学号 E21514033 专业 软件工程 姓名 何铭春

实验日期 **5.17**  教师签字 成绩

实验报告

【实验名称】 输入输出流实验

【实验目的】

1. 掌握输入输出基本操作
2. 了解Java基本类型字节数

【实验原理】

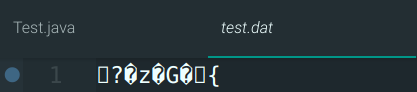
1. 多种输入输出流之间的关系
2. 能够自行使用对象流

【实验内容】

1. 计算输出文件字节数

12字节=4字节+8字节

截图：



1. 输入五个数到data.txt

源代码：

import java.io.\*;

import java.util.\*;

public class Test{

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

System.out.println("请输入5位学生的成绩：");

Scanner in = new Scanner(System.in);

int[] hmc = new int[5];

for (int i=0; i<5; i++) {

hmc[i] = in.nextInt();

}

in.close();

FileOutputStream fout = new FileOutputStream("data.txt");

DataOutputStream dout = new DataOutputStream(fout);

for (int i=0; i<5; i++) {

dout.writeInt(hmc[i]);

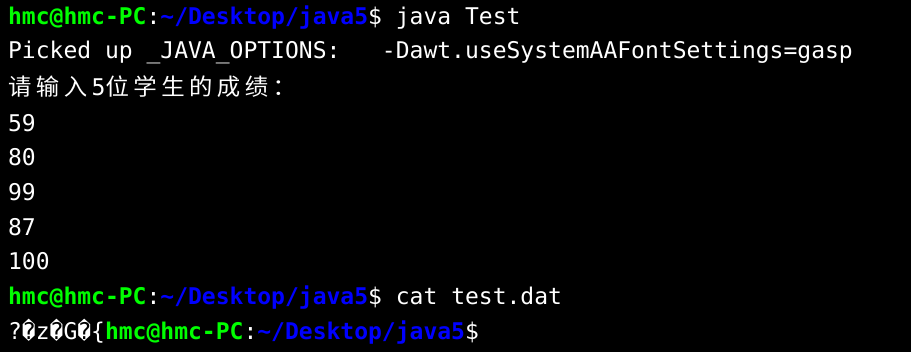
}

dout.close();

}

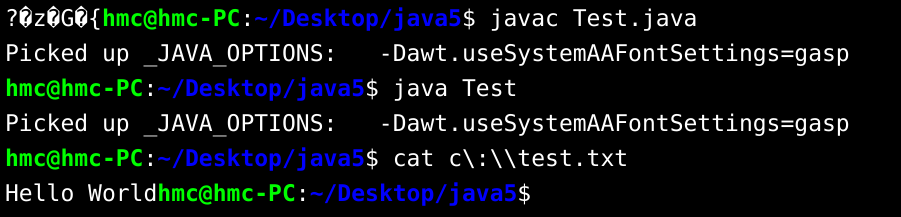
}

截图：



1. 测试是否写入文件成功

写入成功，但是路径错误



和同学对比发现，我在代码后添加了bw.close()，使得写入成功。

在无关闭的情况下，写入失败。

1. 显示当前目录下所有Java文件内容写入collection.txt并加上行号

源代码：

import java.io.\*;

import java.util.\*;

public class Hmc {

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

File dir = new File(".");

FilenameFilter fileAccept = new FilenameFilter(){

@Override

public boolean accept(File dir, String name) {

String temp = name.substring(name.length()-5,name.length());

if (temp.equals(".java")) {

return true;

}

return false;

}

};

String[] fileName = dir.list(fileAccept);

FileWriter out = new FileWriter("collection.txt");

FileReader in = null;

int ch;

for (String name : fileName ) {

in = new FileReader(name);

while ((ch = in.read())!=-1) {

out.write(ch);

