练习1

在开始本练习课程之前，请确保您已阅读

课程作业。此练习所需的一些信息，例如保存用户和

文本文件中的消息数据在此处进行了描述，但在本文档中缺失。在这个练习环节中，你将

实现聊天系统的前两个版本：ChatSys 0.1和ChatSys 0.2。

配置IntelliJ

在开始编写任何代码之前，让我们配置IntelliJ。首先使用Java8创建一个新的Maven项目。（如果

Java8没有安装，您可以单击“Download JDK”并从AdoptOpenJDK OpenJ9中选择Java8。），然后单击

下一步，把你的项目命名为ChatSys。在“工件坐标”中，可以用

“programming3”，artifactid为“ChatSys”，版本为“0.1”。

在开始之前，创建一个包“programming3.chatsys”。你应该把你所有的课程都放在那个包里

或者是一个包裹。

同时在IntelliJ项目中初始化git存储库。可以使用命令行或使用

IntelliJ的图形用户界面。理想情况下，您应该定期提交您的更改，以便保存您的进度并可以继续

如果需要，向后。

ChatSys 0.1-发送消息

聊天信息

让我们首先在包“programming3”中创建类ChatMessage。chatsys.数据“那将能够

阅读并保存聊天信息。创建类及其属性、getter方法和要初始化的构造函数

他们。同时实现toString、equals和hashCode方法。您可以使用IntelliJ为

你。

完成后，添加两个空方法：

•format（）：字符串

•parse（字符串格式）：void

format方法将返回表示对象的格式化字符串，而parse方法将设置

属性指定为参数的格式化字符串的值。暂时，为

格式化方法，并将解析方法保留为空。

创建一个测试类以便为ChatMessage编写单元测试：ChatMessageTest。为创建测试方法

格式和解析方法。这些可以用IntelliJ自动创建。

实现创建的测试方法。运行测试。这些应该会失败，因为您还没有实现

方法还没有。

现在实现format和parse方法。运行你的测试。如果他们不通过，就意味着有一个错误

你的方法，你应该修正它们。继续进行，直到所有的测试都通过，并且您对方法有信心

工作正常。

现在创建一个方法save。该方法以文件作为参数，并将对象保存到该文件中。制造

确保文件以“附加”模式打开。这样可以防止文件每次都被完全删除

我们在里面写字！您可以通过如下方式首先创建一个文件写入程序：“new FileWriter（filename，true）”。那么

您可以将文件写入程序传递给PrintWriter。

别忘了为save方法编写单元测试。在测试内部，将结果保存到一个文件中

“消息\_测试.txt”. 执行测试后，不要忘记删除临时文件。你能做到的

通过在测试类中定义一个方法“clean”，并使用注释@AfterAll或@AfterEach。

使用git创建第一次提交以保存进度。

数据库

创建一个类“数据库”。目前，该类将有一个属性引用

消息数据库。这个类有一个公共方法：readMessages并返回ChatMessages列表。会的

通过读取数据库文件获得。目前，只需返回null。

为readMessages方法创建一个新的测试类“DatabaseTest”和一个测试方法。实施测试

方法。运行测试类，测试应该失败。

现在实现这个方法。使用BufferedReader读取每行文本并使用parse解析它们

聊天信息的方法。

向数据库添加一个addMessage方法，该方法将ChatMessage作为参数并将其添加到数据库中

文件。添加的聊天消息的id应大于中所有聊天消息的id

数据库。如果没有，该方法应该抛出一个异常。

