|  |  |
| --- | --- |
| haut | 河南工业大学信息科学与工程学院 |

**Java编程基础实验指导书**

|  |  |
| --- | --- |
| 课程名称： | Java编程基础 |
| 课程代码： | 51610951 |
| 课程性质： | 必修 |
| 实验学时： | 14学时 |
| 授课对象： | 软件工程 |
| 开课时间： | 第四学期 |

实验三 输入输出机制

实验日期： 2019 年 5 月 17 日 班级： 软件1702

学号（后四位）： 0224 姓名： 刘文博 成绩：

1. 实验目的

1.了解File类和常用方法；

2.掌握字节输入流和字节输出流进行文件的操作；

3.掌握字符输入流和字符输出流进行文件的操作；

4.掌握数据输入流和数据输出流进行数据的写入和读取；

5.掌握通过对象输入流和对象输出流进行对象的写入和读取。

二、支撑的课程目标

本实验项目可以支撑“课程目标1：能根据软件工程领域的复杂工程问题，进行研究，运用Java的编程思想设计项目开发方案，并按照合理步骤和软件工程流程进行方案的实施”、“课程目标2：掌握Java语言的基本特点、语法和面向对象的编程思想，能应用Java语言的编程原理、方法和技术，针对软件工程领域中复杂工程问题的设计解决方案要求进行软件开发”。

本实验通过“任务驱动”方法，使学生根据软件工程中项目对输入输出的需求，设计方案，通过Java输入输出的相关知识点，合理设计输入输出流，并进行满足任务需要的输入输出开发。

1. 实验内容

编写能够满足如下条件的程序（1和2题选做一题，5题选做）：

## 1 递归遍历目录，显示其中的文件名和目录名。若为文件，则直接显示文件的大小；若为目录，则显示目录中的文件和子目录。注意显示文件大小的单位（KB或MB）。

**代码：**

import java.io.File;

public class FileDemo {

public static void main(String[] args) {

readFile("src");

}

public static void readFile(String filepat){

final String filepath = filepat;

File file = new File(filepath);

String []str = file.list();

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

File file1 = new File(filepath + "/"+str[i]);

if(file1.isFile()){

System.out.println(str[i] + " -文件大小：" + file1.length()/1024 + "kb");

}

else if(file1.isDirectory()){

System.out.println("");

System.out.println(str[i] + " -文件夹" );

readFile(filepath + "/" + str[i]);

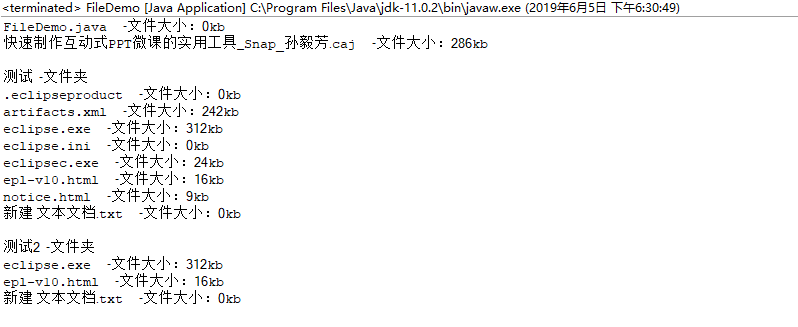
}

}

}

}

**实验截图：**



## 2 列出指定目录下包含某一指定字符串的所有文件名。

import java.io.File;

import java.io.FilenameFilter;

public class MyFileFliter {

public static void main(String[] args) {

String filepath = "F:/QQ下载";

File file = new File(filepath);

String [] str = file.list();

System.out.println("筛选前：");

for(String s : str) {

System.out.println(s);

}

String []str1 = file.list(new Myfilefliter("docx"));

System.out.println("筛选后：");

for(String s :str1) {

System.out.println(s);

}

}

static class Myfilefliter implements FilenameFilter{

private String dex;

public Myfilefliter(String dex) {

super();

this.dex = dex;

}

@Override

public boolean accept(File dir, String name) {

if(name.endsWith(this.dex)) {

return true;

