

关键术语

Client socket (客户端套接字)	packet-based communication (基于包的通信)
Domain name (域名)	server socket (服务器套接字)
Domain name server (域名服务器)	socket (套接字)
localhost (本地主机)	stream-based communication (基于流的通信)
IP address (IP 地址)	TCP (传输控制协议)
port (端口)	UDP (用户数据报协议)

本章小结

- 1. Java 支持流套接字和数据报套接字。流套接字使用 TCP (传输控制协议) 来进行数据传输。而数据报套接字使用 UDP (用户数据报协议)。由于 TCP 协议检测丢失的传输并重新提交它们, 所以, 传输是无损的和可靠的。相反地, UDP 不能保证传输是无损的。
- 2. 要创建一个服务器, 必须首先使用语句 `new ServerSocket(port)` 获取一个服务器套接字。在创建服务器套接字之后, 可以启动服务器, 使用服务器套接字上的 `accept()` 方法监听连接请求。客户端通过使用 `new socket(serverName,port)` 来创建一个客户端套接字, 用于向服务器发送连接请求。
- 3. 当服务器与客户端的连接建立之后, 流套接字通信与输入输出流通信非常相似。可以通过套接字上的 `getInputStream()` 方法获得一个输入流, 通过 `getOutputStream()` 方法获得一个输出流。
- 4. 一个服务器经常同时与多个客户端协同工作。通过为每个连接创建一个线程, 可以利用线程同时处理服务器的多个客户端。

测试题

回答位于网址 [www.cs.armstrong.edu/liang/intro10e/quiz.html](http://www.cs.armstrong.edu/liang/intro10e/quiz.html) 的本章测试题。

编程练习题

31.2 节

\*31.1 (贷款服务器) 为一个客户端编写一个服务器。客户端向服务器发送贷款信息 (年利率、贷款年限和贷款总额) (如图 31-17a 所示)。服务器计算月偿还额和总偿还额, 并把它们发回给客户端 (如图 31-17b 所示)。将客户端程序命名为 `Exercise31_01Client`, 将服务器程序命名为 `Exercise31_01Server`。

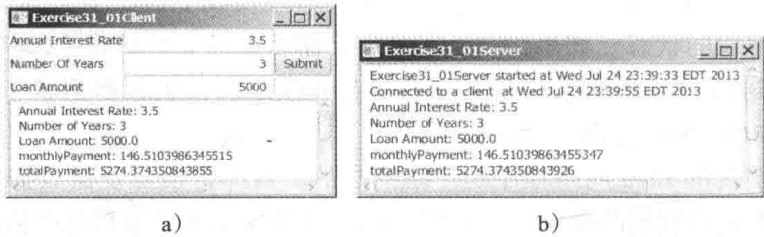


图 31-17 a) 客户端向服务器发送年利率、贷款年限和贷款总额; b) 从服务器接收月偿还额和总偿还额

\*31.2 (BMI 服务器) 为一个客户端编写一个服务器。客户端向服务器发送体重和身高信息 (如图 31-18a 所示)。服务器计算 BMI (Body Mass Index, 身体质量指数), 发回一个报告 BMI 的字符串给客户端 (如图 31-18b 所示)。如何计算 BMI 参见 3.8 节。将客户端程序命名为

Exercise31\_02Client, 将服务器程序命名为 Exercise31\_02Server。

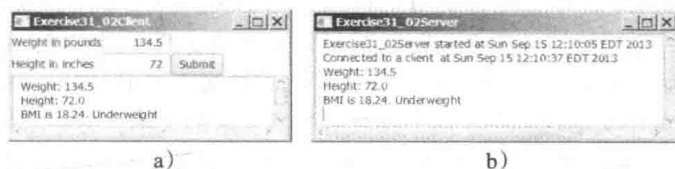


图 31-18 a) 客户端向服务器发送一个人的体重和身高; b) 从服务器接收 BMI

### 31.3 节和 31.4 节

\*31.3 (可用于多客户端的贷款服务器) 修改编程练习题 31.1, 编写一个可以用于多客户端的贷款服务器。

### 31.5 节

31.4 (对客户计数) 编写一个服务器程序, 追踪连接到服务器的客户数量。当一个新的连接建立的时候, 计数加 1。计数采用随机访问文件存储。编写一个客户程序, 接收从服务器发来的数字并且显示一条消息, 比如 “You are visitor number 11”, 如图 31-19 所示。将客户端程序命名为 Exercise31\_04Client, 将服务器程序命名为 Exercise31\_04Server。

