实验报告

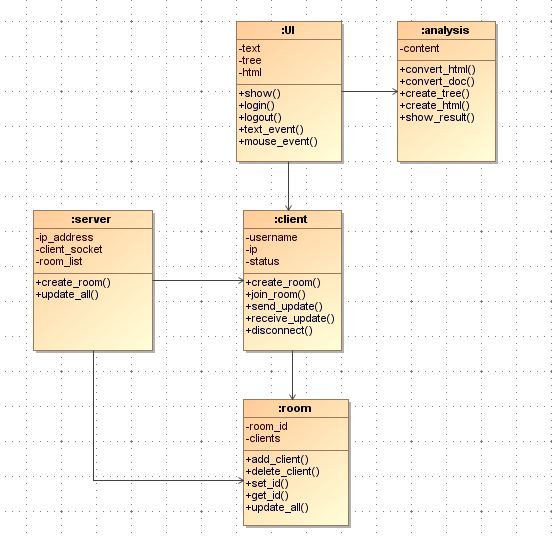
1. 问题说明

实现一个Markdown编辑器，具有如下功能：

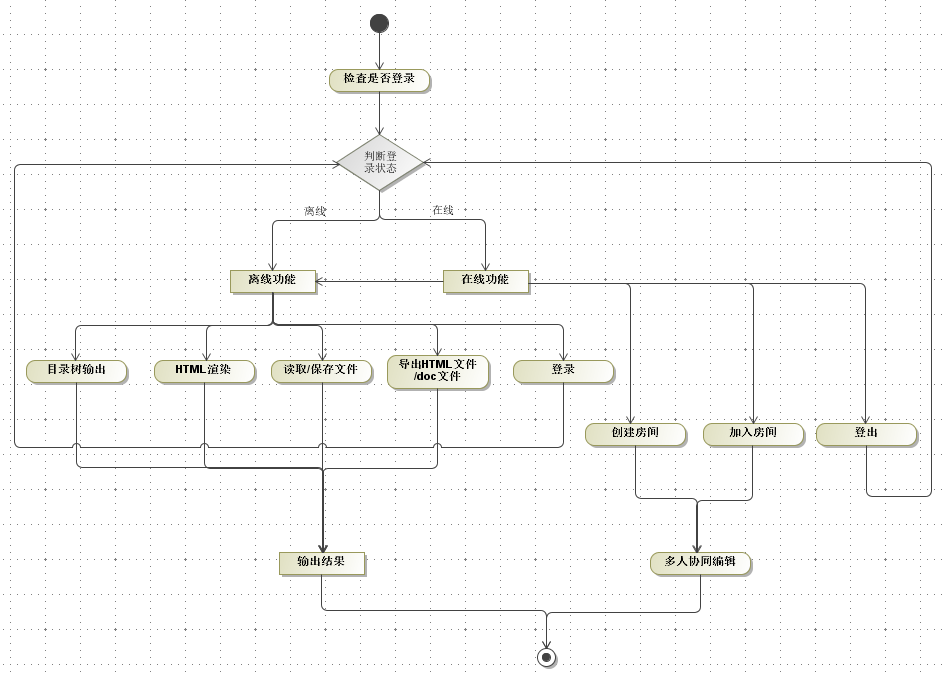
* 能编辑Markdown文档
* 能在编辑区的左侧看到实时的文档目录
* 能保存和打开Markdown文档
* 能输出html
* 能建立一个网络服务，以供其他编辑器连接
* 能连接其他编辑器，连接后可编辑对方正在编辑的文档
* 连接了其他编辑器后，能实时同步反映服务器上的文件在其他编辑器上的修改
* 加分项（5分）：能实时在编辑区右侧看Markdown渲染后的效果

