***Study of coding Remote Method Invocation and User Datagram Protocol***

Investigation of sending messages between client and server by using two different methods, which are Java **Remote Method Invocation (RMI)** and **User Datagram Protocol (UDP)**, is introducing in the following report. By sending varying numbers of messages, we will analyze the performance, efficiency and ease of set-up of each type of system.

**UDP:**

There are three main ways that messages can be lost; whilst they are being sent, whilst they are in transit or whilst they are being received.

When packets are sent via UDP, they are first placed in a ‘socket send buffer’, which is part of the networking hardware. If this becomes full, or cannot output packets as quickly as they come in, there will a loss of packets.

Majority of the packets are being lost in transit, due to congestion on the network. This can be caused by an overloaded router, which is receiving data faster than it can send it. At the same time the lab was so busy that the load on the Internet was high.

The chance of failure to receive message and loss of packages increases when the number of message above **300** with the distance between server and client increases.

**RMI:**

Data loss could not be happened for RMI, so the performance of receiving message is perfect.

There is no pattern of message loss.

**Comparison:**

1. RMI performs perfectly with no data loss, so it seems more reliable.
2. UDP has lots of data loss, especially when the distance of server and client increases. It also requires long time to receive message if the amount of data is large.
3. In the real life, UDP has wider use in anything where users do not care too much if users get all data always, such as media streaming (lost frames are fine). RMI can be used in funds transfer, which cares a lot the amount of data has received.

**Ease of setup:**

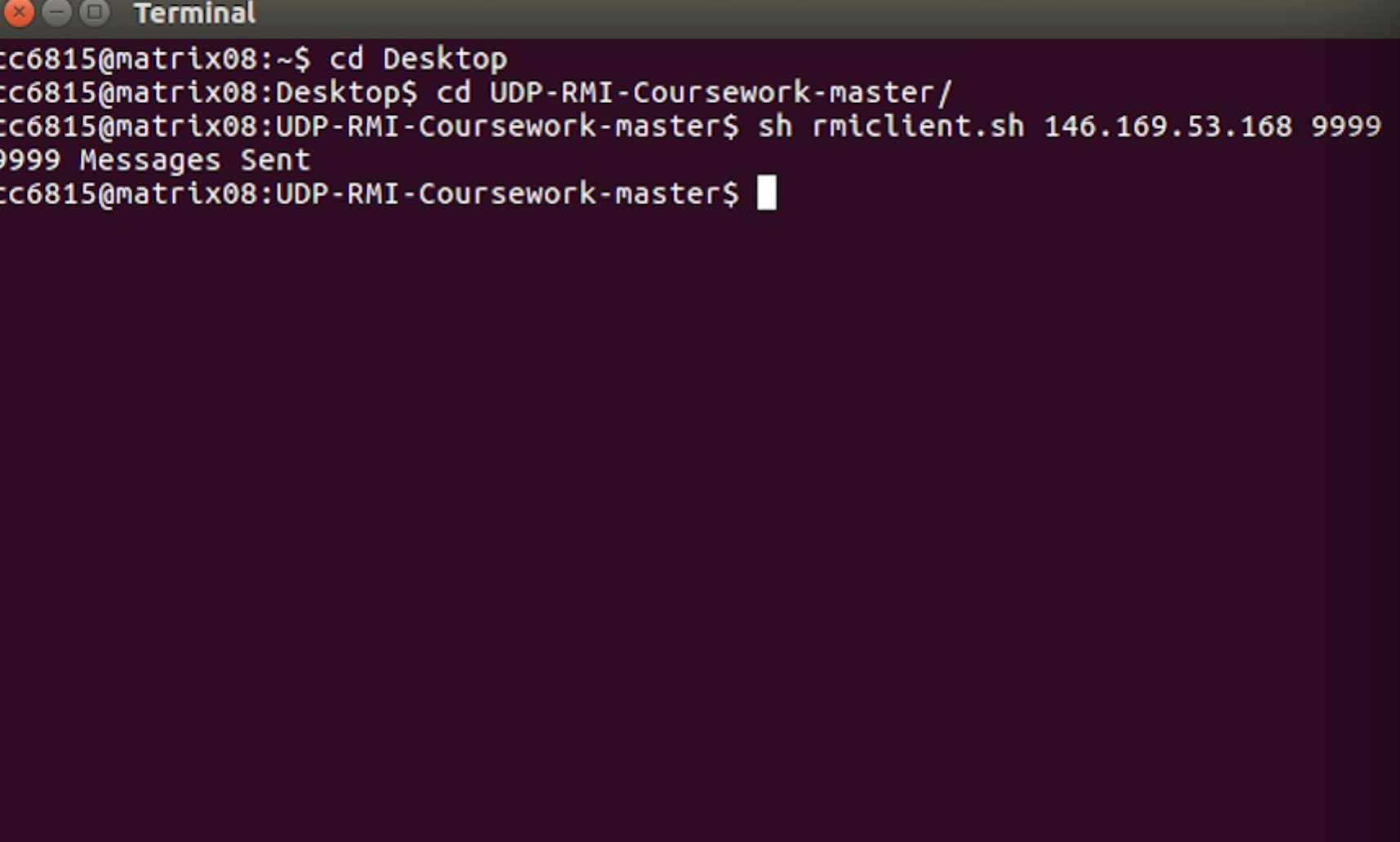
UDP is easier to understand the whole flow of the program, and the time cost of writing the program is shorter. Because the way of UDP implementing is straightforward,

