# 实验8： 二进制I/O

## 实验目的和要求

理解文本I/O和二进制I/O的不同

掌握FileInputStream和FileOutputStream的使用方法

掌握FilterInputStream和FilterOutputStream的使用方法

掌握BufferInputStream使用方法

掌握BufferOutputStream使用方法

## 实验题目

### P606 17.2

原题：（创建二进制数据文件）编写一个文件，如果文件Exercise17\_02.dat不存在，就创建一个名为Exercise17\_02.dat的文件。向这个文件追加新数据。使用二进制I/O将100个随机生成的整数写入这个文件。

源代码：

**package** zuoye;

**import** java.io.DataInputStream;

**import** java.io.DataOutputStream;

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**public** **class** Test1 {

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

// **TODO** Auto-generated method stub

File file = **new** File("e://Exercise17\_02.dat");

**if**(file.exists())

{

System.***out***.println("文件已存在！");

}

**int**[] a = **new** **int**[100];

**try** {

DataOutputStream dos = **new** DataOutputStream(**new** FileOutputStream(file));

**for**(**int** i=0; i<100; i++)

{

a[i] = (**int**)(Math.*random*()\*10);

dos.writeInt(a[i]);

}

dos.close();

} **catch** (FileNotFoundException e) {

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**int**[] b = **new** **int**[100];

**try** {

DataInputStream dis = **new** DataInputStream(**new** FileInputStream(file));

**for**(**int** i=0; i<100; i++)

{

b[i] = dis.readInt();

System.***out***.print(b[i]+" ");

}

dis.close();

} **catch** (FileNotFoundException e) {

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

结果及截图：



### P606 17.3

原题：（对二进制数据文件中的所有的数据求和）假设已经使用DataOutputStream中的writeInt（int）方法创建了一个名为Exercise17\_03.dat的二进制数据文件，文件包括数目不确定的整数，编写一个程序来计算这些整数的和。

源代码：

**package** zuoye;

**import** java.io.DataInputStream;

**import** java.io.DataOutputStream;

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**public** **class** Test2 {

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

// **TODO** Auto-generated method stub

File file = **new** File("e://Exercise17\_02.dat");

**int**[] b = **new** **int**[100];

**int** sum = 0;

**try** {

DataInputStream dis = **new** DataInputStream(**new** FileInputStream(file));

**for**(**int** i=0; i<100; i++)

{

b[i] = dis.readInt();

sum += b[i];

}

System.***out***.println("sum: "+sum);

dis.close();

} **catch** (FileNotFoundException e) {

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

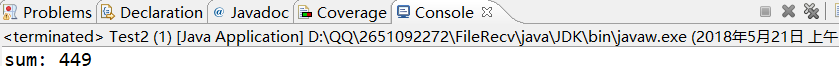
e.printStackTrace();

}

}

}

结果及截图：



### P606 17.4

原题：（将文本文件转换为UTF格式）编写一个程序，每次从文本文件中读取多行字符，并将这些行字符以UTF-8字符串格式写入一个二进制文件中。显示文本文件和二进制文件的大小。使用下面的命令运行这个程序：

Java Exercise17\_04 Welcome.Java Welcome.utf

源代码：

**package** zuoye;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.DataOutputStream;

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**import** java.io.OutputStreamWriter;

**import** java.io.PrintWriter;

**import** java.util.Scanner;

**public** **class** Test3 {

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

// **TODO** Auto-generated method stub

File a = **new** File("e://t1.txt");

File b = **new** File("e://t2.txt");

**try** {

Change.*changeTo*(a, b);

}

**catch**(IOException e)

{

// **TODO** Auto-generated catch block

e.printStackTrace();

}

System.***out***.println(a.getName()+" : "+a.length());

System.***out***.println(b.getName()+" : "+b.length());

}

}

**class** Change

{

**public** **static** **void** changeTo(File a,File b)**throws** IOException

{

StringBuilder s = **new** StringBuilder();

