

C\_S环境下的Java应用程序运行过程讲解 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/267/2021\\_2022\\_C\\_S\\_E7\\_8E\\_AF\\_E5\\_A2\\_83\\_E4\\_c104\\_267214.htm](https://www.100test.com/kao_ti2020/267/2021_2022_C_S_E7_8E_AF_E5_A2_83_E4_c104_267214.htm) 客户机/服务器在分布处理过程中，使用基于连接的网络通信模式。该通信模式首先在客户机和服务器之间定义一套通信协议，并创建一Socket类，利用这个类建立一条可靠的链接；然后，客户机/服务器再在这条链接上可靠地传输数据。客户机发出请求，服务器监听来自客户机的请求，并为客户机提供响应服务。这就是典型的"请求--应答"模式。下面是客户机/服务器的一个典型运作过程：1、服务器监听相应端口的输入；2、客户机发出一个请求；3、服务器接收到此请求；4、服务器处理这个请求，并把结果返回给客户机；5、重复上述过程，直至完成一次会话过程。按照以上过程，我们使用Java语言编写一个分别针对服务器和客户机的应用程序（Application）。该程序在服务器上时，程序负责监听客户机请求，为每个客户机请求建立Socket连接，从而为客户机提供服务。本程序提供的服务为：读取来自客户机的一行文本，反转该文本，并把它发回给客户机。通过该程序实例我们看到，使用Java语言设计C/S程序时需要注意以下几点：（1）、服务器应使用ServerSocket类来处理客户机的连接请求。当客户机连接到服务器所监听的端口时，ServerSocket将分配一新的Socket对象。这个新的Socket对象将连接到一些新端口，负责处理与之相对应客户机的通信。然后，服务器继续监听ServerSocket，处理新的客户机连接。Socket和ServerSocket是Java网络类库提供的两个类。（2）、服务器使用了多线程机制。Server

对象本身就是一个线程，它的run()方法是一个无限循环，用以监听来自客户机的连接。每当有一个新的客户机连接时，ServerSocket就会创建一个新的Socket类实例，同时服务器也将创建一新线程，即一个Connection对象，以处理基于Socket的通信。与客户机的所有通信均由这个Connection对象处理。Connection的构造函数将初始化基于Socket对象的通信流，并启动线程的运行。与客户机的通信以及服务的提供，均由Connection对象处理。（3）、客户机首先创建一Socket对象，用以与服务器通信。之后需创建两个对象

：DataInputStream和PrintStream，前者用以从Socket的InputStream输入流中读取数据，后者则用于往Socket的OutputStream中写数据。最后，客户机程序从标准输入（如：控制台）中读取数据，并把这些数据写到服务器，在从服务器读取应答消息，然后把这些应答消息写到准输出。

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](http://www.100test.com)