
        InputClassTest.BufferInputStream("D:\\test.txt");
        //Copy The Image And Text File Test
        IOStream.OutputClass OutputClassTest=io.new OutputClass();
        OutputClassTest.CopyTextFile("D:\\test.txt","D:\\test2.txt");
        OutputClassTest.CopyImageFile("D:\\road.jpg","D:\\road2.jpg");
        //Student Class Test

System.out.println("----------------------------------------");
        System.out.println("Start to test student class.");

System.out.println("----------------------------------------");
        Student StudentA=new Student();
        StudentA.ShowInfo();
        StudentA.SaveToFile("D:\\student.txt");
        Student StudentB=new Student("002","Mike",20);
        StudentB.ShowInfo();
        StudentB.SaveToFile("D:\\student.txt");
    }

}
```

B. IOStream.java

```
package Experiment10;

import java.io.*;

public class IOStream {
    class InputClass {
        public void ListAllFileInTheDirectory(String DirectoryPath) {
            if (!DirectoryPath.isEmpty()) {
                try {
                    File open = new File(DirectoryPath);
                    File[] FileList = open.listFiles();
```

```
                System.out.println("Types                 Name
File Size");
                for (File EveryFile : FileList) {
                    if (EveryFile.isDirectory()) {
                        System.out.printf("Directory        %-25s
---\n", EveryFile.getName());
                    } else if (EveryFile.isFile()) {

System.out.printf("File            %-25s               %.2f KB\n",
EveryFile.getName(), (double) EveryFile.length() / 1024.0);
                    }
                }

System.out.println("----------------------------------------------
-----");
            } catch (Exception e) {
                e.printStackTrace();
            }
        } else {
            System.out.println("You must set the correct direcory");
        }
    }
    public void CharReader(String FilePath) {
        if(!FilePath.isEmpty()) {
            int i;
            char c;
            try {
                Reader f=new FileReader(FilePath);
                while((i=f.read())!=-1) {
                    c = (char) i;
                    System.out.print(c);
                }
                System.out.print("\n");
                f.close();
            }
            catch (IOException e) {
                e.printStackTrace();
            }
        }
        else {
            System.out.println("You must set the correct file path");
        }
    }
    public void BufferReader(String FilePath) {
```

```
    if(!FilePath.isEmpty()) {
        try {
            Reader f=new FileReader(FilePath);
            char[] buffer=new char[1048];
            StringBuffer output=new StringBuffer();
            while(f.read(buffer)!=-1) {
                String temp=new String(buffer);
                output.append(temp);
            }
            System.out.println(output.toString());
            f.close();
        }
        catch (IOException e) {
            e.printStackTrace();
        }
    }
    else {
        System.out.println("You must set the correct file path");
    }
}
public void CharInputStream(String FilePath) {
    if(!FilePath.isEmpty()) {
        int i;
        char c;
        try {
            InputStream f=new FileInputStream(FilePath);
            while((i=f.read())!=-1) {
                c = (char) i;
                System.out.print(c);
            }
            System.out.print("\n");
            f.close();
        }
        catch (IOException e) {
            e.printStackTrace();
        }

    }
    else {
        System.out.println("You must set the correct file path");
    }
}
public void BufferInputStream(String FilePath) {
    if(!FilePath.isEmpty()) {
```

```
                try {
                    InputStream f=new FileInputStream(FilePath);
                    byte[] buffer=new byte[1048];
                    StringBuffer output=new StringBuffer();
                    while(f.read(buffer)!=-1) {
                        String temp=new String(buffer);
                        output.append(temp);
                    }
                    System.out.println(output.toString());
                    f.close();
                }
                catch (IOException e) {
                    e.printStackTrace();
                }
            }
            else {
                System.out.println("You must set the correct file path");
            }
        }
    }
    class OutputClass {
        public void CopyImageFile(String SourceFilePath,String
DestinationFilePath) {

if(!SourceFilePath.isEmpty()&&!DestinationFilePath.isEmpty()) {
                InputStream in=null;
                OutputStream out=null;
                try {
                    in=new FileInputStream(SourceFilePath);
                    out=new FileOutputStream(DestinationFilePath);
                    byte[] buffer=new byte[8*1024];
                    while(in.read(buffer)!=-1) {
                        out.write(buffer);
                    }
                    if(in!=null)
                        in.close();
                    if(out!=null)
                        out.close();
                } catch (IOException e) {
                    e.printStackTrace();
                }
            }
            else {
                System.out.println("You must set the correct source file
```

```
path and destination file path");
            }
        }
        public void CopyTextFile(String SourceFilePath,String
DestinationFilePath) {

if(!SourceFilePath.isEmpty()&&!DestinationFilePath.isEmpty()) {
                FileReader in=null;
                FileWriter out=null;
                try {
                    in=new FileReader(SourceFilePath);
                    out=new FileWriter(DestinationFilePath);
                    char[] buffer=new char[8*1024];
                    while(in.read(buffer)!=-1) {
                        out.write(buffer);
                    }
                    if(in!=null)
                        in.close();
                    if(out!=null)
                        out.close();
                } catch (IOException e) {
                    e.printStackTrace();
                }
            }
            else {
                System.out.println("You must set the correct source file
path and destination file path");
            }
        }
    }
}
```

C. Student.java

```
package Experiment10;

import java.io.FileWriter;
import java.io.IOException;
import java.util.Scanner;

public class Student {
    private String ID="NULL";
    private String name="NULL";
```