**UDP Screenshot whilst running:**



**RMI Screenshot whilst running:**

****

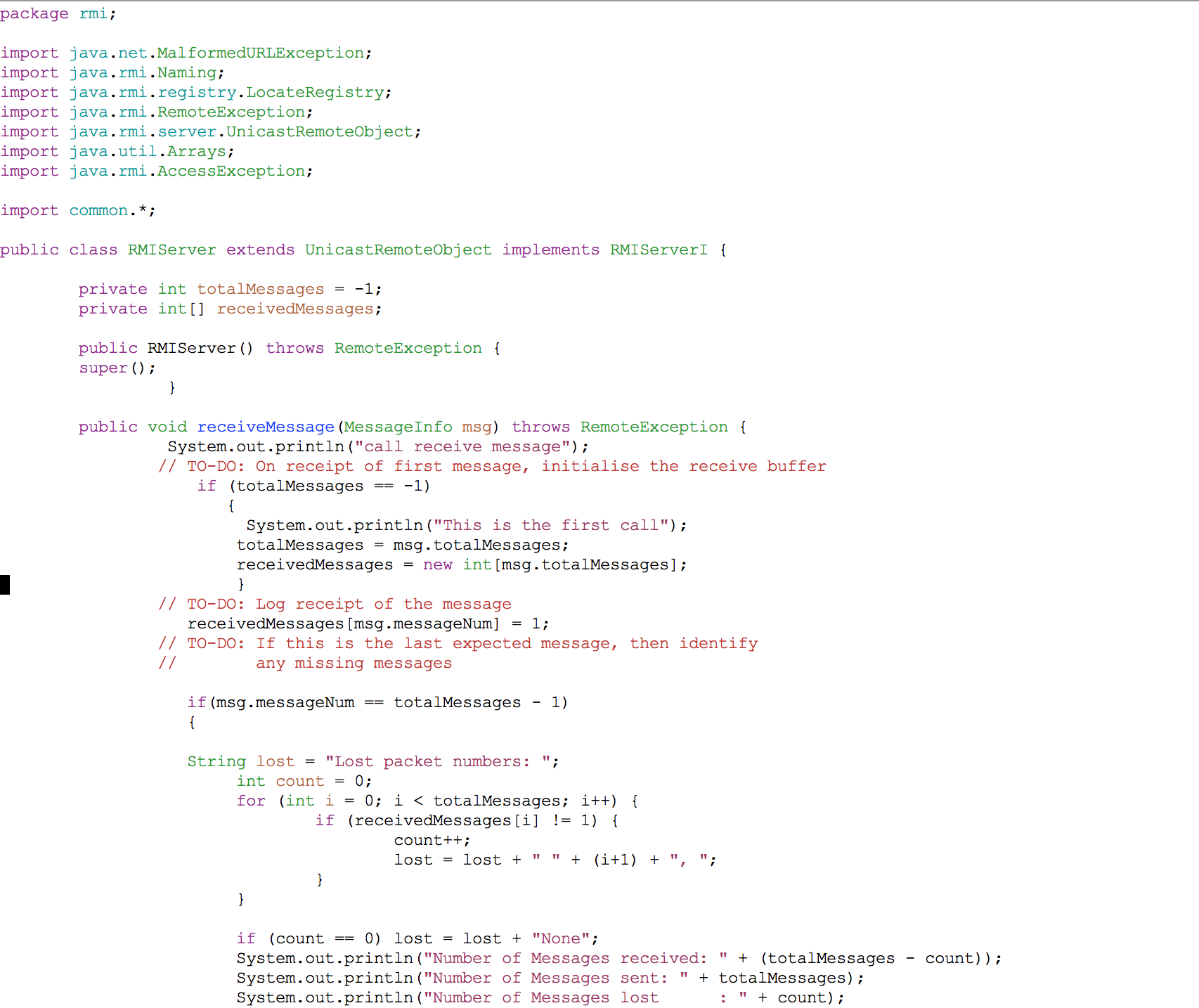
****

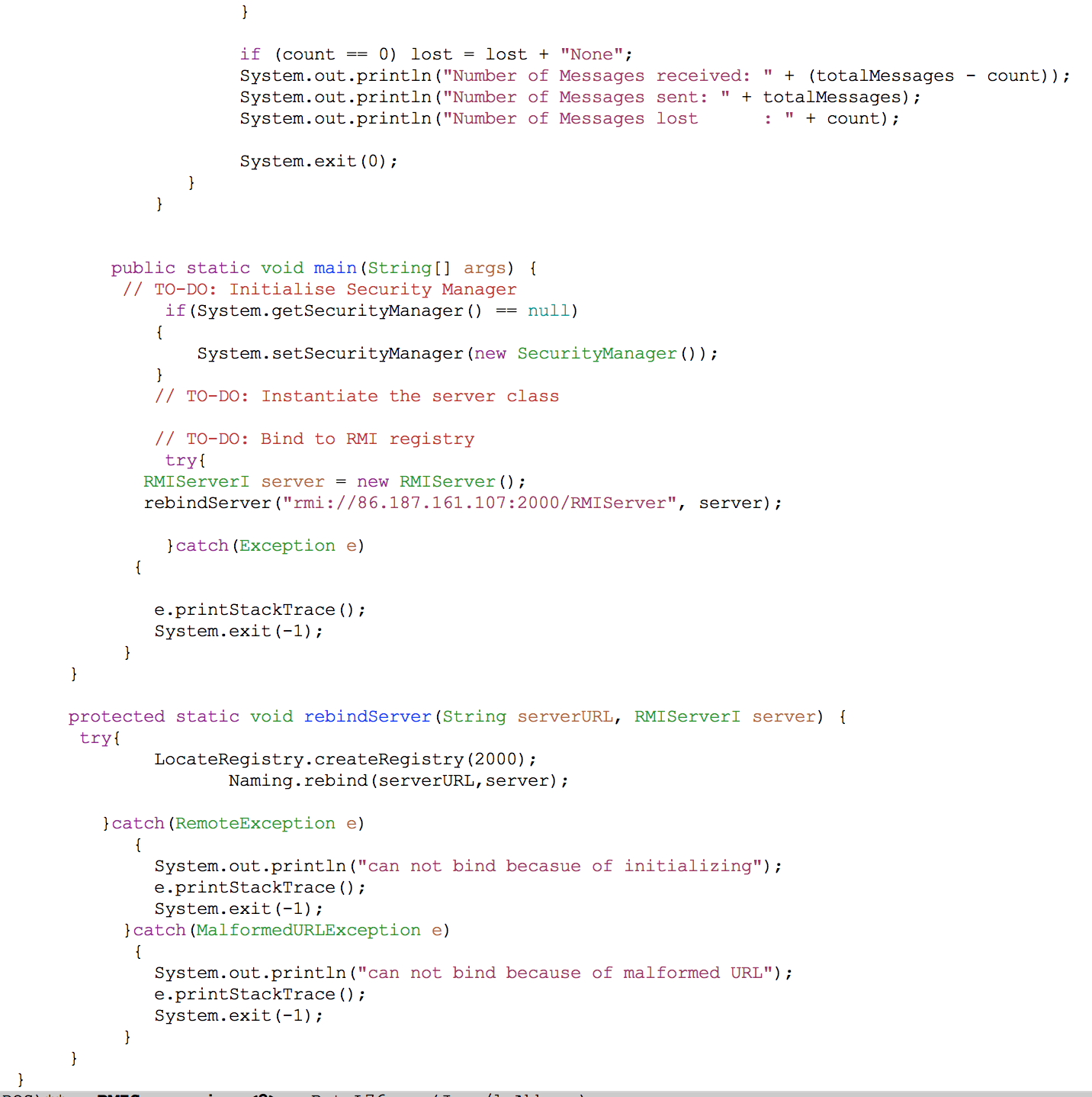
