

文章

[姚鑫](#) · 七月 16, 2021 阅读大约需 4 分钟

第五章 使用文件

第五章 使用文件

使用文件

%Library.File类提供了几个类方法，允许对文件执行各种操作。

复制文件

若要复制文件，请使用CopyFile()方法，该方法返回一个布尔值来指示成功或失败。

此方法采用四个参数：

1. from 从—指定源文件的名称。
2. to至—指定目标文件的名称。
3. pDeleteBeforeCopy —指定在执行复制之前是否删除目标文件(如果存在)。默认值为0。
4. return 返回—输出参数。如果为负，则包含操作系统返回的错误代码，以防方法失败

下面的第一个示例将目录e:\temp中的文件old.txt复制到new.txt。第二个示例将相同的文件复制到默认目录中的new.txt。

```
DHC-APP>write ##class(%File).CopyFile("e:\temp\old.txt", "e:\temp\new.txt", 0, .return)
1
DHC-APP>write ##class(%File).CopyFile("e:\temp\old.txt", "new.txt", 0, .return)
1
```

最后一个示例失败，Windows错误代码为2，或“找不到文件”

```
DHC-APP>write ##class(%File).CopyFile("foo.txt", "new.txt", 0, .return)
0
DHC-APP>w return
-2
```

删除文件

要删除文件，请使用delete()方法，该方法成功时返回1，失败时返回0。这个方法需要两个参数。第一个参数是要删除的文件的名称。第二个参数是输出参数。如果为负，它包含操作系统返回的错误代码，以防方法失败。

在下面的第一个示例中，方法成功了。第二个示例失败，出现Windows错误代码2或“找不到文件”。

```
DHC-APP>write ##class(%File).Delete("e:\temp\myfile.txt", .return)
1
DHC-APP>write ##class(%File).Delete("e:\temp\myfile.txt", .return)
```

```
0
DHC-APP>w return
-2
```

要在删除文件时匹配通配符，请使用ComplexDelete()方法。第一个参数指定要删除的文件的名称。第二个参数是输出参数。如果为负，它包含操作系统返回的错误代码，以防方法失败。

下面的示例删除所有带有。e:/temp目录中的out扩展名。

```
DHC-APP>write ##class(%File).ComplexDelete("e:\temp\*.out", .return)
1
```

截断文件

要截断文件，请使用truncate()方法，该方法成功时返回1，失败时返回0。这个方法需要两个参数。第一个参数是要截断的文件的名称。第二个参数是输出参数。如果为负，它包含操作系统返回的错误代码，以防方法失败。

如果截断现有文件，方法会从文件中删除内容，但不会从文件系统中删除内容。如果截断不存在的文件，方法会创建一个新的空文件。

在下面的第一个示例中，方法成功了。第二个示例失败，Windows错误代码为5，或“访问被拒绝”

```
USER>write ##class(%File).Truncate("e:\temp\myfile.txt", .return)
1
USER>write ##class(%File).Truncate("e:\no access.txt", .return)
0
USER>write return
-5
```

重命名文件

若要重命名文件，请使用rename()方法，该方法成功时返回1，失败时返回0。这个方法需要三个参数。第一个参数是要重命名的文件的名称，第二个参数是新名称。第三个参数是输出参数。如果为负，它包含操作系统返回的错误代码，以防方法失败。

在下面的第一个示例中，方法成功了。第二个示例失败，错误代码为183，或者“当文件已经存在时，无法创建该文件。”

```
DHC-APP>write ##class(%File).Rename("e:\temp\oldname.txt", "e:\temp\newname.txt", .return)
1
DHC-APP>write ##class(%File).Rename("e:\temp\another.txt", "e:\temp\newname.txt", .return)
0
DHC-APP>write return
-2
```

使用此方法时，请小心指定路径，因为以下示例会将e:/temp/oldname.txt移动到默认目录，然后将其重命名为newname.txt。

```
DHC-APP>write ##class(%File).Rename("e:\temp\oldname.txt", "newname.txt", .return)
1
```

比较文件

若要比较两个文件，请使用Compare()方法，如果两个文件相同，则返回布尔值1，否则返回0。该方法没有用于返回系统错误代码的输出参数。

在下面的第一个示例中，两个文件是相同的，方法返回1。在第二个示例中，两个文件不同，因此方法返回0。

```
DHC-APP>write ##class(%File).Compare("e:\temp\old.txt", "e:\temp\new.txt")
1
DHC-APP>write ##class(%File).Compare("e:\temp\old.txt", "e:\temp\another.txt")
0
```

如果一个或两个文件都不存在，如下例所示，则该方法也返回0。

```
DHC-APP>write ##class(%File).Compare("foo.txt", "bar.txt")
0
DHC-APP>write ##class(%File).Exists("foo.txt")
0
```

生成临时文件

要生成临时文件，请使用TempFilename()方法，该方法返回临时文件的名称。这个方法需要三个参数。第一个参数是临时文件所需的文件扩展名。第二个是生成临时文件的目录。如果未提供，该方法将在操作系统提供的临时目录中生成文件。第三个参数是输出参数。如果为负，它包含操作系统返回的错误代码，以防方法失败。

Windows示例:

```
USER>write ##class(%File).TempFilename("txt")
C:\WINDOWS\TEMP\GATqk8a6.txt
USER>write ##class(%File).TempFilename("txt", "C:\temp")
C:\temp\WpSwuLlA.txt
```

Unix示例:

```
USER>write ##class(%File).TempFilename("", "", .return)
/tmp/filsfHGzc
USER>write ##class(%File).TempFilename("tmp", "/InterSystems/temp", .return)
/InterSystems/temp/file0tnuh.tmp
USER>write ##class(%File).TempFilename("", "/tmp1", .return)

USER>write return
-2
```

在上面的第三个示例中，目录不存在，该方法失败，系统错误代码为2，或“没有这样的文件或目录。”

[#Cache](#)

源

URL:

<https://cn.community.intersystems.com/post/%E7%AC%AC%E4%BA%94%E7%AB%A0-%E4%BD%BF%E7%94%A8%E6%96%87%E4%BB%B6>