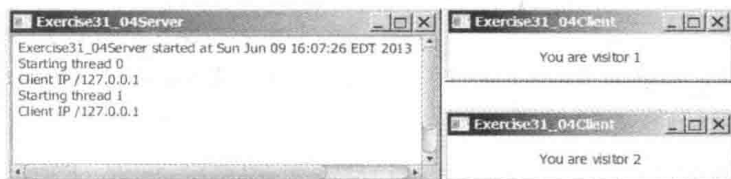


图 31-19 客户端显示服务器访问了多少次, 服务器端存储该计数

31.5 (将贷款信息以对象发送) 修改编程练习题 31.1, 使得客户端可以发送一个贷款对象, 该对象包含了年利率、年数以及贷款数等信息, 对于服务器来说, 可以发送月付和总支付金额。

### 31.6 节

31.6 (显示和添加地址) 开发一个客户端 / 服务器应用程序, 可以查看和添加地址, 如图 31-20 所示。

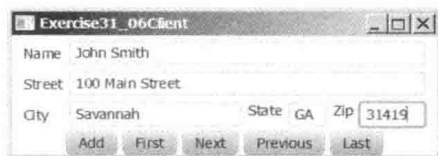


图 31-20 可以查看和添加地址

- 使用代码清单 31-5 中定义的 StudentAddress 类, 在其对象中保存 name (姓名)、street (街道)、city (城市)、state (州) 和 zip (邮编) 等属性。
- 用户可以使用按钮 First、Next、Previous 和 Last 来查看地址, 并且使用按钮 Add 添加新地址。
- 限制同时连接两个客户端。

将客户端程序命名为 Exercise31\_06Client, 将服务器程序命名为 Exercise31\_06Server。

\*31.7 (在数组中传递最后 100 个数字) 编程练习题 22.12 从文件 PrimeNumbers.dat 中获取最后 100 个素数。编写一个客户端程序, 要求服务器发送数组中的最后 100 个素数。将服务器程序命名为 Exercise31\_07Server, 将客户端程序命名为 Exercise31\_07Client。假设 PrimeNumbers.dat 中的 long 类型的数字以二进制形式存储。

\*31.8 (在 ArrayList 中传递最后 100 个数字) 编程练习题 24.12 从名为 PrimeNumbers.dat 的文件中获取最后 100 个素数。编写一个客户端程序, 要求服务器发送 ArrayList 中的最后 100 个素数。将服务器程序命名为 Exercise31\_08Server, 将客户端程序命名为 Exercise31\_08Client。假设

PrimeNumbers.dat 中的 long 类型的数字以二进制形式存储。

**\*\*31.9 (聊天程序)** 编写一个程序，使两个用户可以互相聊天。把一个用户实现为服务器（见图 31-21a），另一个用户实现为客户端（见图 31-21b）。服务器有两个文本域：一个用于输入文本，另一个（不可编辑的）显示从客户端接收的文本。当用户按下 Enter 键时，当前行就被发送给客户端。客户端也有两个文本域：一个用于接收服务器发来的文本，另一个用于输入文本。当用户按下 Enter 键时，当前行被发送给服务器。将客户端程序命名为 Exercise31\_09Client，将服务器程序命名为 Exercise31\_09Server。



图 31-21 服务器与客户端互相发送或接收文本信息

**\*\*\*31.10 (多客户端聊天程序)** 编写一个程序，允许任意数目的客户端互相聊天。实现一个服务器为所有的客户端服务，如图 31-22 所示。将客户端程序命名为 Exercise31\_10Client，将服务器程序命名为 Exercise31\_10Server。

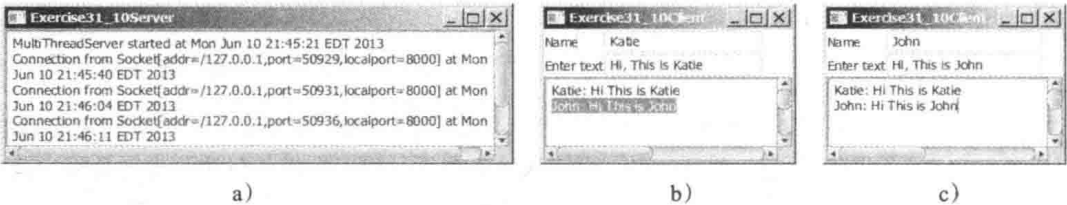


图 30-22 服务器在 a 中启动，它有三个客户端，如 b 和 c 中所示