

JAVA教程第八讲Java网络编程8.3 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/138/2021_2022_JAVA_E6_95_99_E7_A8_8B_c97_138729.htm 8.3 基于Socket（套接字）的低

层次Java网络编程8.3.1 Socket通讯 网络上的两个程序通过一个双向的通讯连接实现数据的交换，这个双向链路的一端称为一个Socket。Socket通常用来实现客户方和服务方的连接。Socket是TCP/IP协议的一个十分流行的编程界面，一个Socket由一个IP地址和一个端口号唯一确定。在传统的UNIX环境下可以操作TCP/IP协议的接口不止Socket一个，Socket所支持的协议种类也不光TCP/IP一种，因此两者之间是没有必然联系的。在Java环境下，Socket编程主要是指基于TCP/IP协议的网络编程。说Socket编程是低层次网络编程并不等于它功能不强大，恰恰相反，正因为层次低，Socket编程比基于URL的网络编程提供了更强大的功能和更灵活的控制，但是却要更复杂一些。由于Java本身的特殊性，Socket编程在Java中可能已经是层次最低的网络编程接口，在Java中要直接操作协议中更低的层次，需要使用Java的本地方法调用（JNI），在这里就不予讨论了。

8.3.2 Socket通讯的一般过 前面已经提到Socket通常用来实现C/S结构。使用Socket进行Client/Server程序设计的一般连接过程是这样的：Server端Listen(监听)某个端口是否有连接请求，Client端向Server端发出Connect(连接)请求，Server端向Client端发回Accept（接受）消息。一个连接就建立起来了。Server端和Client端都可以通过Send，Write等方法与对方通信。如图 对于一个功能齐全的Socket，都要包含以下基本结构，其工作过程包含以下四

个基本的步骤：（1）创建Socket；（2）打开连接到Socket的输入/出流；（3）按照一定的协议对Socket进行读/写操作；（4）关闭Socket. 第三步是程序员用来调用Socket和实现程序功能的关键步骤，其他三步在各种程序中基本相同。以上4个步骤是针对TCP传输而言的，使用UDP进行传输时略有不同，在后面会有具体讲解。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