1. 实现方式

类图：



流程图：



UI类：

1. 定义用户界面，定义菜单栏，未登录状态下可以保存和读取文件，并能导出HTML和doc文件，同时提供在线登录功能，只有在登录状态下才提供房间创建和加入功能。

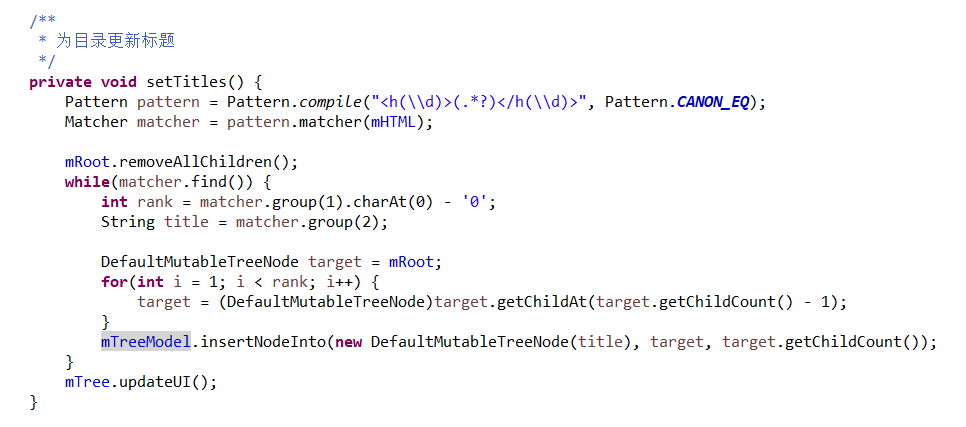




1. 添加鼠标事件，为不同的按钮定义动作：



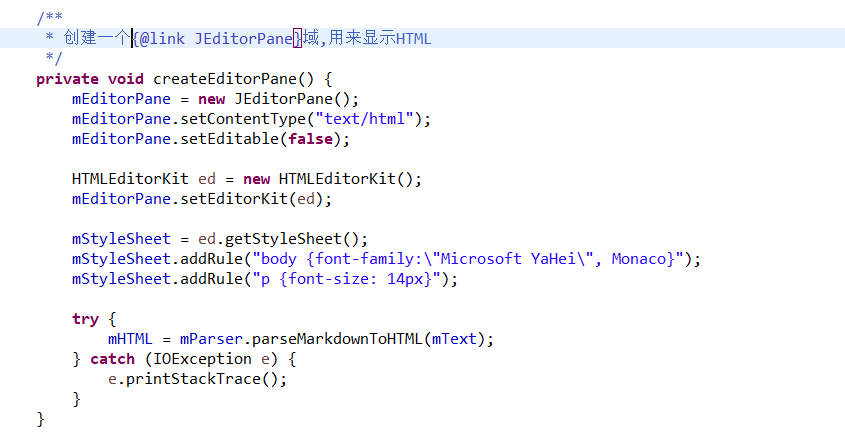
1. 目录结构，用了JTree，生成关于目录的树状结构，并对每个节点添加点击事件：



1. 编辑区，使用了JTextArea，生成了编辑区模块：

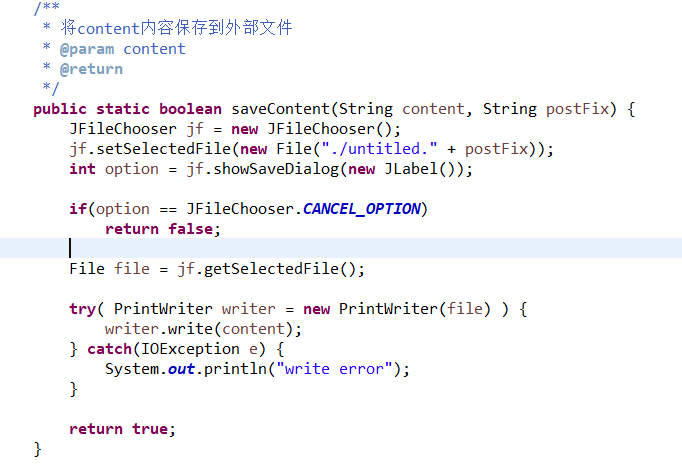


1. HTML渲染区，使用了JEditorPane，用于显示渲染之后的内容：



Analysis类：

1. Markdown导出，主要用了开源的markdown4j包：



1. 文件读入，同样使用了开源包：



1. 实时显示，为了减少不必要的计算，这里用线程阻塞的方式，实现高效地更新。每次，想要更新的线程先sleep 1000毫秒，如果之后又有想要更新的线程，就将上一个阻塞，这样的话上一个就直接返回，放弃更新的操作。这样的效果就是，只要持续输入或删除内容，对文本内容做出改变的间隔不超过1秒钟，就一直不会更新，当用户停止1秒后，才会执行更新操作。大大减少了无用的计算：



Socket编程部分：

1. 首先创建一个本地服务器Server，开放2个端口，8080和8081。同时可以创建若干个Client。Server处理客户端发来的所有请求，并返回回复；Client用于在客户端帮助用户发送所有请求。工程建立在**C/S**架构上。



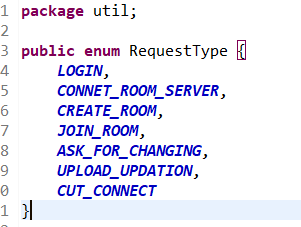
1. 所有客户端一旦试图进行登录或注册，就会创建一个Client，并建立相应的端口进行通信。服务器也会新开一个线程对这个客户端进行服务。客户端进行的一切操作都是在向服务器发送请求，然后得到服务器的回复。当客户端退出登录或是退出程序，就断开连接的端口，释放资源，防止内存浪费。



1. 一旦一个用户创建或是加入了一个房间，那么服务器就必须通过另一个端口——8081，来进行房间内内容的更新等操作。所以加入房间后，客户端会试图连接服务器的8081端口。对应的服务器上8081端口专门有一个线程来进行处理，依然是每有一个客户端连接过来就新开一个线程进行服务。这个端口做的事就是专门把房间主人发过来的文本内容发送给房间其他所有成员让他们进行同步。所以，每个客户端也必须开一个线程，不断尝试从对应8081端口的socket读取内容，然后更新。

