厦門大學

软件学院

《中间件技术》实验报告

题　　目 中间件技术实验一

姓　　名 陈澄

学　　号 32420212202930

班　　级 软工三班

实验时间 2024/2/29

2024　年　2　月　29　日

1 实验目的

1） 理解中间件的基本概念
2） 理解消息型中间件的基本概念
3） 掌握基于中间件技术进行简单开发的基本过程。

2 实验环境

操作系统：Windows11

编译环境：Intellij IDEA

平台技术：JMS（JavaEE 开发平台）

3 实验题目

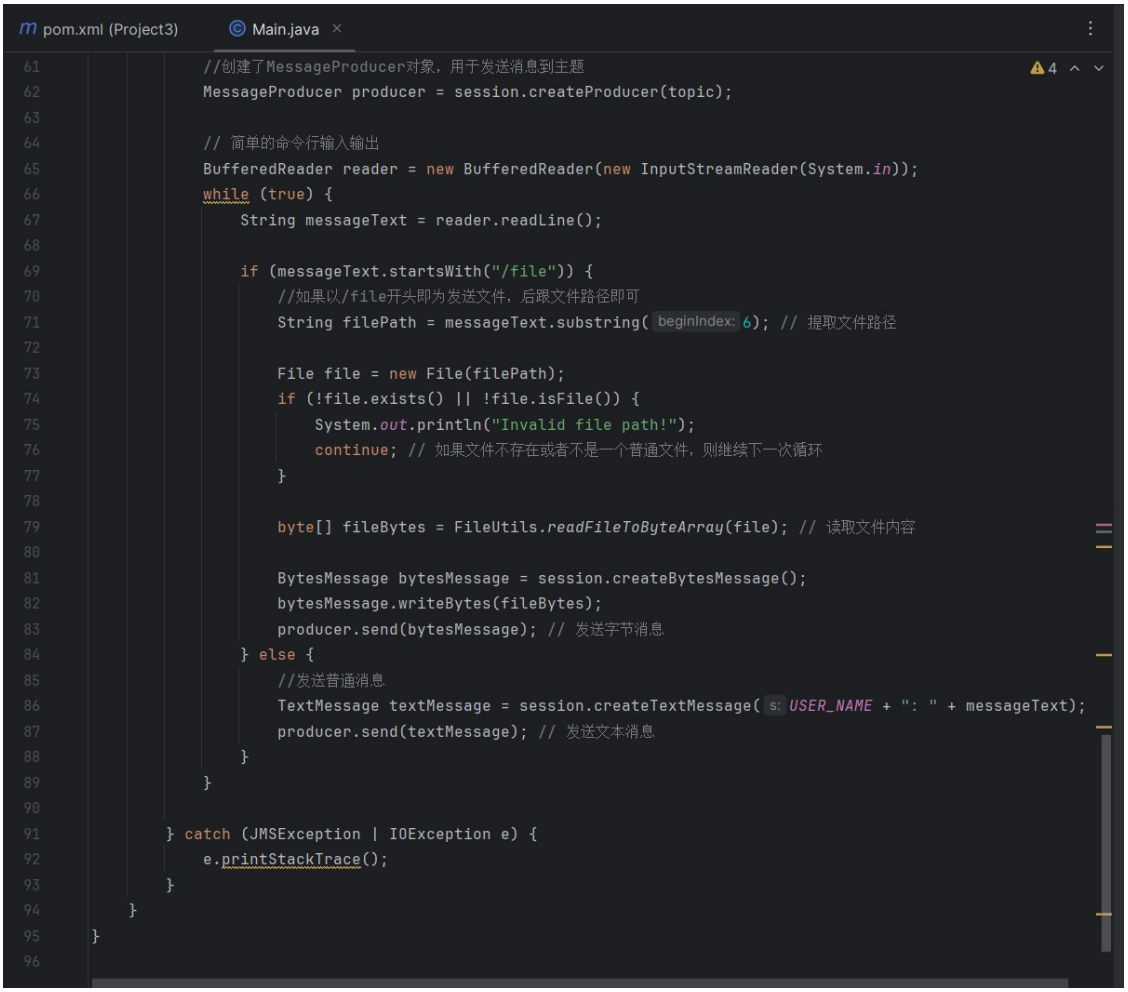
Step1.不依赖于中间件概念（完全基于你的现有知识），完成一个软件设计（只要设计，不需要编码实现），实现 A 机器发送消息给 B 机器，并在 B 机器上显示的功能。此为第一版设计。
Step2.从中间件的角度，分析步骤 1 软件设计的优缺点，从架构、工作量、标准化、跨异构能力等角度提出若干改进点，并给出理由。完成改进版设计，此为第二版设计。
Setp3.自学消息型中间件 Queue 和 Topic 的简单概念（大约 10-15 分钟），选择你认为合适的模型，改进步骤 2 的设计，并完成第三版设计（写出理由）。
Step4.学习使用 JMS（JavaEE 平台）或 MSMQ（微软平台）（也可以选用其它中间件产品），设计开发一个即时通讯程序（重点在核心功能）。（除简单的文字通信功能外，要求实现包括但不限于传输文件、watchdog、存储转发、群聊等诸多 feature 中的一个或多个）
此处跳过 step1-3，直接提交 step4 结果