}

return false;

}

}

}

**实验截图：**



## 3 选择输入流和输出流，实现目录拷贝。注意目的地的文件名

**代码：**

import java.io.BufferedInputStream;

import java.io.BufferedOutputStream;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.InputStream;

import java.io.OutputStream;

import javax.swing.JFileChooser;

public class MyCopy {

private File file;

private String fFilePath;

private String tFilePath;

public MyCopy(File file) {

super();

this.file = file;

fFilePath = this.file.getAbsolutePath();

}

public MyCopy() {

super();

}

public static File choicefFile() {

JFileChooser jc = new JFileChooser();

jc.setDialogTitle("选择要复制的文件或文件夹");

jc.setFileSelectionMode(JFileChooser.FILES\_AND\_DIRECTORIES);

jc.setCurrentDirectory(new File("c:/"));

int value = jc.showSaveDialog(null);

if (value == JFileChooser.APPROVE\_OPTION) {

return jc.getSelectedFile();

} else {

System.err.println("未选择文件或文件夹！");

return null;

}

}

public static String choicetFileDir(){

JFileChooser jc = new JFileChooser();

jc.setDialogTitle("选择目标文件夹");

jc.setFileSelectionMode(JFileChooser.DIRECTORIES\_ONLY);

jc.setCurrentDirectory(new File("c:/"));

int value1 = jc.showSaveDialog(null);

if (value1 == JFileChooser.APPROVE\_OPTION) {

return jc.getSelectedFile().getAbsolutePath();

} else {

System.err.println("未选择文件夹！");

return null;

}

}

public boolean copy(String tFilepath) {

this.tFilePath = tFilepath;

if(this.file.getAbsolutePath().equals(this.tFilePath)) {

System.err.println("相同目录无法复制！");

return false;

}

if (!this.file.exists()) {

System.err.println("文件不存在！");

return false;

} else if (this.file.isDirectory()) {

File file1 = new File(this.tFilePath + "/" + this.file.getName()); //目标文件

file1.mkdir(); //生成目标目录一级文件夹

String [] str = this.file.list();

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

File file2 = new File(this.fFilePath + "/" + str[i]);

String s1 = tFilePath + "/" + file.getName();

MyCopy mycopy = new MyCopy(file2);

mycopy.copy(s1);

}

return true;

} else if (this.file.isFile()) {

String tFilePath = this.tFilePath;

this.tFilePath = tFilePath + "/" + this.file.getName();

byte[] b = new byte[1024];

int num = 0;

try (InputStream in = new FileInputStream(this.file);

BufferedInputStream bin = new BufferedInputStream(in);

OutputStream out = new FileOutputStream(this.tFilePath);

BufferedOutputStream bout = new BufferedOutputStream(out)) {

while ((num = bin.read(b)) != -1) {

bout.write(b);

}

return true;

} catch (FileNotFoundException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

return false;

}

return false;

}

public void show() {

System.out.println("From:" + this.file.getAbsolutePath());

System.out.println("To:" + this.tFilePath);

}

}

import java.io.File;

public class MyCopyTest {

public static void main(String[] args) {

MyCopy c = new MyCopy(MyCopy.choicefFile());

long begantime ;

long endtime;

begantime = System.currentTimeMillis();

if(c.copy(MyCopy.choicetFileDir())) {

endtime = System.currentTimeMillis();

System.out.println("用时：" + (endtime - begantime) + "ms");

c.show();

}else {

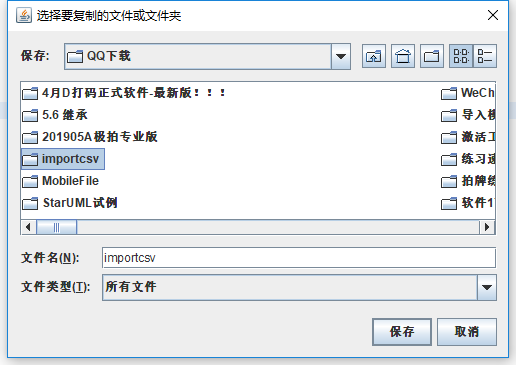
System.out.println("复制失败！！！");

