

以下实例演示了如何实现客户端发送消息到服务器，服务器接收到消息并读取输出，然后写出到客户端客户端接收到输出。

1、建立服务器端

服务器建立通信ServerSocket

服务器建立Socket接收客户端连接

建立IO输入流读取客户端发送的数据

建立IO输出流向客户端发送数据消息

服务器端代码:

Server.java 文件

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
import java.net.ServerSocket;
import java.net.Socket;

public class Server {
    public static void main(String[] args) {
        try {
            ServerSocket ss = new ServerSocket(8888);
            System.out.println("启动服务器....");
            Socket s = ss.accept();
            System.out.println("客户端:"+s.getInetAddress().getLocalHost()+"已连接到服务器");

            BufferedReader br = new BufferedReader(new InputStreamReader(s.getInputStream()));
            //读取客户端发送来的消息
            String mess = br.readLine();
            System.out.println("客户端: "+mess);

            BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(s.getOutputStream()));
            bw.write(mess+"\n");
            bw.flush();

        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

Socket就是IP和端口绑定的一个东西，插头

socket返回的是一个字节流

读取客户端发过来的数据后又返回给客户端

以上代码运行输出结果为：

```
启动服务器....
```

2、建立客户端

创建Socket通信，设置通信服务器的IP和Port

建立IO输出流向服务器发送数据消息

建立IO输入流读取服务器发送来的数据消息

客户端代码:

Client.java 文件

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.net.Socket;
import java.net.UnknownHostException;

public class Client {
    public static void main(String[] args) {
        try {
            Socket s = new Socket("127.0.0.1", 8888);

            //构建IO
            InputStream is = s.getInputStream();
            OutputStream os = s.getOutputStream();

            BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(os));
            //向服务器端发送一条消息
            bw.write("测试客户端和服务器通信，服务器接收到消息返回到客户端\n");
            bw.flush();

            //读取服务器返回的消息
            BufferedReader br = new BufferedReader(new InputStreamReader(is));
            String mess = br.readLine();
            System.out.println("服务器: "+mess);
        } catch (UnknownHostException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

以上代码运行输出结果为：

服务器：测试客户端和服务器通信，服务器接收到消息返回到客户端