

实验八文件IO处理实验.xls

返回

 姓名：李全欣 班级：计算机2018级4-7 成绩：100分

一.简答题 (共3题,100.0分)

1

- 一、实验目的
1. 理解基于IO流文件操作的过程及意义

2. 掌握基于字节数组缓冲区，实现IO操作方法

3. 掌握基于NIO文件目录的创建方法

4. 掌握基于NIO文件创建/删除的创建方法
- 二、实验环境
- (5.0分)

正确答案：

我的答案：  
操作系统： windows 10 64位  
JDK版本 jdk-11.0.6  
编译器： eclipse-inst-win64

2

三、实验内容、代码及结果截图

所有异常，全部在调用方法内捕获处理

将以下IOTest类的全部方法实现

```
public class IOTest {  
  
    public static void main(String[] args) {  
  
        String fileName = "C:/example/from.txt";  
  
  
        System.out.println("----- 创建文件 -----");  
  
        createFile(fileName);  
  
  
        System.out.println("----- 将字符串写入文件 -----");  
  
        // \r\n在txt文本中换行  
  
        String str =  
  
            "白日依山尽\r\n" +  
  
            "黄河入海流\r\n" +  
  
            "欲穷千里目\r\n" +  
  
            "更上一层楼\r\n";  
  
    }  
}
```

```
writeToFile( fileName,str);

System.out.println("----- 基于基本IO流实现文件的复制 -----");
String toFile = "C:/example/to.txt";
copyByIO(fileName, toFile);

System.out.println("----- 基于NIO实现文件的复制 -----");
String toFile2 = "C:/example/nio/to.txt";
copyByIO(fileName, toFile2);

System.out.println("----- 删除指定文件 -----");
deleteFile(toFile);
System.out.println("----- 遍历指定目录文件 -----");
String dir = "C:/example";
walkDirectories(dir);
}

/**
 * 基于指定文件名称创建目录及文件
 * 如果文件已经存在，则忽略
 *
 * @param fileName
 */
private static void createFile(String fileName) {

}

/**
 * 提示：文件以字节操作，因此可以
 * 字符串，转字节数组，直接基于Files写入文件
 *
 * @param fileName
 * @param content
 */
private static void writeToFile(String fileName, String content) {

}

/**
 * 基于基本IO，以及字节数组缓冲区，复制文件
 * 打印显示循环读写循环次数
 *
 * @param sourceFile
 * @param targetFile
 */
```

```
private static void copyByIO(String sourceFile, String targetFile) {

}

/**
 * 基于NIO, 实现文件的复制
 *
 * @param sourceFile
 * @param targetFile
 */
private static void copyByNIO(String sourceFile, String targetFile) {

}

/**
 * 删除文件
 *
 * @param fileName
 */
private static void deleteFile(String fileName) {

}

/**
 * 遍历打印指定目录下全部目录/文件名称
 *
 * @param dir
 */
private static void walkDirectories(String dir) {

}
}
```

(80.0分)

正确答案:

我的答案:

```
private static void createFile(String fileName) throws IOException {
    Path file=Paths.get(fileName);
    Files.createDirectories(file.getParent());
    if(!Files.exists(file)) {
        Files.createFile(file);
        System.out.println("文件创建成功!");
    }
    else {
        System.out.println("文件已存在");
    }
}
```

Problems Javadoc Declaration Search Con

<terminated> IOTest (1) [Java Application] C:\Program File:

----- 创建文件 -----  
文件创建成功!

```
private static void writeToFIle(String fileName, String content) throws FileNotFoundException, IOException{
    byte[] buffer=content.getBytes();
    try(FileOutputStream out=new FileOutputStream(fileName)){
        out.write(buffer);
    }
}
```

----- 将字符串写入文件 -----  
写入成功!

from.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

白日依山尽  
黄河入海流  
欲穷千里目  
更上一层楼

```
private static void copyByIO(String sourceFile, String targetFile) throws IOException {
    try(FileInputStream in=new FileInputStream(sourceFile);
        FileOutputStream out=new FileOutputStream(targetFile)){
        int len=0;
        int c=0;
        while((c=in.read())!=-1) {
            out.write(c);
            len++;
        }
        System.out.println("循环读写次数:"+len); }
}
```

----- 基于基本IO流实现文件的复制 -----  
循环读写次数:48

```
private static void copyByNIO(String sourceFile, String targetFile) throws IOException {
    Path p1=Paths.get(sourceFile);
    Path p2=Paths.get(targetFile);
    Files.createDirectories(p2.getParent());
    if(!Files.exists(p2)) {
        Files.createFile(p2);
    }
    Files.copy(p1, p2, StandardCopyOption.REPLACE_EXISTING);
    System.out.println("复制成功!");
}
```

<terminated> IOTest (1) [Java Application] C:\P

----- 基于NIO实现文件的复制 -----  
复制成功!

```
private static void deleteFile(String fileName) throws IOException {
    Path file=Paths.get(fileName);
    Files.deleteIfExists(file);
}
```

----- 删除指定文件 -----  
删除成功!

```
private static void walkDirectories(String dir) throws IOException {
    Path p=Paths.get(dir);
    Files.walk(p).forEach(System.out::println);
}
```

<terminated> IOTest (1) [Java Application] C:\Program Fil

----- 遍历指定目录文件 -----

C:\example  
C:\example\from.txt  
C:\example\nio  
C:\example\nio\to.txt

3

## 四、实验过程分析与讨论

(15.0分)

正确答案:

我的答案:

IO是面向流的，NIO是面向缓冲区的，使用NIO更加灵活