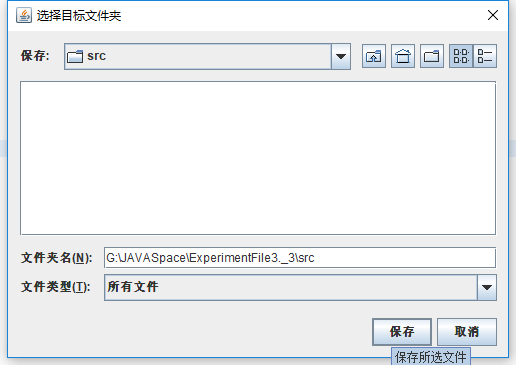
}

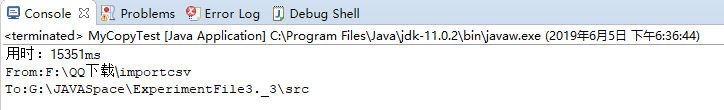
}

}

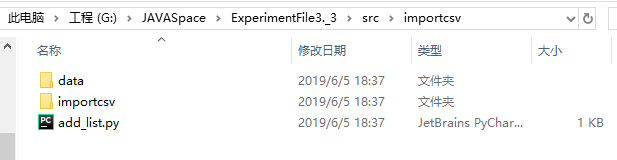
**实验截图：**











## 4 根据上学期自己在面向对象课程设计的题目，例如学生成绩管理系统，实现学生对象的输入输出，从控制台输入Student类的信息，包括学号、姓名、年龄，如输入错误，提示用户重新输入；实现保存对象到文件，能从文件中读出对象并显示。

**代码：**

**import** java.io.\*;

**public** **class** ObjFileRW{

File file;

**public** ObjFileRW(File file) {

**super**();

**this**.file = file;

}

**public** ObjFileRW(String filepath) {

**this**.file = **new** File(filepath);

}

**public** Object read() {

**try** (ObjectInputStream oin = **new** ObjectInputStream(**new** FileInputStream(**this**.file))){

**try** {

**return** oin.readObject();

} **catch** (ClassNotFoundException e) {

e.printStackTrace();

}

} **catch** (FileNotFoundException e) {

e.printStackTrace();

} **catch** (IOException e) {

e.printStackTrace();

}

**return** **null**;

}

**public** **void** write(Object obj) {

**try** (ObjectOutputStream oout = **new** ObjectOutputStream(**new** FileOutputStream(**this**.file))){

oout.writeObject(obj);

} **catch** (FileNotFoundException e) {

e.printStackTrace();

} **catch** (IOException e) {

e.printStackTrace();

}

}

}

**import** java.io.\*;

**public** **class** ObjTestDemo {

**public** **static** **void** main(String[] args) {

File file = **new** File("G:/JAVASpace/ExperimentFile3.\_4/src/data/stuData.dat");

ObjFileRW of = **new** ObjFileRW(file);

Student stu = **new** Student();

stu.inPutStuInof();

of.write(stu);

Student stu1 = (Student)of.read();

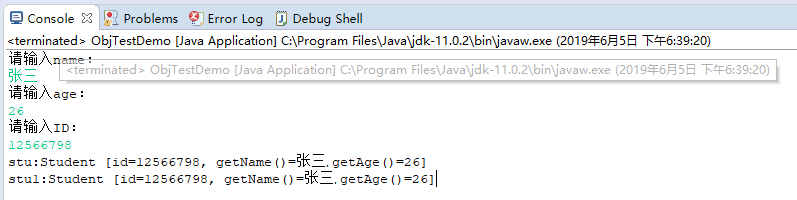
System.***out***.println("stu:" + stu.toString());

System.***out***.println("stu1:" + stu1.toString());

}

}

**实验截图：**



5 选择输入和输出流，读入二维数组，保存为文件，并实现从文件读出相应的数据。

四 在本次实验中的问题和解决方法

五 心得体会

复制文件要考虑文件重复的问题！！！