



北京圣思园科技有限公司
<http://www.shengsiyuan.com>

主讲人：张龙

Java I/O系统

- 课程目标
 - 理解Java I/O系统
 - 熟练使用java.io包中的相关类与接口进行I/O编程
 - **掌握Java I/O的设计原则与使用的设计模式**



Java I/O系统

- 对程序设计者来说，设计一个令人满意的**I/O**（输入输出）系统，是件极艰巨的任务
 - 摘自《Thinking in Java》



File类

- 一个File类的对象，表示了磁盘上的文件或目录
- File类提供了与平台无关的方法来对磁盘上的文件或目录进行操作
- File类直接处理文件和文件系统。
- File类没有指定信息怎样从文件读取或向文件存储



File类

- **File**类描述了文件本身的属性
- **File**对象用来获取或处理与磁盘文件相关的信息，例如权限，时间，日期和目录路径
- **File**类还可以浏览子目录层次结构



File类

- java.io包中的File类提供了与具体平台无关的方式来描述目录和文件对象的属性功能。其中包含大量的方法可用来获取路径、目录和文件的相关信息，并对它们进行创建、删除、改名等管理工作。因为不同的系统平台，对文件路径的描述不尽相同。为做到平台无关，在Java语言中，使用抽象路径等概念。Java自动进行不同系统平台的文件路径描述与抽象文件路径之间的转换。
- File类的直接父类是Object类。



File类

- 下面的构造方法可以用来生成File 对象：
 - File(String directoryPath)
File(String directoryPath, String filename)
File(File dirObj, String filename)
- 这里，directoryPath是文件的路径名，filename 是文件名，dirObj 是一个指定目录的File 对象



举例说明:

```
File myFile = new File(" myfile. txt");  
File myDir = new File(" MyDocs");  
File myFile = new File( myDir, "myfile.  
txt");
```

这些构造方法取决于访问文件的方式。例如，若在应用程序里只用一个文件，第一种创建文件的结构是最容易的。但若在同一目录里打开数个文件，则后种方法更好一些。



File类

- 目录管理

- 目录操作的主要方法为:

- `public boolean mkdir()` **根据抽象路径名创建目录。**

- `public String[] list()` **返回抽象路径名表示路径中的文件名和目录名。**

- 文件管理

- 在进行文件操作时，常需要知道一个关于文件的信息。Java的File类提供了方法来操纵文件和获得一个文件的信息。另外，File类还可以对目录和文件进行删除、属性修改等管理工作



File类

- **File** 类定义了很多获取**File**对象标准属性的方法。例如**getName()**返回文件名，**getParent()**返回父目录名，**exists()**在文件存在的情况下返回**true**，反之返回**false**。然而**File**类是**不对称**的。说它不对称，意思是虽然存在允许验证一个简单文件对象属性的很多方法，但是没有相应的方法来改变这些属性



File类

- File类中的常用方法

String getName()

String getPath()

String getAbsolutePath()

String getParent()

boolean renameTo(File newName)

long length()

boolean delete()

boolean mkdir()

String[] list()



- File类中的常用方法
 - boolean exists()
 - boolean canWrite()
 - boolean canRead()
 - boolean isFile()
 - boolean isDirectory()



File类

- 目录是一个包含其他文件和路径列表的File类。
- 当你创建一个File对象且它是目录时，`isDirectory()`方法返回`true`。这种情况下，可以调用该对象的`list()`方法来提取该目录内部其他文件和目录的列表



File类

- 示例程序



- 使用FilenameFilter

- 希望能够限制由list()方法返回的文件数目，使它仅返回那些与一定的文件名方式或者过滤(filter)相匹配的文件。为达到这样的目的，必须使用list()的第二种形式 (方法重载)
- String[] list(FilenameFilter FFObj)
- 该形式中，FFObj是一个实现FilenameFilter接口的类的对象



File类

- **FilenameFilter**仅定义了一个方法，**accept()**。该方法被列表中的每个文件调用一次。它的通常形式如下：
- **boolean accept(File directory, String filename)**
- 当被**directory** 指定的目录中的文件（也就是说，那些与**filename** 参数匹配的文件）包含在列表中时，**accept()**方法返回**true**，当这些文件没有包括在列表中时，**accept()**返回**false**
- 参见程序 **FileTest11.java**



File类

- listFiles () 方法

File[] listFiles()

File[] listFiles(FilenameFilter FFObj)

File[] listFiles(FileFilter FObj)



File类

- 上述三种形式以**File**对象数组的形式返回文件列表，而不是用字符串形式返回。第一种形式返回所有的文件，第二种形式返回满足指定**FilenameFilter**接口的文件。除了返回一个**File** 对象数组，这两个**listFiles()**方法就像**list()**方法一样工作



File类

- 第三种**listFiles()**形式返回满足指定**FileFilter**的路径名的文件。**FileFilter**只定义了一个 **accept()**方法，该方法被列表中的每个文件调用一次。它的通常形式如下：
- **boolean accept(File path)**
- 如果文件被包括在列表中（即与**path**参数匹配的文件），**accept()**方法返回**true**，如果不被包括，则返回**false**。