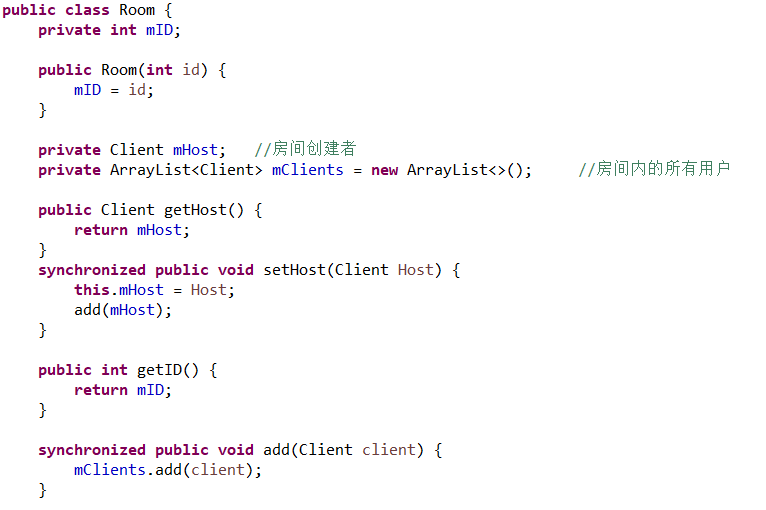


1. 关于请求的类型定义在RequestType.java中：



Room类：

1. 保存房间ID，房主的IP信息，房间成员的信息：



1. 创建房间，房间序号自动递增，最多为100：



1. 加入房间：

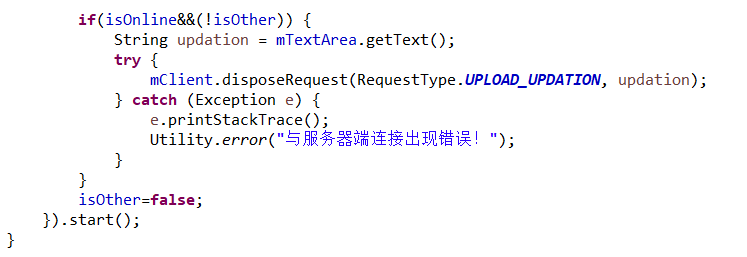


1. 房间内共享更新内容，每一次有一个用户进行修改之后都将修改后的内容上传至服务器，再通过服务器发送给房间内的另外所有用户，实现编辑内容的实时共享：



所有房间内的用户在收到服务器的更新指令后会自动刷新UI显示的内容，但是必须使用一个布尔值变量，用于表示这一次的更新是由他人发起的还是由自身发起的，如果由他人发起的修改就不再上传修改内容至服务器，如果是自身修改就上传至服务器，否则会引起循环更新的问题。