4 代码展示

实现思路已经体现在注释中

pom.xml (Project3) Main.java

```java
import org.apache.commons.io.FileUtils;

import javax.jms.*;
import java.io.BufferedReader;
import java.io.File;
import java.io.IOException;
import java.io.InputStreamReader;

public class Main {
    //ActiveMq服务所在的地址和端口
    1 个用法
    private static final String BROKER_URL = "tcp://localhost:61616";
    //用户订阅的TOPIC，只有相同的TOPIC才能收到相同的消息（即群聊）
    1 个用法
    private static final String CHAT_TOPIC = "chatTopic";
    //用户的用户名，用于唯一辨识一个用户，后续也会作为ClientID使用
    3 个用法
    private static final String USER_NAME = "user1";

    public static void main(String[] args) {
        try {
            //与ActiveMq建立连接
            ConnectionFactory connectionFactory = new ActiveMQConnectionFactory(BROKER_URL);
            Connection connection = connectionFactory.createConnection();
            connection.setClientID(USER_NAME);
            connection.start();

            //设定消息确认模式（此处是自动确认）和TOPIC
            Session session = connection.createSession( b: false, Session.AUTO_ACKNOWLEDGE);
            Topic topic = session.createTopic(CHAT_TOPIC);

            //创建消息监听事件，便于收到其他用户消息
            MessageConsumer consumer = session.createDurableSubscriber(topic, USER_NAME);
            consumer.setMessageListener(message -> {
                if (message instanceof TextMessage) {
                    //如果是TextMessage即为普通聊天消息
                    try {
                        String receivedMessage = ((TextMessage) message).getText();
                        System.out.println("Received: " + receivedMessage);
                    } catch (JMSException e) {
                        e.printStackTrace();
                    }
                } else if (message instanceof BytesMessage) {
                    //如果是BytesMessage即为文本消息
                    try {
                        BytesMessage bytesMessage = (BytesMessage) message;
                        byte[] fileBytes = new byte[(int) bytesMessage.getBodyLength()];
                        bytesMessage.readBytes(fileBytes);

                        // 指定保存文件的路径，此处以文本文件为例
                        String filePath = "C:\\Users\\CC507\\Desktop\\output\\2.txt";
                        FileUtils.writeByteArrayToFile(new File(filePath), fileBytes);

                        System.out.println("File received and saved: " + filePath);
                    } catch (JMSException | IOException e) {
                        e.printStackTrace();
                    }
                }
            });

            //创建了MessageProducer对象，用于发送消息到主题
```



5 实验结果

（命令行的红字是没有设置日志实现所致，并不影响运行）

（发送方和接收方的代码一致，但是需要保证 USER_NAME 不同，两者导致仍然可以运行）

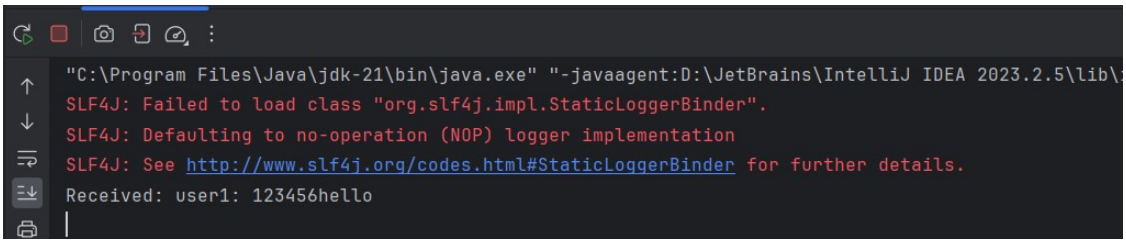
（需要保证 ActiveMq 服务正在运行，此处在本机上运行，实际投入生产需要更改到正确的地址）

功能 1：群聊。

所有订阅了相同 TOPIC 的用户都将收到相同的消息

发送方：

接收方：（这里只以一个用户为例，实际上多个接受方收到的也相同）



功能 2：存储转发。

我使用了持久订阅者，意味着消息在 ActiveMq 上将会以队列的形式组织，并存储起来。存储后，中间节点会负责将消息从发送方传递到接收方。这个过程可以包括消息的路由、传输和最终投递给接收方。

```
//创建消息监听事件，便于收到其他用户消息
MessageConsumer consumer = session.createDurableSubscriber(topic, USER_NAME);
```