**Source Code:**

**1. RMI Client:**

****

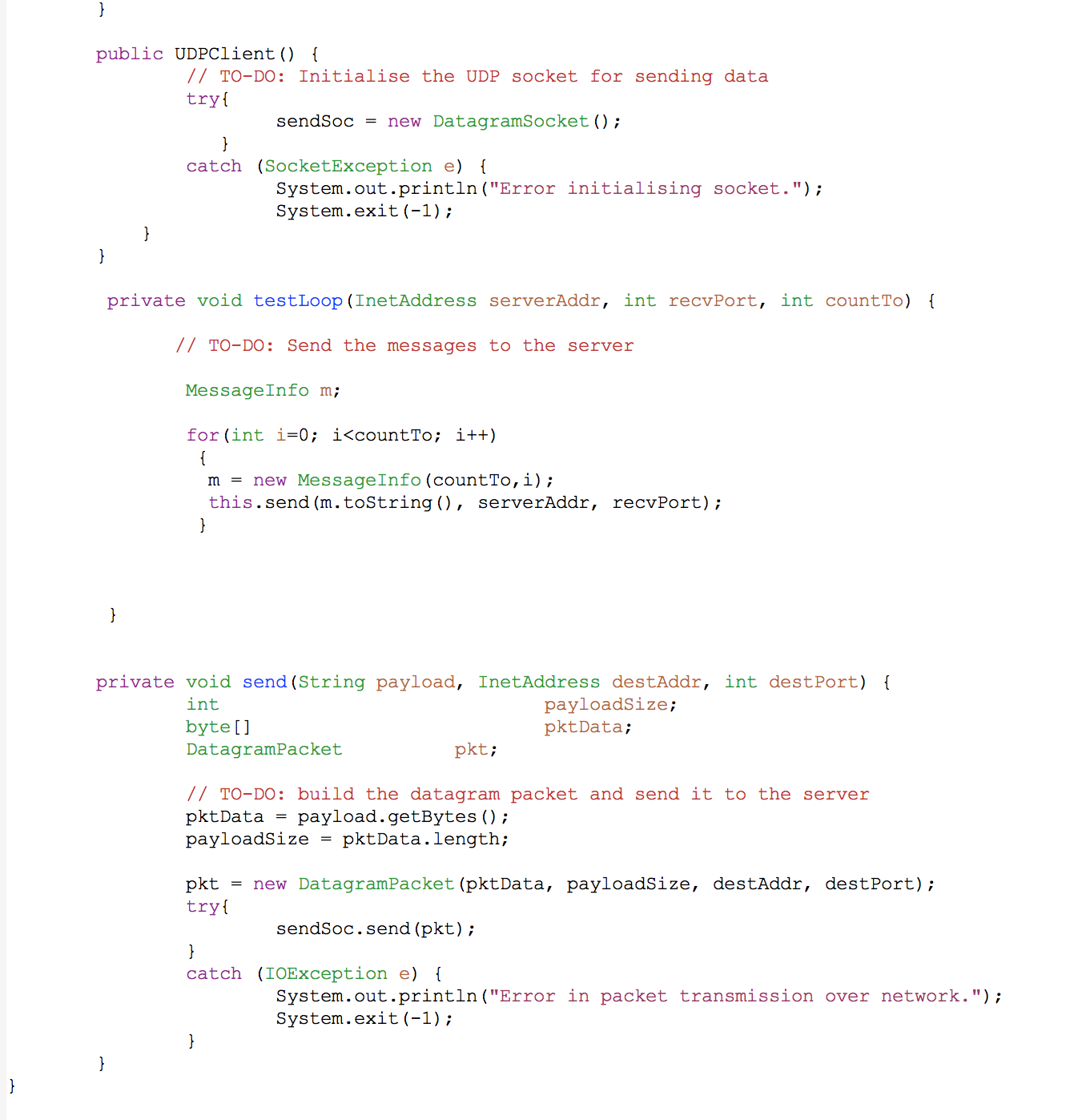
