

java初学者实践教程26 - 网络程序 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/244/2021_2022_java_E5_88_9D_E5_AD_A6_c104_244681.htm Java在网络编程这个地方做的很好，java的主要目的也是为了网络而生的，它能方便的访问网络上的资源。我们这节课来介绍网络通讯的两种机制

：URL通信机制，Socket通信机制。URL表示了Internet上一个资源的引用或地址。Java网络应用程序也是使用URL来定位要访问的Internet的资源。在jdk里面java.net.URL也是一个类，它来封装URL的一些细节。目前大家可以把URL理解为网址，default.aspx 这就是个URL.http是协议名（超文本传输协议）用“://”隔开www.chinaitlab.com 是主机名。Default.aspx是文件名。它的端口号没有写，默认是80. 实践：

```
import java.net.*;public class ParseURL { public static void main(String[] args) throws MalformedURLException{ URL url = new URL("http://www.100jq.com:45175/default.aspx"); System.out.println("协议是 " + url.getProtocol()); System.out.println("主机是 " + url.getHost()); System.out.println("文件名是 " + url.getFile()); System.out.println("端口号是 " + url.getPort()); }}/* URL这个对象中提供了很多方法像是 getProtocol() getHost() getFile() getPort()*/ 我们可以通过URL对文件或资源读取，也可以通过URLConnection读取，也可以通过这个写入数据限于cgi脚本。 实践：import java.net.*;import java.io.*;public class URLConnectionReader { public static void main(String[] args) throws IOException { URL google = new URL(""); URLConnection g = google.openConnection();
```

BufferedReader in = new BufferedReader(new
InputStreamReader(g.getInputStream())). String inputLine. while
((inputLine=in.readLine())!=null) System.out.println(inputLine).
in.close(). } } URL和URLConnection类提供了较高层次的网络访
问。有时候需要进行较低层次的访问。编写C/S模型的程序时
, 就要使用Socket通信机制了。因为在网络上不一定非得访问
文件。 实践://先写个客户端的应用 import java.net.*.import
java.io.*.public class SimpleClient { public static void main(String
args[]) { try { // 在5432端口打开服务器连接 // 在这里用localhost
与127.0.0.1是一个意思 Socket s1 = new Socket("127.0.0.1", 5432).
// 对这个端口连接一个reader,注意端口不能够占用别的
BufferedReader br = new BufferedReader(new
InputStreamReader(s1.getInputStream())). // 读取输入的数据并
且打印在屏幕上 System.out.println(br.readLine()). //当完成时关
闭流和连接 br.close(). s1.close(). } catch (ConnectException
connExc) { System.err.println("Could not connect to the server."). }
catch (IOException e) { // ignore }}//这是服务端的应用import
java.net.*.import java.io.*.public class SimpleServer { public static
void main(String args[]) { ServerSocket s = null. // 注册服务端
口为5432 try { s = new ServerSocket(5432). } catch (IOException e) {
e.printStackTrace(). } // 运行监听器并接收, 永远循环下去。因
为服务器总要开启的 while (true) { try { // 等待一个连接的请求
Socket s1 = s.accept(). // 得到端口的输出流 OutputStream s1out
= s1.getOutputStream(). BufferedWriter bw = new BufferedWriter(
new OutputStreamWriter(s1out)). // 发送一个字符串 bw.write("
百家拳软件项目研究室欢迎您!\n"). // 关闭这个连接,但不是服

```
务端的socket bw.close(). s1.close(). } catch (IOException e) {  
e.printStackTrace(). }}}
```

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