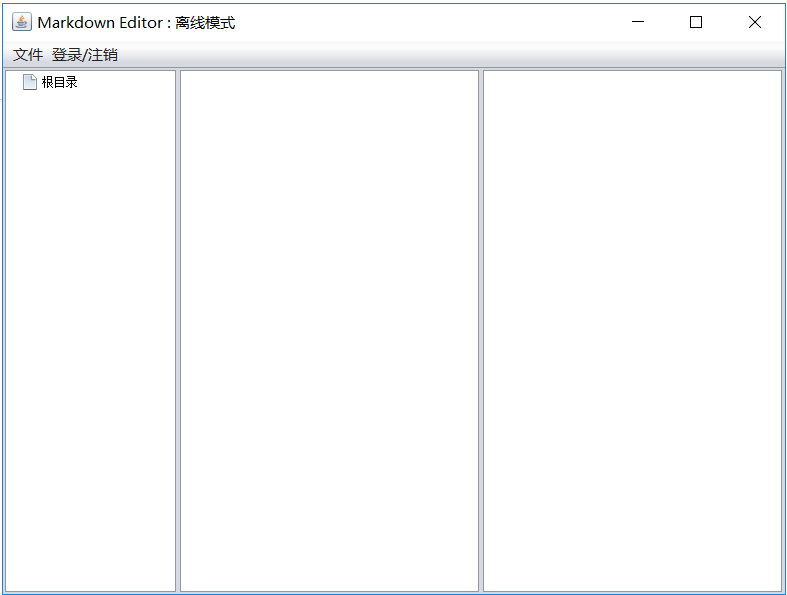




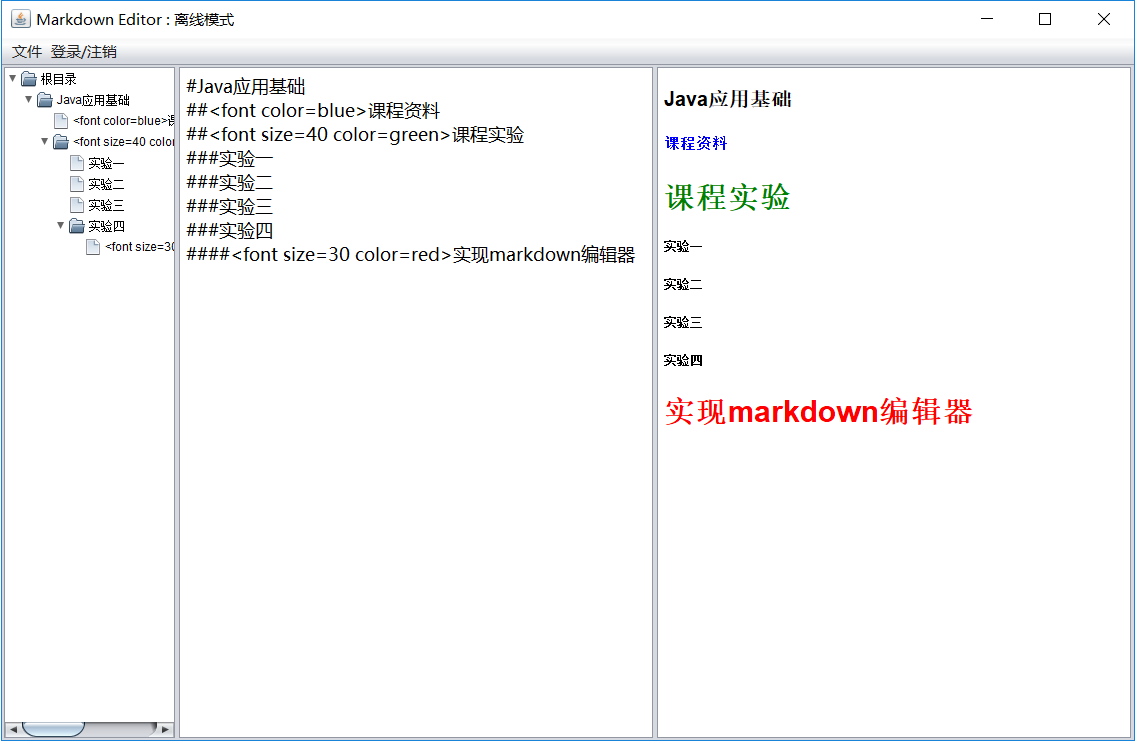
1. 实验截图

离线状态：

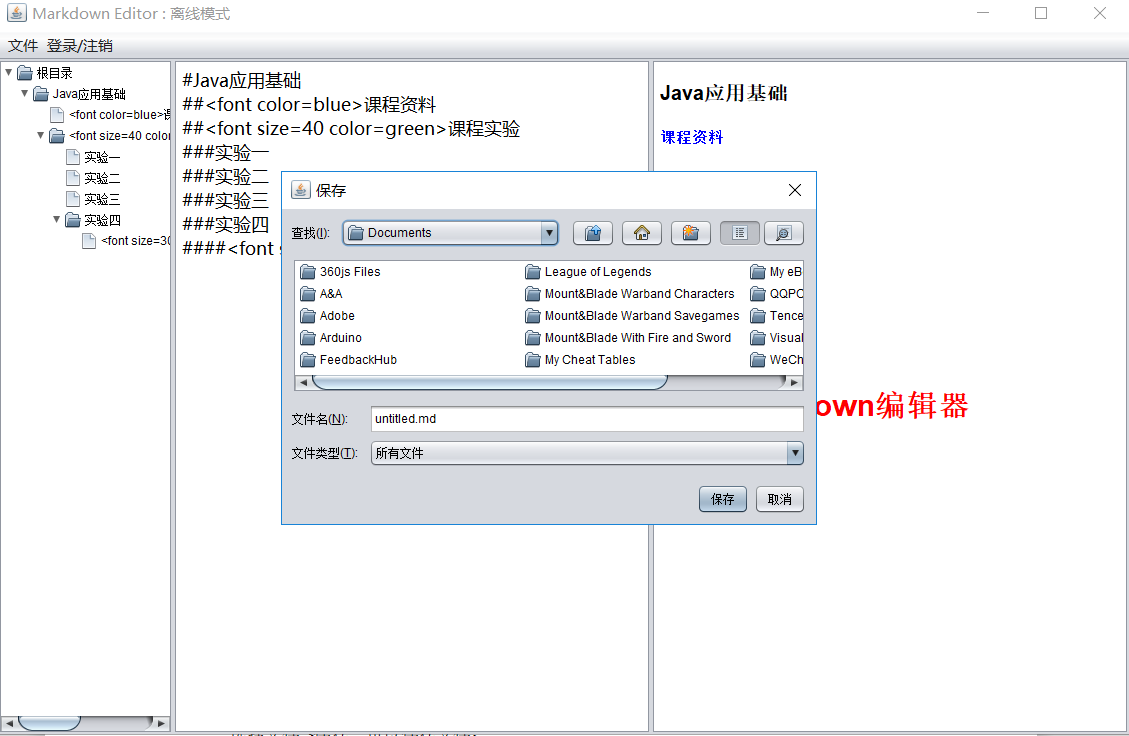
主页面：