**2. RMI Server:**

****

****

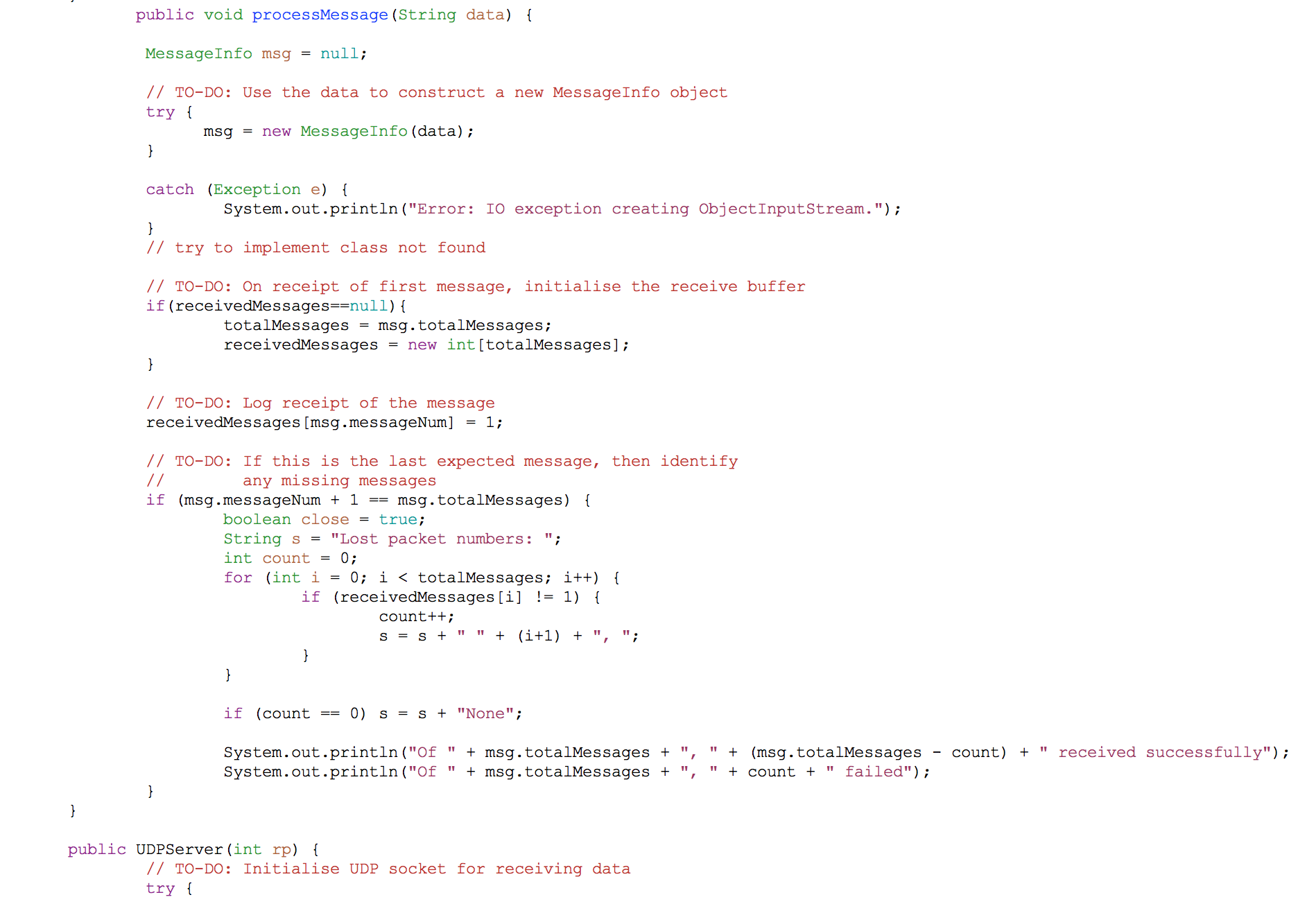
1. **UDP Client:**

