1.) Traditional socket based TCP server/client example

1/a) Socket server source

```
import java.io.*;
import java.net.*;
public class Provider{
      ServerSocket providerSocket;
      Socket connection = null;
      ObjectOutputStream out;
      ObjectInputStream in;
      String message;
      Provider() {}
      void run() {
    try{
        providerSocket = new ServerSocket(8080, 10);
        connection = providerSocket.accept();
        out = new ObjectOutputStream(connection.getOutputStream());
        in = new ObjectInputStream(connection.getInputStream());
                    do {
            try {
                message = (String)in.readObject();
                System.out.println("client>" + message);
                if (message.equals("bye")) sendMessage("bye");
            }
            catch(ClassNotFoundException classnot){
                System.err.println("Data received in unknown format");
            }
        } while(!message.equals("bye"));
    }
    catch(IOException ioException){
        ioException.printStackTrace();
    }
    finally{
        try {
            in.close();
            out.close();
            providerSocket.close();
        }
        catch(IOException ioException){
            ioException.printStackTrace();
        }
    }
}
void sendMessage(String msg) {
    try {
        out.writeObject(msg);
        out.flush();
        System.out.println("server>" + msg);
    }
```

```
    catch(IOException ioException){
        ioException.printStackTrace();
    }
}
public static void main(String args[]) {
    Provider server = new Provider();
    while(true){
        server.run();
    }
}
}
```

1/b.) Client source

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

public class Requester{
      Socket requestSocket;
      ObjectOutputStream out;
      ObjectInputStream in;
      String message;
      Requester(){}
void run() {
    try{
        requestSocket = new Socket("localhost", 8080);
        out = new ObjectOutputStream(requestSocket.getOutputStream());
        in = new ObjectInputStream(requestSocket.getInputStream());
        do {
            try {
                sendMessage("Hello szerver");
                sendMessage("bye");
                                    message = (String)in.readObject();
            }
            catch(Exception e){
                System.err.println("data received in unknown format");
            }
        } while(!message.equals("bye"));
    }
    catch(UnknownHostException unknownHost){
        System.err.println("You are trying to connect to an unknown host!");
    }
    catch(IOException ioException){
        ioException.printStackTrace();
    }
    finally{
        try{
            in.close();
            out.close();
            requestSocket.close();
```

```
        }
        catch(IOException ioException){
            ioException.printStackTrace();
        }
    }
}
void sendMessage(String msg) {
    try {
        out.writeObject(msg);
        out.flush();
        System.out.println("client>" + msg);
    }
    catch(IOException ioException){
        ioException.printStackTrace();
    }
}

public static void main(String args[]) {
    Requester client = new Requester();
    client.run();
}
}
```