```
    private int ages=0;
    Student(String _ID,String _name,int _ages) {
        this.SetInfo(_ID,_name,_ages);
    }
    Student() {
        this.SetInfo();
    }
    public void ShowInfo() {
        System.out.println(this.toString());
    }
    @Override
    public String toString() {
        return "The student 's ID is " + this.ID+",\n"
               + "The person 's name is "+this.name+",\n"
               +"The person 's age is "+this.ages+".";
    }
    public void SetInfo(String _ID,String _name,int _ages) {
        Scanner sc = null;
        if(_ID.matches("^[0-9]*$") == false) { //Check if the received ID
value is reasonable
            sc = new Scanner(System.in);
            System.out.println("Construction error,please input the
correct ID and retry again.");
            this.InputID(sc);
        }
        else {
            this.ID=_ID;
        }

if(_name.matches("^[\\u4E00-\\u9FA5A-Za-z\\s]+(·[\\u4E00-\\u9FA5A-Za-
z]+)*$")==false) {
            System.out.println("Construction error,please input the
correct name and retry again.");
            this.InputName(sc);
        }
        else {
            this.name=_name;
        }
        if(_ages<0||_ages>120) {
            System.out.println("Construction error,please input the
correct ages and retry again.");
            this.InputAges(sc);
        }
        else {
```

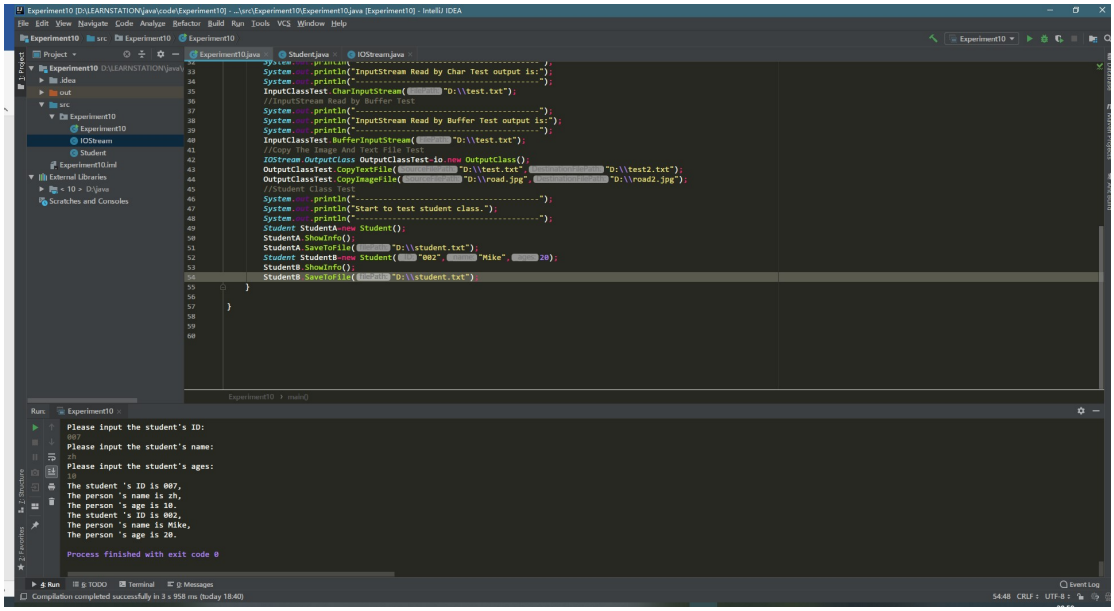
```
            this.ages=_ages;
        }
        if(sc!=null) {
            sc.close();
        }
    }
    public void SetInfo() {
        Scanner sc=new Scanner(System.in);
        this.InputID(sc);
        this.InputName(sc);
        this.InputAges(sc);
        sc.close();
    }
    private void InputName(Scanner sc) {
        String _name;
        while(true) {
            System.out.println("Please input the student's name:");
            _name=sc.next();

if(_name.matches("^[\\u4E00-\\u9FA5A-Za-z\\s]+(·[\\u4E00-\\u9FA5A-Za-
z]+)*$") == false) {
                System.out.println("Input error,please input the correct
name and retry again.");
            }
            else {
                this.name=_name;
                break;
            }
        }
    }
    private void InputID(Scanner sc) {
        String _ID;
        while(true) {
            System.out.println("Please input the student's ID:");
            _ID=sc.next();
            if(_ID.matches("^[0-9]*$") == false) {
                System.out.println("Input error,please input the correct
ID and retry again.");
            }
            else {
                this.ID=_ID;
                break;
            }
        }
```

```
    }
    private void InputAges(Scanner sc) {
        int _ages = 0;
        while(true) {
            System.out.println("Please input the student's ages:");
            _ages=sc.nextInt();
            if(_ages<0||_ages>120) {
                System.out.println("Input error,please input the correct
name and retry again.");
            }
            else {
                this.ages=_ages;
                break;
            }
        }
    }
    public void SaveToFile(String filePath)  {
        if(!filePath.isEmpty()) {
            try {
                FileWriter f = new FileWriter(filePath,true);
                String outPut="ID: "+this.ID+" name: "+this.name+" ages:
"+this.ages+"\n";
                f.write(outPut); //Written into the buffer, not yet written
to the file
                f.flush(); //Flush the buffer,written to the file
                f.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
        else {
            System.out.println("The save file 's path input error!");
        }
    }
}
```

【实验结果】

请见下页：



【实验体会】

写这个实验有点花费时间，不过还是老师的认真教导，以及我的朋友和学长几人帮助下完成了，不过我觉得我写的可能仍然不好，我是初学，基础还是不够牢，还有用的写的太少了，我还需要多多加油。

另外那些参考连接都是写的很不错的，我从上面也学了很多，我觉得有必要收藏一下，需要的时候再查。

最后一个实验，干！