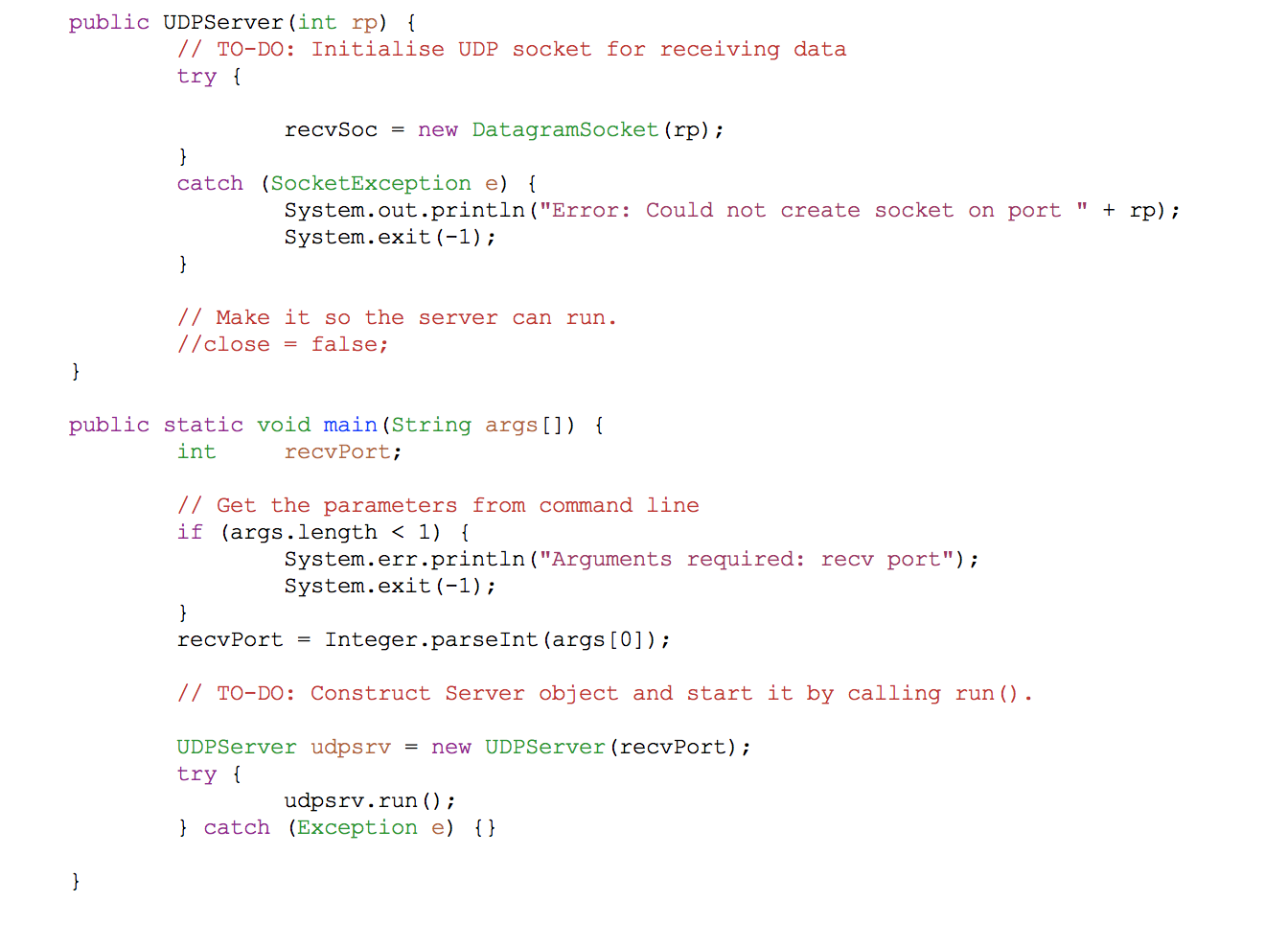
****

****

1. **UDP Server:**

****

****

****