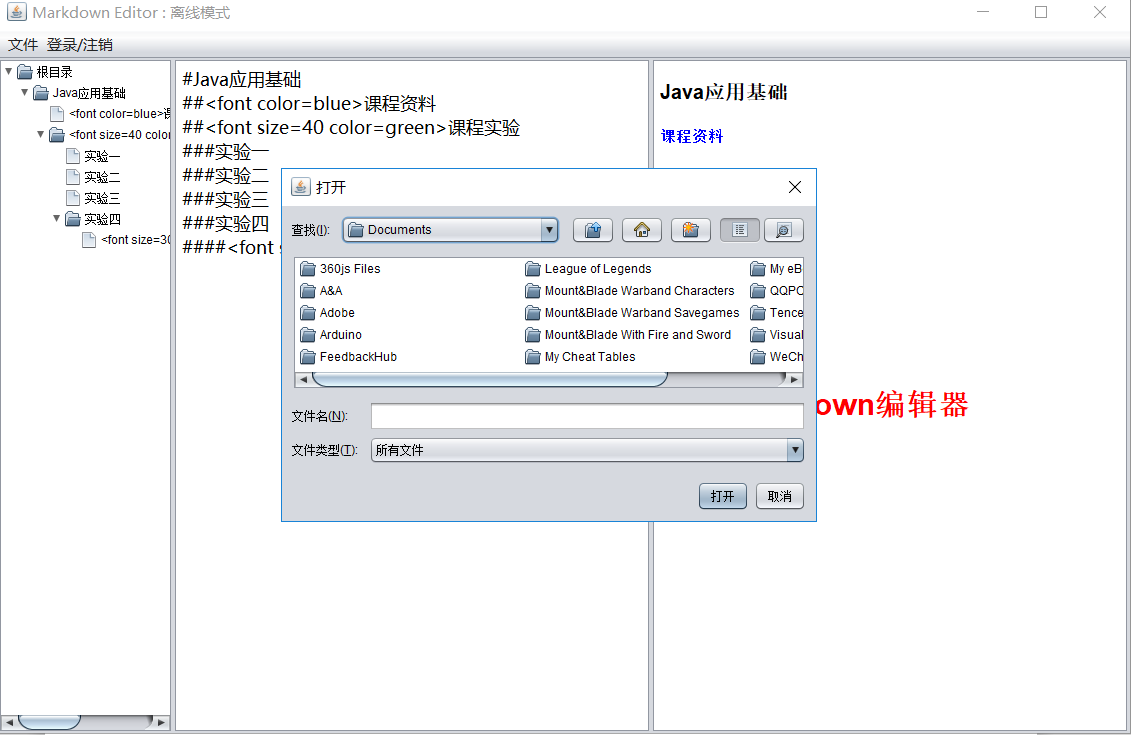
添加一些内容，显示目录树结果与修饰结果：



选择文件->保存，可以保存文件：



选择文件->打开，可以读取文件：

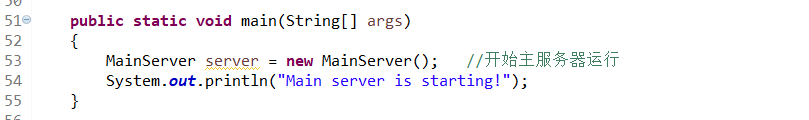


