

## Server side code

```
import java.io.*;
import java.net.*;

public class EchoServer {
    private static final int PORT = 34522;

    public static void main(String[] args) {
        try (ServerSocket server = new ServerSocket(PORT)) {
            while (true) {
                try (
                    Socket socket = server.accept(); // accept a new client
                    DataInputStream input = new
DataInputStream(socket.getInputStream());
                    DataOutputStream output = new
DataOutputStream(socket.getOutputStream())
                ) {
                    String msg = input.readUTF(); // read a message from the client
                    output.writeUTF(msg); // resend it to the client
                }
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

The program creates only a single `ServerSocket` and then accepts client connections in an infinite loop. The program stops when a shutdown occurs.

We also use the **try-with-resources** statement so the socket and streams can be closed automatically to avoid resource leaks.