}

}

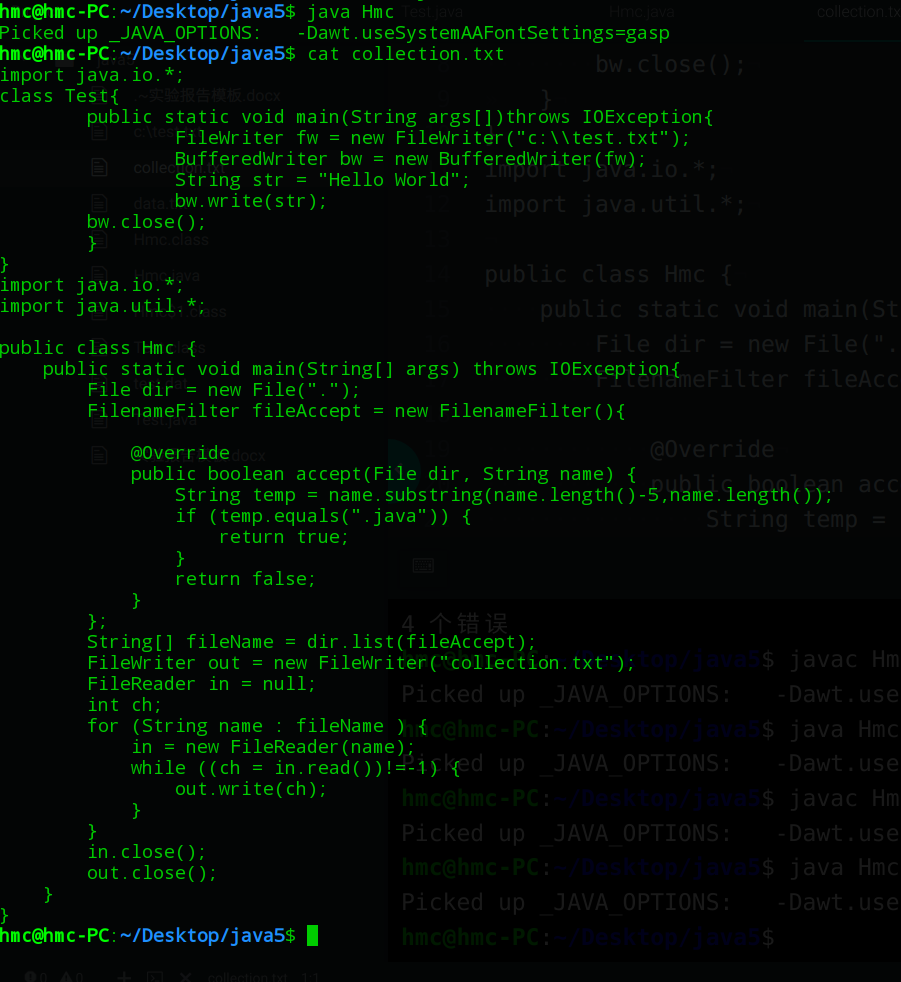
in.close();

out.close();

}

}

实验截图：



1. 对象流

源代码：

import java.io.\*;

public class Spring {

public static void main(String[] args) {

Student hmc = new Student();

hmc.setName("hmc");

hmc.setAge(21);

hmc.setAddress("合肥",231500);

File file = new File("student.txt");

try (FileOutputStream fileOut = new FileOutputStream(file);

ObjectOutputStream objectOut = new ObjectOutputStream(fileOut);

FileInputStream fileIn = new FileInputStream(file);

ObjectInputStream objectIn = new ObjectInputStream(fileIn)){

objectOut.writeObject(hmc);

objectOut.flush();

Student spring = (Student)objectIn.readObject();

System.out.println("姓名："+spring.getName());

System.out.println("年龄："+spring.getAge());

System.out.println("地址："+spring.getAddressName());

System.out.println("邮编："+spring.getZipCode());

}

catch (ClassNotFoundException e) {

System.out.println("不能读出对象");

}

catch (IOException e) {

System.out.println(e);

}

}

}

class Student implements Serializable{

String name;

int age;

Address addr = new Address();

public void setName(String name){

this.name = name;

}

public String getName(){

return this.name;

}

public void setAge(int age){

this.age = age;

}

public int getAge(){

return this.age;

}

public void setAddress(String addressName,int zipCode){

this.addr.setAddressName(addressName);

this.addr.setZipCode(zipCode);

}

public String getAddressName(){

return this.addr.getAddressName();

}

public int getZipCode(){

return this.addr.getZipCode();

}

}

class Address implements Serializable{

String addressName;

int zipCode;

public void setAddressName(String addressName){

this.addressName = addressName;

}

public String getAddressName(){

return this.addressName;

}

public void setZipCode(int zipCode){

this.zipCode = zipCode;

}

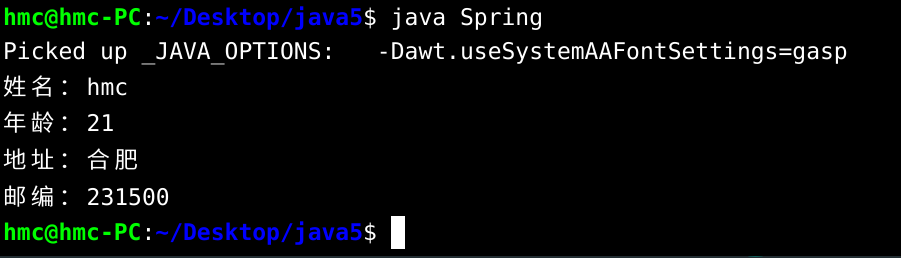
public int getZipCode(){

return this.zipCode;

}

}

实验截图：



【小结或讨论】

1. 输出流需要关闭或者刷新才能输出
2. FileNameFilter需要自行重写accept函数来实现文件名筛选
3. Java writeInt等函数莫名编码错误。