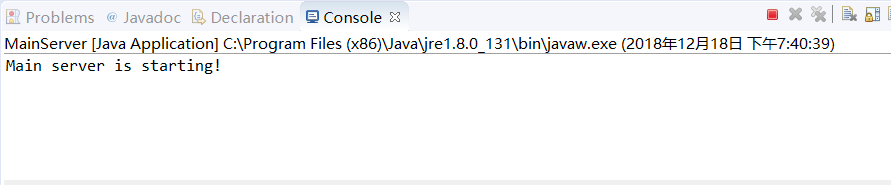
选择文件->导出为HTML可以保存为HTML文件：



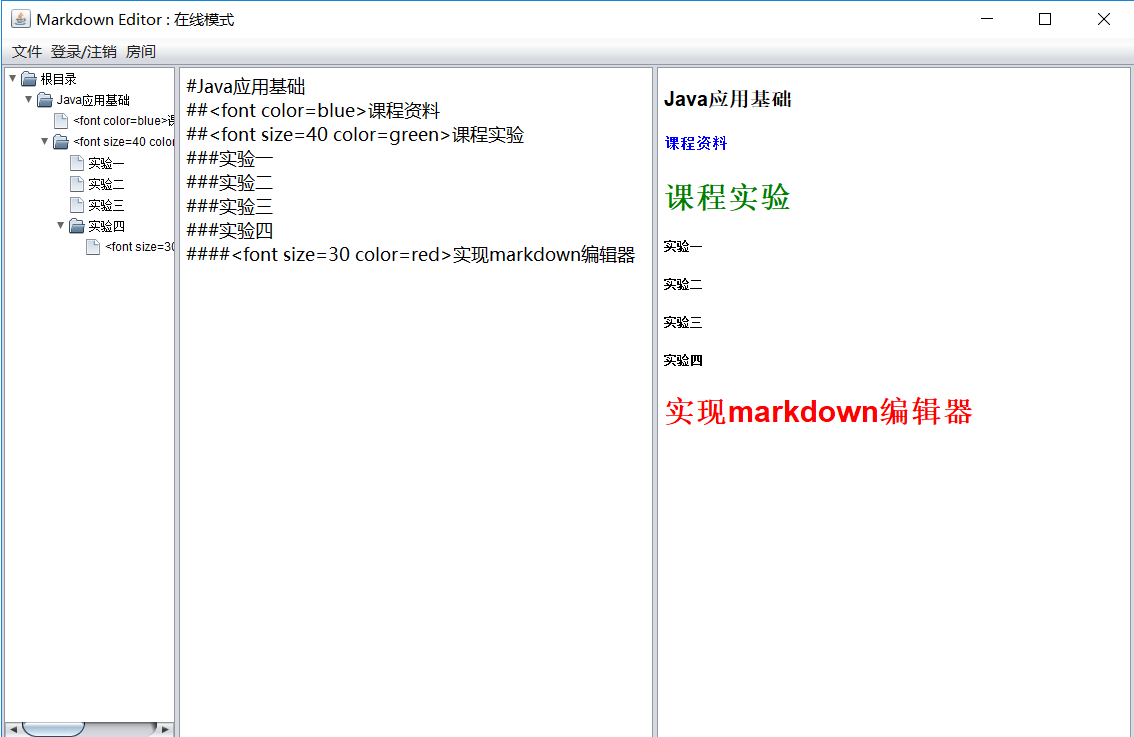
在线状态：

首先需要启动服务器：

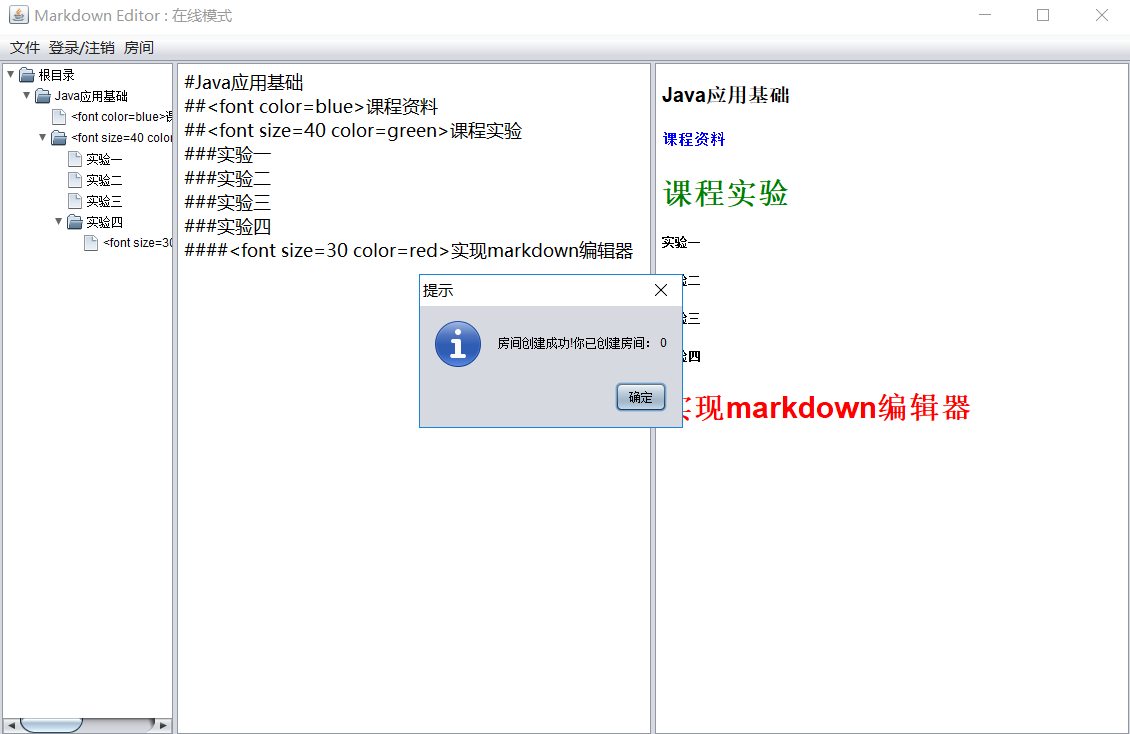




