

JAVA资格认证:文件操作之File类使用Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/571/2021\\_2022\\_JAVA\\_E8\\_B5\\_84\\_E6\\_A0\\_BC\\_c104\\_571340.htm](https://www.100test.com/kao_ti2020/571/2021_2022_JAVA_E8_B5_84_E6_A0_BC_c104_571340.htm)

I/O类使用 由于在IO操作中，需要使用的数据源有很多，作为一个IO技术的初学者，从读写文件开始学习IO技术是一个比较好的选择。因为文件是一种常见的数据源，而且读写文件也是程序员进行IO编程的一个基本能力。本章IO类的使用就从读写文件开始。

### 1 文件操作

文件（File）是最常见的数据源之一，在程序中经常需要将数据存储到文件中，例如图片文件、声音文件等数据文件，也经常需要根据需要从指定的文件中进行数据的读取。当然，在实际使用时，文件都包含一个的格式，这个格式需要程序员根据需要进行设计，读取已有的文件时也需要熟悉对应的文件格式，才能把数据从文件中正确的读取出来。文件的存储介质有很多，例如硬盘、光盘和U盘等，由于IO类设计时，从数据源转换为流对象的操作由API实现了，所以存储介质的不同对于程序员来说是透明的，和实际编写代码无关。

#### 1.1 文件的概念

文件是计算机中一种基本的数据存储形式，在实际存储数据时，如果对于数据的读写速度要求不是很高，存储的数据量不是很大时，使用文件作为一种持久数据存储的方式是比较好的选择。存储在文件内部的数据和内存中的数据不同，存储在文件中的数据是一种“持久存储”，也就是当程序退出或计算机关机以后，数据还是存在的，而内存内部的数据在程序退出或计算机关机以后，数据就丢失了。在不同的存储介质中，文件中的数据都是以一定的顺序依次存储起来，在实际读取时由硬件以及操作系统完成对于

数据的控制，保证程序读取到的数据和存储的顺序保持一致。每个文件以一个文件路径和文件名称进行表示，在需要访问该文件的时，只需要知道该文件的路径以及文件的全名即可。在不同的操作系统环境下，文件路径的表示形式是不一样的，例如在Windows操作系统中一般的表示形式为C:\windows\system，而Unix上的表示形式为/user/my。所以如果需要让Java程序能够在不同的操作系统下运行，书写文件路径时还需要比较注意。

### 1.1.1 绝对路径和相对路径

绝对路径是指书写文件的完整路径，例如d:\java\Hello.java，该路径中包含文件的完整路径d:\java以及文件的全名Hello.java。使用该路径可以唯一的找到一个文件，不会产生歧义。但是使用绝对路径在表示文件时，受到的限制很大，且不能在不同的操作系统下运行，因为不同操作系统下绝对路径的表达形式存在不同。

相对路径是指书写文件的部分路径，例如\test\Hello.java，该路径中只包含文件的部分路径\test和文件的全名Hello.java，部分路径是指当前路径下的子路径，例如当前程序在d:\abc下运行，则该文件的完整路径就是d:\abc\test。使用这种形式，可以更加通用的代表文件的位置，使得文件路径产生一定的灵活性。

在Eclipse项目中运行程序时，当前路径是项目的根目录，例如工作空间存储在d:\javaproject，当前项目名称是Test，则当前路径是：d:\javaproject\Test。在控制台下面运行程序时，当前路径是class文件所在的目录，如果class文件包含包名，则以该class文件最顶层的包名作为当前路径。另外在Java语言的代码内部书写文件路径时，需要注意大小写，大小写需要保持一致，路径中的文件夹名称区分大小写。由于‘\’是Java语言中的特

殊字符，所以在代码内部书写文件路径时，例如代表“c : \test\java\Hello.java”时，需要书写成“c : \\test\\java\\Hello.java”或“c : /test/java/Hello.java”，这些都需要在代码中注意。