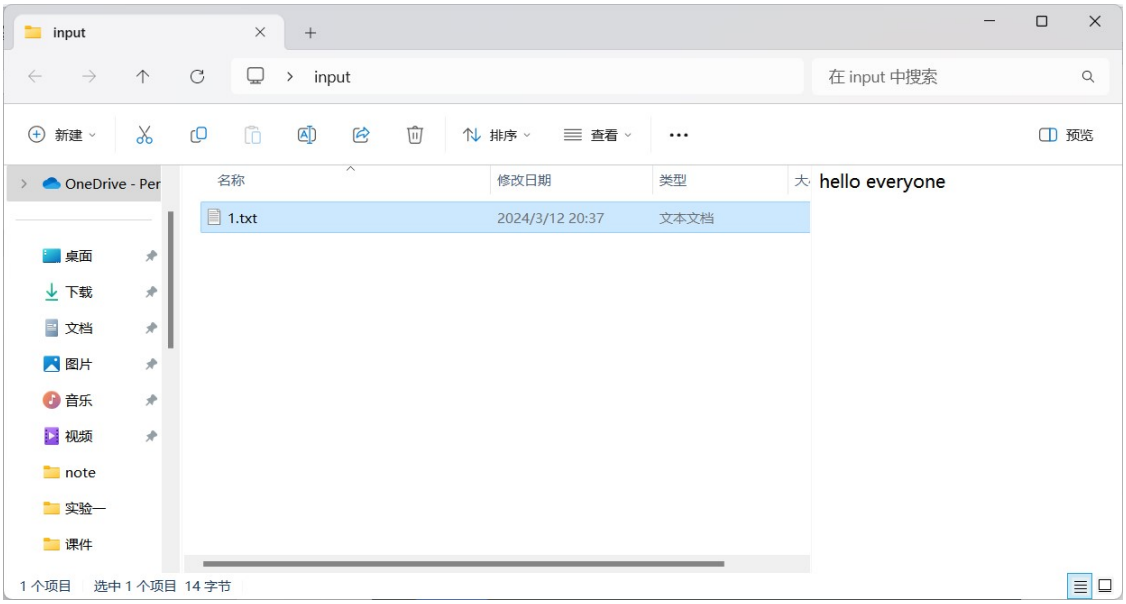
具体的实现效果是：将接收方的程序关闭，发送方发送一个消息，再启动接收方的程序，仍然可以收到消息

功能 3：文件传输。

我使用字节消息和文本消息来区分文件消息和普通消息，当收到字节消息的时候以文本方式处理并存储。

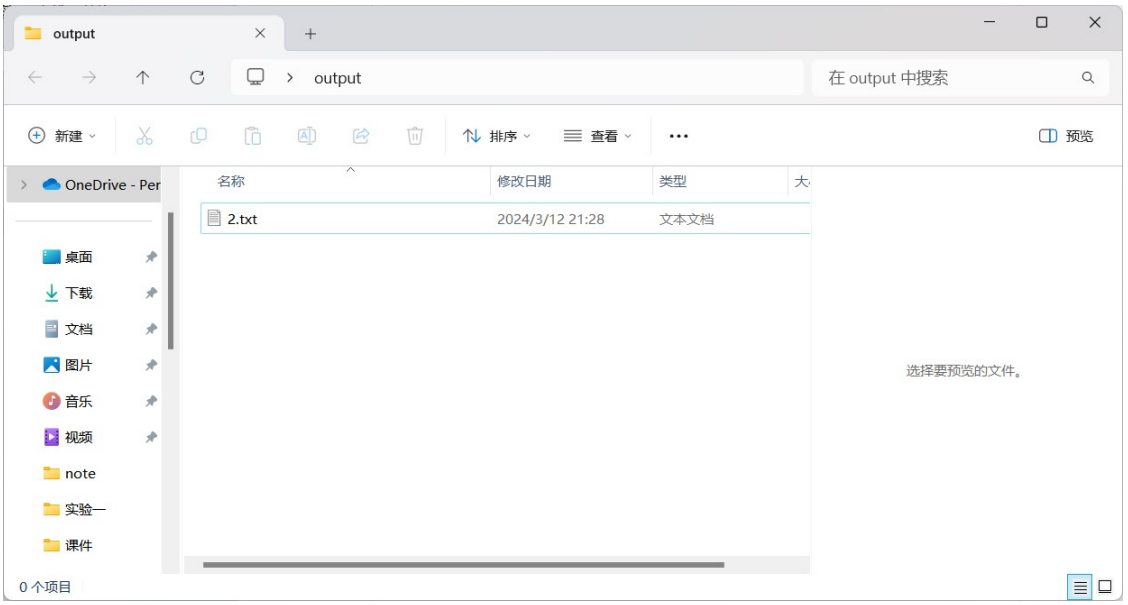
此处我将路径指定在代码文件中，实际投入生产可以更改使之更加灵活。

发送方：



接收方：


运行
Main
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:D:\JetBrains\IntelliJ IDEA 2023.2.5\lib\ide
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Received: user1: 77789
File received and saved: C:\Users\CC507\Desktop\output\2.txt


output
output
在 output 中搜索
新建
排序
查看
预览
OneDrive - Per
名称
修改日期
类型
2.txt
2024/3/12 21:28
文本文档
桌面
下载
文档
图片
音乐
视频
note
实验一
课件
选择要预览的文件。
0 个项目