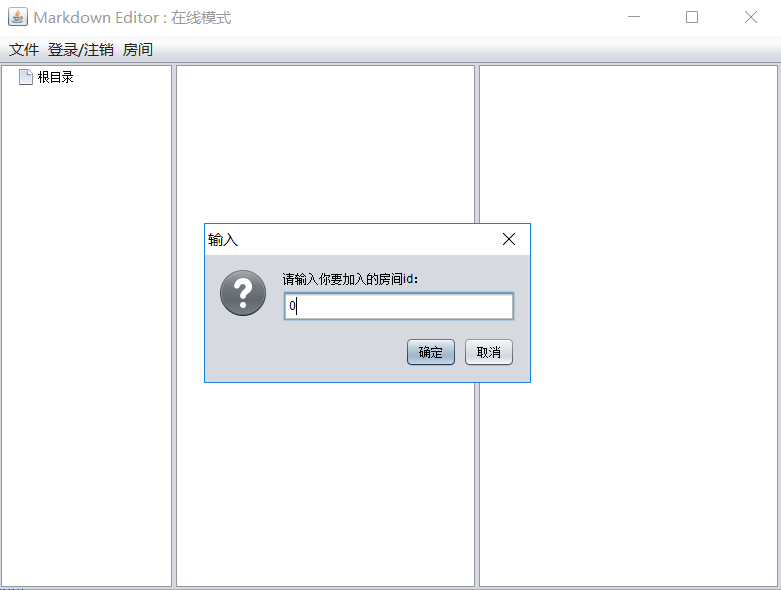
在客户端上点击登录按钮切换为在线模式：



点击房间->创建房间，可以创建一个新房间：

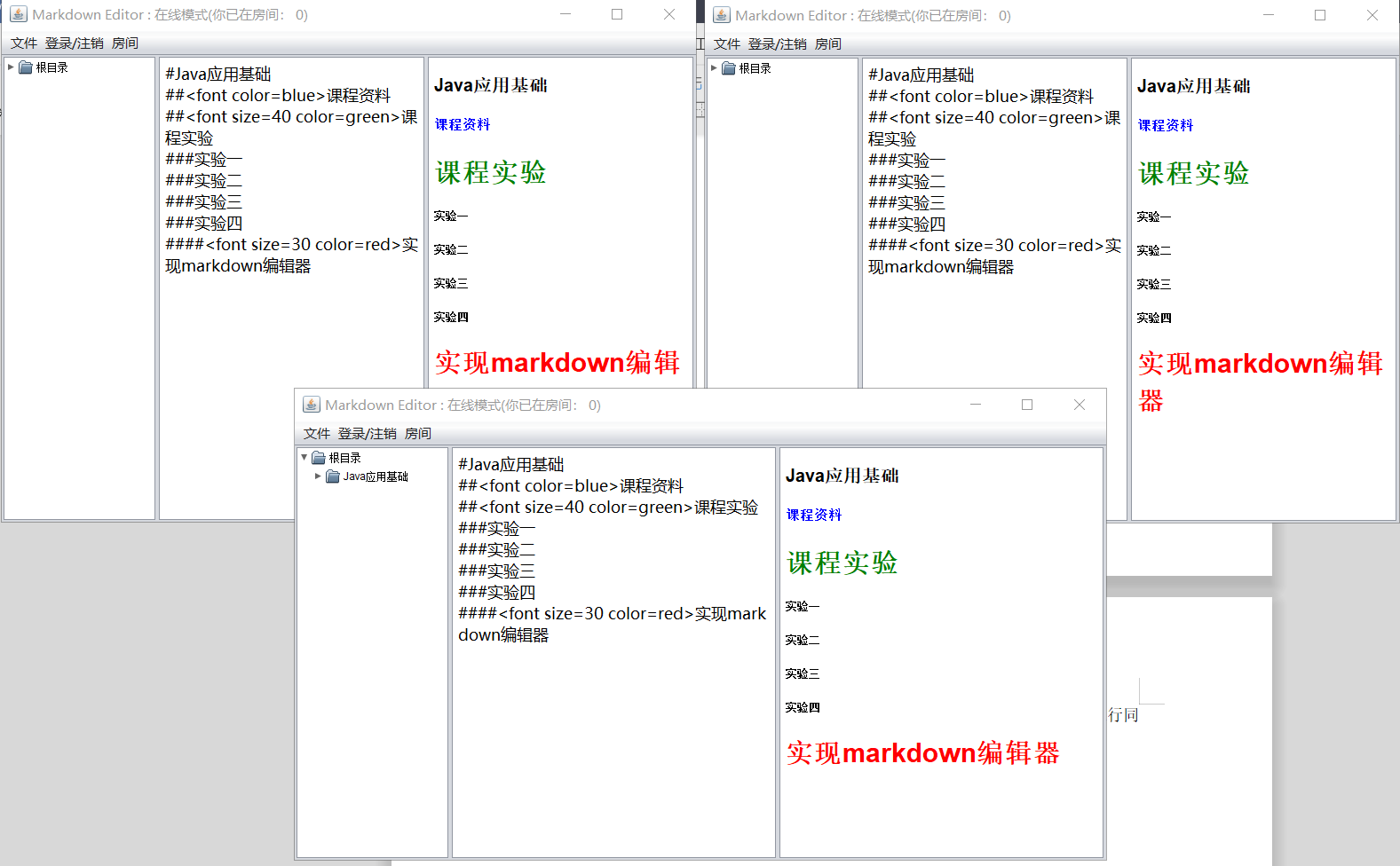