我们的消息文本文件的一个问题是，如果在任何

文本字段。此外，用户名只能包含字母、数字和下划线字符。添加

如果要分析的字符串或

字符串类属性的格式不正确。添加一个单元测试来检查

当输入无效时，方法失败（即引发异常）。

做一个新的承诺来保存你的进度。

命令行界面（CLI）

现在在包“programming3”中创建两个类。聊天系统.cli使用静态方法“main”将

用作命令行界面来读取和保存消息。

SendMessages执行以下操作：

•

练习2和3

在本练习中，将有两个主要任务：

•实现聊天系统的多线程版本。

•重构上一个练习中的代码，以改进其结构以适应

在下一个练习中更改ChatSys。

除了学习重构和线程，这些任务的目标也是让您练习使用

接口和抽象类是Java中两个重要的面向对象的概念，

尤其是线程和网络编程。

在Moodle上可以找到多个类图，每个版本的ChatSys都有一个类图：

https://moodle.njit.edu.cn/moodle/mod/folder/view.php？内径=2218

注意：您不需要在练习第2课时（23日星期五）结束前实施所有这些更改。

在练习3（星期二27日）中，尽可能多地实施

你会继续工作的

从你完成的地方。另外，你不应该在第三堂课之前学习ChatSys 0.4.4和ChatSys 0.5

（26日星期一）因为这两个练习所涉及的主题还没有介绍，下一个将会介绍

演讲。

ChatSys 0.4.0版

对于这两个练习会话，主要目标是实现一个不同线程可以

通过“服务器线程”彼此通信。在这个版本的ChatSys中，没有

网络通信。相反，我们将有不同的线程通过

消息队列。

客户机将通过在消息队列中添加消息来向服务器发送消息。当服务器接收到

队列中的消息，它将消息转发给所有客户端。客户机也有一个消息队列。它被使用了

由服务器向客户端发送（即转发）消息。客户端不能直接与每个客户端通信

其他的则必须通过服务器的中间层进行通信。

消息队列

因为服务器和所有客户机都有自己的消息队列，所以我们将从创建一个类开始

新包编程中的消息队列3。chatsys.threads公司.

消息队列具有queue<ChatMessage>类型的属性“queue”。队列是由

Java标准API。（LinkedList是此类队列的实现。）

消息队列将有两种方法：

•send（ChatMessage message），向队列中添加聊天信息

•ChatMessage getMessage（int waitTime），从队列返回消息

getMessage的特殊性在于，如果队列中没有消息，它将等待一定的时间

然后再次尝试从队列中读取。要读取消息，可以使用队列的轮询或删除

方法。如果消息被成功读取，您应该返回它，否则您应该等待一定的时间

以毫秒为单位在参数中提供。（您可以使用方法等待线程睡眠.)

ThreadClient和ThreadServer

现在让我们创建两个继承自MessageQueue的类：ThreadClient和ThreadServer。

这两个类需要实现类“Runnable”，以便可以在单独的线程中运行。

在每个具有无限循环的类中编写run方法，并在循环的每次迭代中执行以下操作

•使用MessageQueue中定义的getMessage方法从队列中读取消息。

•捕捉调用getMessage通常引发的潜在InterruptedException并中断循环

如果线程中断，它可以正常结束

成功检索邮件时，服务器将执行以下操作：

•将信息添加到数据库

•将消息转发给其他客户机（创建一个您将实现的方法“forwardMessage”）

稍后）

当客户机成功检索到消息时，只需使用System.out.println.