### 1.1.2 文件名称

文件名称一般采用“文件名.后缀名”的形式进行命名，其中“文件名”用来表示文件的作用，而使用后缀名来表示文件的类型，这是当前操作系统中常见的一种形式，例如“readme.txt”文件，其中readme代表该文件时说明文件，而txt后缀名代表文件时文本文件类型，在操作系统中，还会自动将特定格式的后缀名和对应的程序关联，在双击该文件时使用特定的程序打开。其实在文件名称只是一个标示，和实际存储的文件内容没有必然的联系，只是使用这种方式方便文件的使用。在程序中需要存储数据时，如果自己设计了特定的文件格式，则可以自定义文件的后缀名，来标示自己的文件类型。和文件路径一样，在Java代码内部书写文件名称时也区分大小写，文件名称的大小写必须和操作系统中的大小写保持一致。另外，在书写文件名称时不要忘记书写文件的后缀名。

## 1.2 File类

为了很方便的代表文件的概念，以及存储一些对于文件的基本操作，在java.io包中设计了一个专门的类File类。在File类中包含了大部分和文件操作的功能方法，该类的对象可以代表一个具体的文件或文件夹，所以以前曾有人建议将该类的类名修改成FilePath，因为该类也可以代表一个文件夹，更准确的说是可以代表一个文件路径。下面介绍一下File类的基本使用。

### 1、File对象代表文件路径

File类的对象可以代表一个具体的文件路径，在实际代表时，可以使用绝对路径也可以使用相对路径。下面是创建的文件对象示例。

```
public File (String
```

pathname) 该示例中使用一个文件路径表示一个File类的对象，例如：`File f1 = new File( " d : \\test\\1.txt " ) ; File f2 = new File( " 1.txt " ) ; File f3 = new File( " e : \\abc " ) ;` 这里的f1和f2对象分别代表一个文件，f1是绝对路径，而f2是相对路径，f3则代表一个文件夹，文件夹也是文件路径的一种。

`public File( String parent , String child )` 也可以使用父路径和子路径结合，实现代表文件路径，例如：`File f4 = new File( " d : \\test\\" , " 1.txt " ) ;` 这样代表的文件路径是：`d : \\test\\1.txt`.

## 2、File类常用方法

File类中包含了很多获得文件或文件夹属性的方法，使用起来比较方便，下面将常见的方法介绍如下：

- a、`createNewFile`方法 `public boolean createNewFile ( ) throws IOException` 该方法的作用是创建指定的文件。该方法只能用于创建文件，不能用于创建文件夹，且文件路径中包含的文件夹必须存在。
- b、`delete`方法 `public boolean delete ( )` 该方法的作用是删除当前文件或文件夹。如果删除的是文件夹，则该文件夹必须为空。如果需要删除一个非空的文件夹，则需要首先删除该文件夹内部的每个文件和文件夹，然后在可以删除，这个需要书写一定的逻辑代码实现。
- c、`exists`方法 `public boolean exists ( )` 该方法的作用是判断当前文件或文件夹是否存在。
- d、`getAbsolutePath`方法 `public String getAbsolutePath ( )` 该方法的作用是获得当前文件或文件夹的绝对路径。例如c : \\test\\1.t则返回c : \\test\\1.t.
- e、`getName`方法 `public String getName ( )` 该方法的作用是获得当前文件或文件夹的名称。例如c : \\test\\1.t，则返回1.t.
- f、`getParent`方法 `public String getParent ( )` 该方法的作用是获得当前路径中的父路径。例如c : \\test\\1.t则返回c : \\test.
- g

、 isDirectory方法 public boolean isDirectory ( ) 该方法的作用是判断当前File对象是否是目录。 h、 isFile方法 public boolean isFile ( ) 该方法的作用是判断当前File对象是否是文件。 i、 length方法 public long length ( ) 该方法的作用是返回文件存储时占用的字节数。该数值获得的是文件的实际大小，而不是文件在存储时占用的空间数。 j、 list方法 public String[] list ( ) 该方法的作用是返回当前文件夹下所有的文件名和文件夹名称。说明，该名称不是绝对路径。 k、 listFiles方法

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)