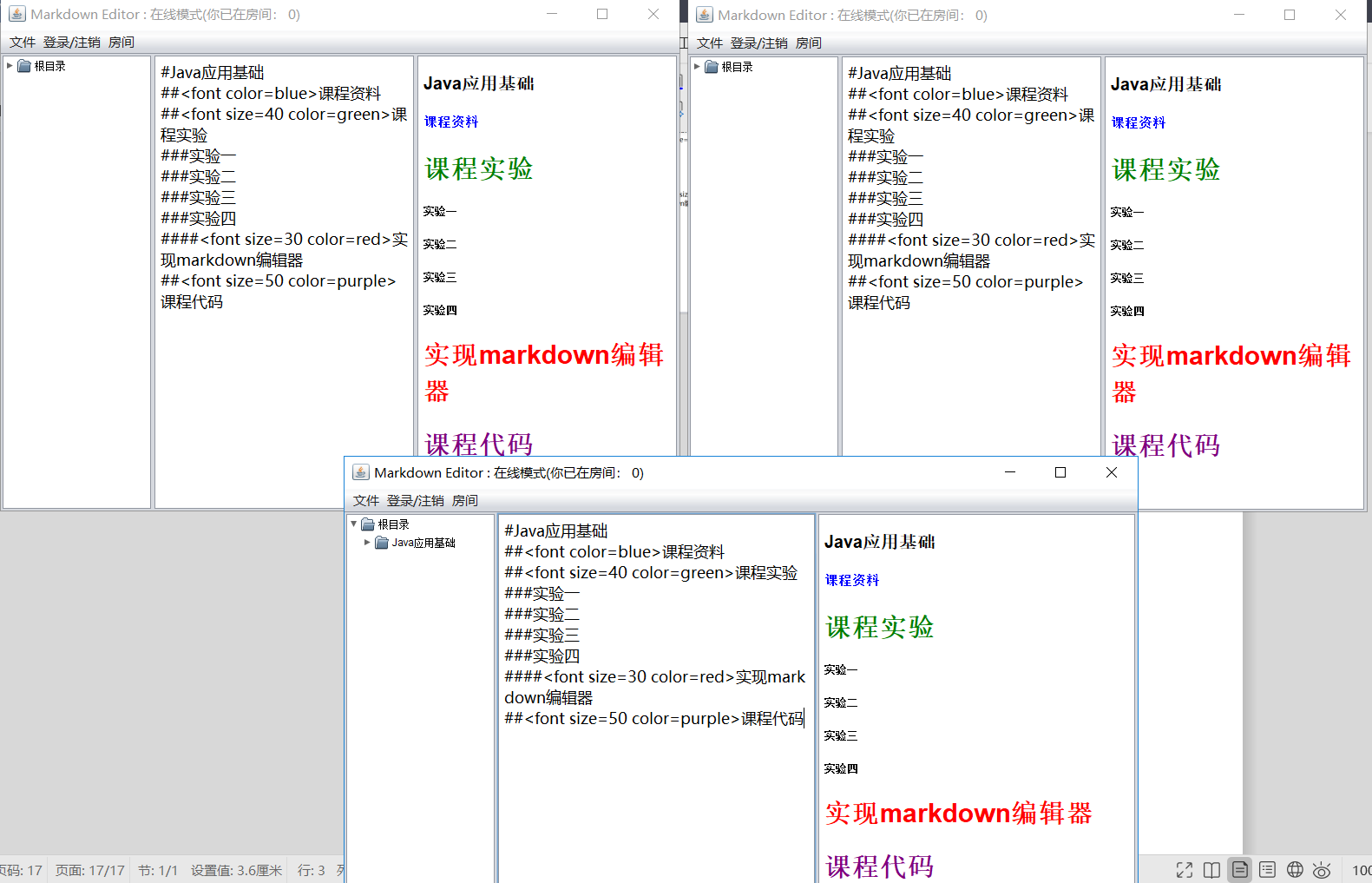
创建大量客户端，登录之后选择房间->加入房间，输入正确房间号之后可以加入房间：





每一个客户端都可以共同对内容进行修改，修改的内容会在其他客户端上进行同步：





在登录状态下点击注销，会退出在线模式，返回离线模式：

