1. 硬盘:
2. 作用:实现计算机永久存储数据
3. 存储方式:以文件方式存储数据.

注意:文件的类型多样化的(如:.txt|.jpg|.gif|.doc|.java|.class….)

1. 什么是文件

文件可认为是相关记录或放在一起的数据集合。

注意:java提供操作文件的api,API都位于java.io包中

1. 使用java.io.File类操作文件的步骤
2. 创建File类的对象关联文件

File file=new File(文件的路径);

路径表示方式:c:\\a.txt 或者 c:/a.txt

1. 使用File对象的方法操作

exists() 判断文件是否存在

isFile() 判断是不是文件

isDirectory() 判断是不是目录

getPath() 获取文件的路径

getName() 获取文件的名称

delete() 删除文件

createNewFile() 创建文件

mkdir() 创建文件夹

length() 获取文件大小以字节为单位

listFiles() 获取目录下所有文件

示例:显示某目录下的所有文件

|  |
| --- |
| public static void main(String[] args) {  //获取某目录下的所有文件和文件夹  //1.创建fILE对象  File file=new File("d://");  //2.获取目录下所有资源  File [] flist=file.listFiles();  //循环显示所有文件  System.out.println("名称\t大小\t类型");  for (File f:flist) {  System.out.print(f.getName()+"\t");  System.out.print(f.length()+"\t");  if(f.isFile())  System.out.println("文件");  else  System.out.println("文件夹");  }  } |

1. 流（Stream）
2. 什么是流:

流是指一连串流动的字符，是以先进先出方式发送信息的通道.

1. 流分类:

2.1按流向分类

输入流(读文件):是InputStream和Reader作为基类

输出流(写文件):是OutputStream和Writer作为基类

2.2按数据处理单元划分

字节流：InputStream和OutputStream进行读写文件

字符流: Reader和Writer进行读写文件

注意:字节流和字符流的区别:

字节流是以8位通用的字节流、字符流是以16位的unicode字符流

1. 读写文件的步骤

3.1创建相应的流对象

3.2使用流对象的方法进行读写操作

3.3关闭流对象

1. 使用二进制的字节流读写文件

FileInputStream和FileOutStream

4.1*使用FileOutputStream来写文件*写文件

|  |
| --- |
| **public static void** main(String[] args) {  *//使用FileOutputStream来写文件:字节输出流* String str=**"hello:我爱你!"**;  *//将字符串转化为字节数组* **byte** [] bs=str.getBytes();  **try** {  *//1.创建FileOutputStream流对象  //FileOutputStream fos=new FileOutputStream("D:\\a.txt"); //每次重新写入内容* FileOutputStream fos=**new** FileOutputStream(**"D:\\a.txt"**,**true**); *//以追加方式写入内容  //2.进行写操作  //2.1写入单个字符  //fos.write('a');  //2.2写入字节数组* fos.write(bs);  *//3.关闭资源* fos.close();  System.***out***.println(**"写入完毕"**);  } **catch** (FileNotFoundException e) {  e.printStackTrace();  } **catch** (IOException e) {  e.printStackTrace();  } |

4.2使用*使用FileInputStream读文件*

|  |
| --- |
| **try** {  *//1.创建FileInputStream对象* FileInputStream fis=**new** FileInputStream(**"d:\\a.txt"**);  *//2.进行读取操作  //2.1一次读取一个  //int math=fis.read();  //System.out.println((char)math);  //2.2读取所有内容  /\*int math;  while((math=fis.read())!=-1){  System.out.println((char)math);  }\*/  //2.3读取多个字节* **byte** [] bs=**new byte**[fis.available()]; *//fis.available()文件的大小* fis.read(bs);  *//将字节数数组转化为字符串* String str=**new** String(bs);  System.***out***.println(str);  *//3.关闭资源* fis.close(); } **catch** (FileNotFoundException e) {  e.printStackTrace(); } **catch** (IOException e) {  e.printStackTrace(); } |

1. 字符串与字节数组之间的转换
2. 将字符串转化为字节数组

|  |
| --- |
| String str=**"hello:我爱你!"**; *//将字符串转化为字节数组* **byte** [] bs=str.getBytes(); |

1. 将字节数组转化为字符串

|  |
| --- |
| **byte** [] bs=**new byte**[10];  *//将字节数数组转化为字符串* String str=**new** String(bs); |