

通过Java.net包建立双向通讯 PDF转换可能丢失图片或格式，
建议阅读原文

https://www.100test.com/kao_ti2020/240/2021_2022__E9_80_9A_E8_BF_87Java_c104_240570.htm 利用java.net包提供的网络能力

。以有连接流通讯方式为例，在服务器端的操作为一般为：

创建一个ServerSocket对象，在指定端口监听客户端发来的请求。在接收到请求时accept()方法将返回一个Socket对象

。用上述Socket对象创建输入、输出流对象。通过输入、输出流与客户交互。交互完毕，关闭输入、输出流

与Socket。服务程序运行结束，关闭ServerSocket。实现代码

代码类如:

```
try{ boolean flag=true. Socket clientSocket=null.
```

```
ServerSocket serverSocket = new ServerSocket(0).
```

```
System.out.println("Server listen on: " serverSocket.getLocalPort()).
```

```
while(flag){ clientSocket=serverSocket.accept(). DataInputStream
```

```
is=new DataInputStream( new bufferedInputStream(client
```

```
Socket.getInputStream()). PrintStream os=new PrintStream( new
```

```
bufferedOutputStream(clientSocket. getOutputStream())). // 处
```

```
理Applet请求 os.close(). is.close(). clientSocket.close(). }
```

```
serverSocket.close(). }catch( IOException e){ System.err.println("
```

```
Exception: " e). }
```

在客户端的操作为： 创建Socket对象建立与服务器的连接。用该Socket对象创建输入、输出流。

与服务器交互。 交互完毕，关闭输入、输出流与Socket。

实现代码类如:

```
try { Socket clientSocket =new
```

```
Socket("serverName",7). OutputStream
```

```
os=clientSocket.getOutputStream(). DataInputStream is=new
```

```
DataInputStream( clientSocket.getInputStream()) . // 其它操作.
```

```
os.close(). is.close(). clientSocket.close(). }catch(Exception e){  
System.err.println("Exception:" e). }
```

这种方法只依赖于标准的Java网络支持，不需要用到附加的软件包或工具，因此显得相当简洁和灵活，易于实现某些特殊的需要。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