

## 1, *fileoutputstream*

```
//创建一个file对象，表明要写入的文件的位置
//输出的物理文件可以不存在，当执行过程中，若不存在，会自动的创建
//若存在则覆盖原有的文件
File f1 = new File("e://hhllo.txt");
//创建一个fileoutputstream的对象，将file的对象作为参数传递给
fileoutputstream
//的构造器中
FileOutputStream fos = null;
try {
    fos = new FileOutputStream(f1);
    //写入操作
    fos.write(new String("linmingjun,nishigezhu!").getBytes());
} catch (Exception e) {
    // TODO: handle exception
    e.printStackTrace();
}
finally {
    //关闭输出流
    if (fos != null) {
        try {
            fos.close();
            System.out.println("fhjfjhdf");
        } catch (Exception e2) {
            // TODO: handle exception
            e2.printStackTrace();
        }
    }
}
```

## 2, 文件的复制

//实现文件复制的方法

@Test

```

public void copyFile(String s1,String s2){
    //从硬盘中读取文件，并写入到另一个位置，（相当于文件的复制）
    //提供读入和写出的文件
    //s1表示源文件路径，s2表示目标文件的路径
    File f1 = new File(s1);
    File f2 = new File(s2);
    //提供相应的流
    FileInputStream fis = null;
    FileOutputStream fos =null;
    try {
        fis = new FileInputStream(f1);
        fos = new FileOutputStream(f2);
        //实现文件的复制
        byte []b = new byte[1024];
        int len;
        while((len=fis.read(b)) != -1){
            fos.write(b, 0, len);
        }
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
    finally {
        if (fis != null) {
            try {
                fos.close();
            } catch (IOException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
        }
        if (fis != null) {
            try {
                fis.close();
            } catch (IOException e) {

```

```
// TODO Auto-generated catch block  
e.printStackTrace();
```

```
}
```

```
}
```

```
}
```

```
}
```