课程作业

张楠

2019104283  
**背景知识**

**Socket**

Socket是两台主机之间的一个连接。它可以完成7个基本操作：

连接远程机器，发送数据，接收数据，关闭连接，绑定端口，监听入站数据，在绑定端口上接受来自远程机器的连接。

socket中TCP的三次握手建立连接：

tcp建立连接要进行“三次握手”，即交换三个分组。大致流程如下：  
客户端向服务器发送一个SYN J  
服务器向客户端响应一个SYN K，并对SYN J进行确认ACK J+1  
客户端再想服务器发一个确认ACK K+1

图示

描述已自动生成

当客户端调用connect时，触发了连接请求，向服务器发送了SYN J包，这时connect进入阻塞状态；服务器监听到连接请求，即收到SYN J包，调用accept函 数接收请求向客户端发送SYN K ，ACK J+1，这时accept进入阻塞状态；客户端收到服务器的SYN K ，ACK J+1之后，这时connect返回，并对SYN K进行确认；服务器收到ACK K+1时，accept返回，至此三次握手完毕，连接建立。

**客户端**

1、创建套接字  
2、向服务器发送连接请求（connect）  
3、通信（send/recv)  
4、关闭套接字

**服务器**

1、创建用于监听的套接字（socket）   
2、将套接字绑定到本地地址和端口上（bind）   
3、将套接字设为监听模式（listen）   
4、等待客户请求（accept），此处要不断的调用accept   
5、通信（send/receive），完成后返回4   
6、关闭套接字（closesocket）

**实验流程**

该项目通过socket客户端与服务器建立tcp连接链接建立成功，此时服务器端就会产生绿灯；当客户端下线的时候绿灯会变成红灯

1.客户端通过 socket = new Socket(IP,PORT); 链接到服务器

2.客户端一旦链接成功那么会通过ObjectOutputStream 流将封装好的MessageInfo对象的通过网络传输到服务器

3.服务通过客户端 Socket accept = serverSocket.accept();得到的accept Socket

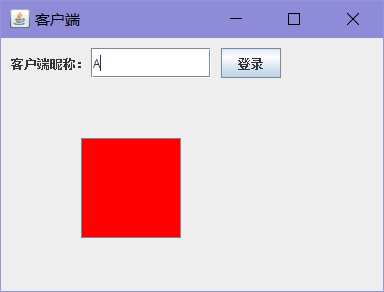
MessageInfo messageInfo=(MessageInfo)objectInputStream.readObject();

得到客户端传来的MessageInfo对象然后 maps.put(accept,messageInfo);中

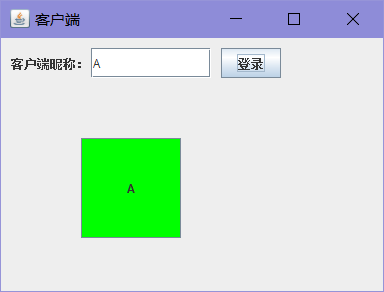
4.服务器启动监控线程将maps里面保存的客户端不停的遍历刷新页面，将灯的状态改变。