为了使服务器和客户机能够相互通信，我们需要一种方法让它们

注意彼此。为此，客户机应该有一个类属性来存储对服务器的引用，该属性将

在构造函数中提供。我们还将为客户机提供一个类属性名，它是一个字符串，并具有

关联的getName方法，这样我们可以更容易地识别客户端。

然后服务器需要为数据库接口提供一个class属性，该属性将在其

构造函数和一个空的客户端集（即set<ThreadClient>类型）。它还需要一个方法

“register”允许客户端向服务器注册自己（即添加到客户端集合）和方法

取消注册以执行相反的操作（即从客户端集中删除自身）。

现在，我们还可以实现方法forward，它将在集合的所有客户端部分循环，并发送

他们要转发的消息。

最后，当我们启动客户机时，在进入等待消息的循环之前，我们的客户机需要注册

它自己和服务器。我们还将让我们的客户发送一条信息“你好，世界！“在注册

服务器。同时确保

练习4

本练习的目标是创建一个TCP服务器，用于向客户机提供请求。在第一部分，我们将

创建一个服务器并用telnet进行测试，telnet是一个连接到TCP服务器并发送虚拟文本终端的程序。

别忘了还要读作业的一般说明

(https://moodle.njit.edu.cn/moodle/mod/resource/view.php？id=2158）。

请记住，虽然与其他学生讨论是可以接受的，但提交的解决方案必须

个人和剽窃是不可接受的。如果你重复使用你不是自己写的代码片段，

记住用英语提到它们的来源，在代码中使用代码注释。然而，

请记住，您仍然应该是您提供的解决方案的主要作者。

远程登录

首先确保您可以在您的计算机上使用telnet。如果您使用的是Windows 10，telnet将被禁用

默认值，但可以这样启用：

https://social.technet.microsoft.com/wiki/contents/articles/38433.windows-10-enabling-telnet-client.aspx

另一个（可能更好的）Windows替代方案是使用PuTTY，即开源SSH和Telnet

Windows客户端。如果不想安装PuTTY，可以下载一个只包含

Telnet客户端：

•32位版本：https://the.earth.li/~sgtatham/putty/latest/w32/puttytel.exe

•64位版本：https://the.earth.li/~sgtatham/putty/latest/w64/puttytel.exe

（这些文件也可以在Moodle上找到）

如果您使用的是MacOS，最新版本默认不再包括telnet，需要安装它（请参阅

例如：https://osxdaily.com/2018/07/18/get-telnet-macos/)

如果您使用的是Linux，则非常像可以使用包管理器安装telnet（例如“sudo apt install

“telnet”在Ubuntu或Debian上）

（在Linux和MacOS上，另一种选择是使用nc（netcat），请参见示例：

https://www.igorkromin.net/index.php/2018/07/12/macos-has-a-much-better-tool-than-telnet-for-testing?远程服务器连接/）

本练习中的说明将假定您将Windows与PuTTY Telnet客户端一起使用。

TCP服务器

创建一个新的包程序3。聊天系统.tcp你将把所有与TCP相关的类放在其中

服务器和客户端。

首先，我们将创建一个基本的HTTP服务器。在Java中，我们可以使用java.net.ServerSocket.

创建具有以下属性的TCPChatServer类：

•带有服务器侦听端口的整数

•带TCP会话超时的整数

•数据库对象

•布尔值，如果服务器正在运行，则为真

•服务器插座

创建一个构造函数来初始化属性并添加两个公共方法start和stop。

方法start用于启动服务器（即初始化ServerSocket），并且应该在循环中接受

通过调用ServerSocket上的accept方法进行连接。方法“accept”返回一个套接字对象。

使用setSoTimeout方法设置套接字的超时，以便在发生以下情况时自动关闭套接字：

在一段时间后没有收到客户的任何信息。然后我们来处理连接

使用具有新线程和TCPChatServerSession的客户端（见下文）。

stop方法只需将服务器的布尔值设置为false以关闭循环并调用该方法

服务器插座上的“关闭”。

TCPChatServer会话

TCPChatServerSession是实现Runnable的类。它有两个属性在

施工单位：

•一个数据库对象（与TCPChatServer使用的对象相同）

•一个套接字对象（当服务器接受与客户机的连接时获得的对象）

由于该类是可运行的，因此需要实现方法“run（）”。该方法执行以下操作：

•从套接字的输出流创建一个BufferedWriter（使用方法获得

getOutputStream）

•从BufferedReader方法获取的

获取输入流）

•从BufferedReader读取一行

•解析ChatSys 4.0协议定义的行

•根据从客户端收到的协议消息采取一些措施。

协议

服务器可以从客户端接收以下消息。

•“OK\r\n”：客户端可以发送OK消息，使服务器保持活动状态（即阻止连接

因为超时而关闭）。服务器会忽略这些消息。

•“GET recent messages<N>\r\N”：客户端发送此消息以获取N条最新消息。

服务器用一个N条消息的列表进行响应，其中包括作者的用户名

时间戳和消息。它将以下消息发送回客户端

o“MESSAGES<N2>\r\n”告诉客户端要读取的消息数。N通常与

N2，但如果数据库中的消息数小于N，则than N2是

数据库中的邮件。

o“消息<username><tim