BufferedWriter bw = **new** BufferedWriter(**new** OutputStreamWriter(**new** FileOutputStream(a),"utf-8"));

bw.write("你好");

bw.newLine();

bw.write("java");

bw.newLine();

bw.write("我是张三");

bw.close();

BufferedReader br = **new** BufferedReader(**new** InputStreamReader(**new** FileInputStream(a),"utf-8"));

**while**(**true**)

{

s.append(br.readLine());

**if**(br.read()==-1)

**break**;

}

br.close();

DataOutputStream dos = **new** DataOutputStream(**new** FileOutputStream(b));

dos.writeUTF(s.toString());

dos.close();

}

}

结果及截图：



### P607 17.14

原题：（加密文件）通过文件中的每个字节加5来对文件进行加密。编写一个程序，提示用户输入一个输入文件名和一个输出文件名，然后将输入文件的加密版本存入到输出文件中。

源代码：

**package** zuoye;

**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.util.Scanner;

**public** **class** Test4 {

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

// **TODO** Auto-generated method stub

Scanner in = **new** Scanner(System.***in***);

System.***out***.println("请输入源文件名：");

String s1 = in.nextLine();

System.***out***.println("请输入目标文件名：");

String s2 = in.nextLine();

File f1 = **new** File("e://"+s1);

File f2 = **new** File("e://"+s2);

**byte**[] b = **new** **byte**[1024\*1024];

**try** {

BufferedInputStream bis = **new** BufferedInputStream(**new** FileInputStream(f1));

bis.read(b);

bis.close();

} **catch** (FileNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**try** {

BufferedOutputStream bos = **new** BufferedOutputStream(**new** FileOutputStream(f2));

**for**(**int** i=0; i<b.length; i++)

{

b[i] = (**byte**) (b[i] + 5);

}

bos.write(b);

bos.close();

} **catch** (FileNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

### P608 17.15

原题：（解密文件）假设文件是用编程练习题17.14中的编码方案进行加密的。编写一个程序，解码这个加密文件。程序应该提示用户输入一个输入文件名和一个输出文件名，然后将输入文件的加密版本存入到输出文件中。

源代码：

**package** zuoye;

**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.util.Scanner;

**public** **class** Test5 {

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

// **TODO** Auto-generated method stub

Scanner in = **new** Scanner(System.***in***);

System.***out***.println("请输入源文件名：");

String s1 = in.nextLine();

System.***out***.println("请输入目标文件名：");

String s2 = in.nextLine();

File f1 = **new** File("e://"+s1);

File f2 = **new** File("e://"+s2);

**byte**[] b = **new** **byte**[1024\*1024];

**try** {

BufferedInputStream bis = **new** BufferedInputStream(**new** FileInputStream(f1));

bis.read(b);

bis.close();

} **catch** (FileNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**try** {

BufferedOutputStream bos = **new** BufferedOutputStream(**new** FileOutputStream(f2));

**for**(**int** i=0; i<b.length; i++)

{

b[i] = (**byte**) (b[i] - 5);

}

bos.write(b);

bos.close();

} **catch** (FileNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

### P608 17.16

原题：（字符的频率）编写一个程序，提示用户输入一个ASCII文本文件名，然后显示文件中每个字符出现的频率。

源代码：

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class Test5 {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.println("请输入一个ASCALL文本文件名：");

String s0 = in.nextLine();

StringBuilder s1 = new StringBuilder();

int n = 0;

File f = new File(s0);

try {

Scanner sc = new Scanner(f);

while(sc.hasNext())

{

s1.append(sc.nextLine());

}

} catch (FileNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

for(char c = 0; c<256; c++)

{

n = 0;

for(int j=0;j<s1.length();j++)

{

if(s1.charAt(j)==c)

{

n++;

}

}

System.out.print(c+" : "+n+" ");

if(c%10==0)

System.out.println();

}

}

}

结果及截图：