**实验截图**

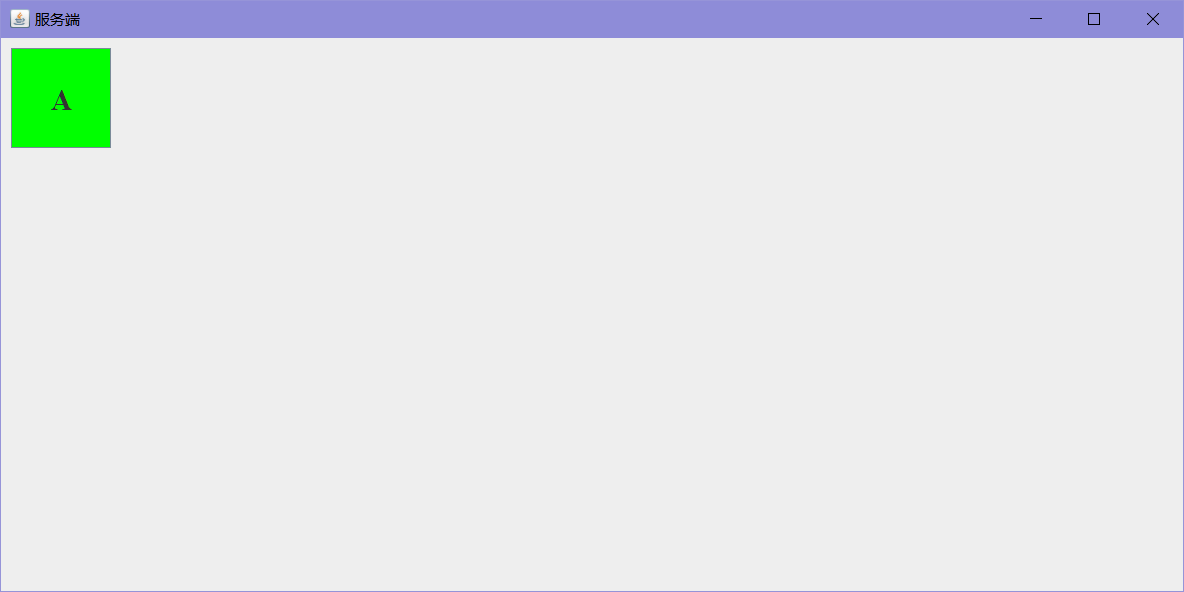
客户端界面：

未登录前  


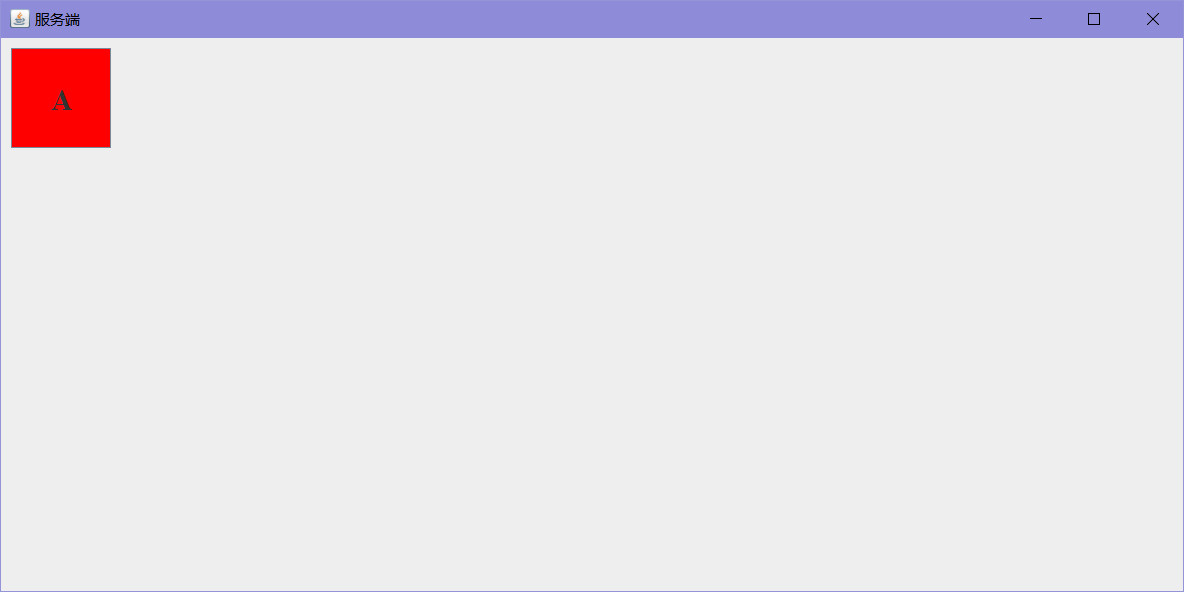
登陆后



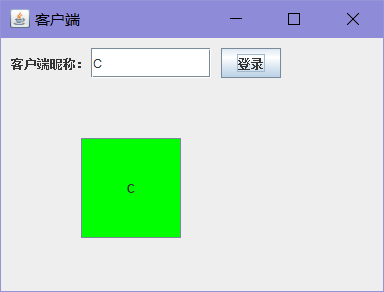
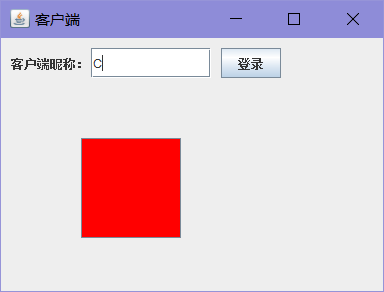
服务器界面：

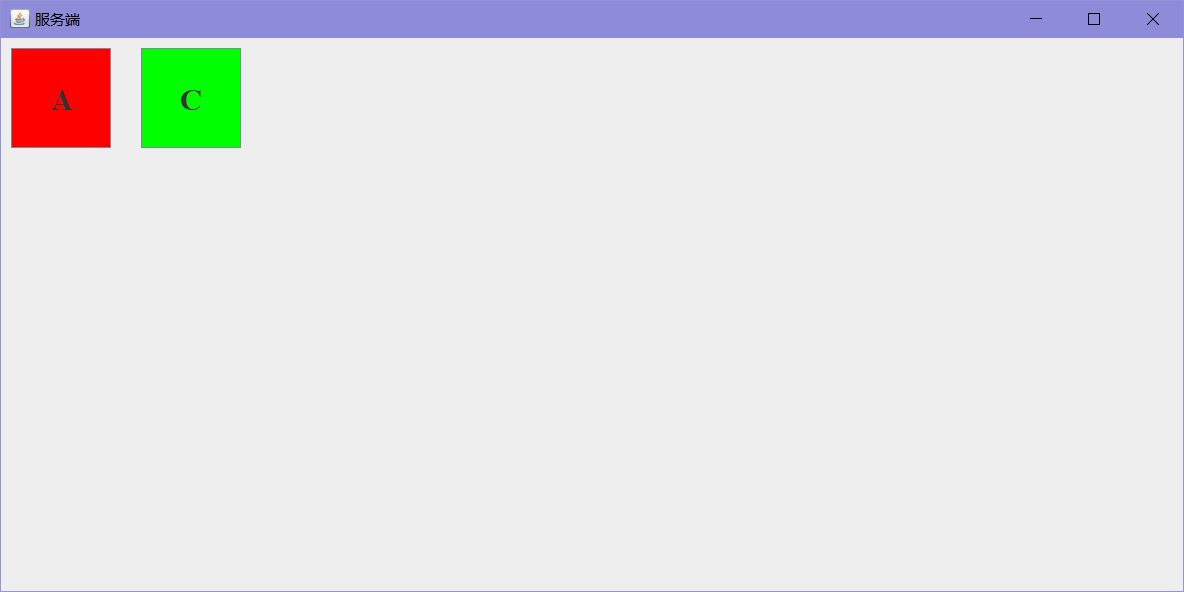


客户A关掉后链接丢失服务器改变客户端A绿灯变红灯

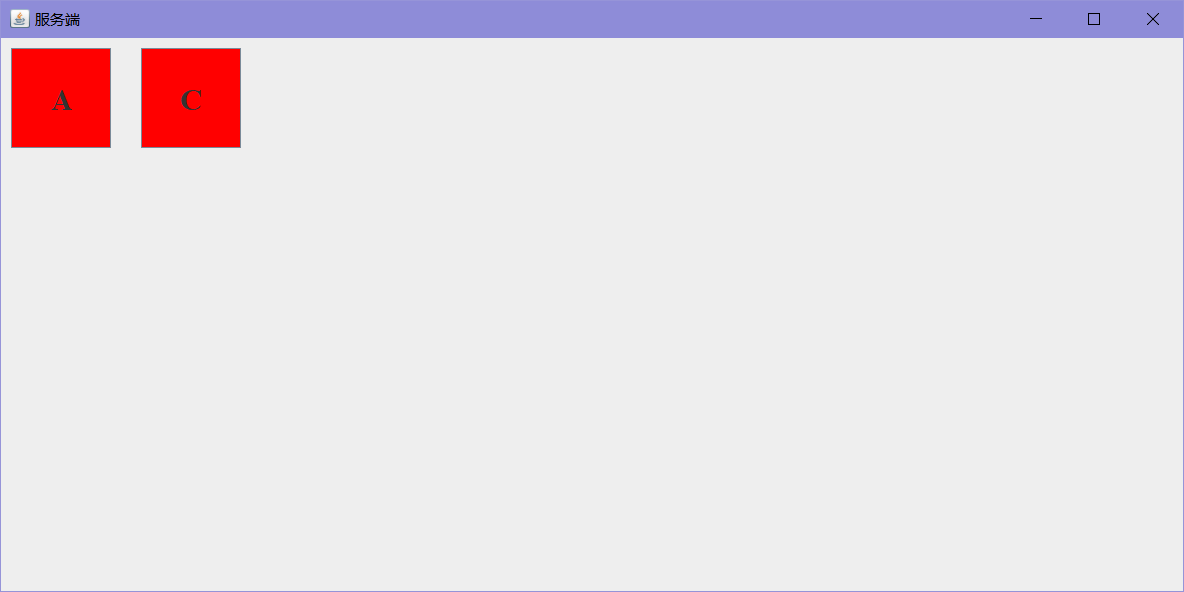


新增一个客户端C





客户端C断开灯变红



**结论**

通过这一阶段课程的学习，以及进一步利用中间件技术完成此次检测客户端上下线的实践，我进一步掌握了socket客户端服务器编程，结合实际背景，设计了一款具有实际应用价值的小程序，通过动手练习，巩固了学到的理论知识，为不久的将来的企业工作打下良好的基础。