Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра экономической информатики

Лабораторная работа №4

**Технология передачи данных Message driver bean**

Выполнила:

студентка группы 714301

факультета КП

Забелло К.В.

Проверил:

Кунцевич А.А

Минск 2020

**Задание:** Клиентское приложение отправляет несколько (не менее 10) сообщений в QUEUE или TOPIC (нечетный вариант – QUEUE, четный – TOPIC). MDB получает и обрабатывает сообщения следующим образом:

11) Получатель удаляет из файла полученные сообщение.

**Текст программы:**

Класс MessageBean

package com.zabello.beans;  
  
import javax.ejb.ActivationConfigProperty;  
import javax.ejb.MessageDriven;  
import javax.jms.JMSException;  
import javax.jms.Message;  
import javax.jms.MessageListener;  
import javax.jms.TextMessage;  
import java.io.FileReader;  
import java.io.FileWriter;  
import java.io.IOException;  
import java.util.Scanner;  
import java.util.logging.Logger;  
  
@MessageDriven(activationConfig = {  
 @ActivationConfigProperty(propertyName = "destinationLookup", propertyValue = "jms/Queue")  
 , @ActivationConfigProperty(propertyName = "destinationType", propertyValue = "javax.jms.Queue")  
})  
public class MessageBean implements MessageListener {  
 static final Logger *logger* = Logger.*getLogger*("MessageBean");  
  
 @Override  
 public void onMessage(Message message) {  
  
 TextMessage msg;  
  
 try {  
 if (message instanceof TextMessage) {  
 msg = (TextMessage) message;  
 deleteFromFile(msg.getText());  
  
 } else {  
 *logger*.warning("Wrong type");  
 }  
 } catch (JMSException e) {  
 e.getMessage();  
 }  
 }  
  
 private void deleteFromFile(String message) {  
 StringBuilder messages = new StringBuilder();  
  
 try (FileReader reader = new FileReader("E:/вучоба/РИС/labs/ris4/messages.txt");  
 Scanner scanner = new Scanner(reader)) {  
  
 while (scanner.hasNextLine()) {  
 String str = scanner.nextLine();  
 if (!str.equals(message))  
 messages.append(str).append("\n");  
 }  
  
 } catch (IOException e) {  
 e.getMessage();  
 }  
 try (FileWriter writer = new FileWriter("E:/вучоба/РИС/labs/ris4/messages.txt", false)) {  
 writer.write(messages.toString());  
 } catch (IOException e) {  
 e.getMessage();  
 }  
 }  
}

Класс MessageServlet

package com.zabello.servlets;  
  
import javax.annotation.Resource;  
import javax.jms.\*;  
import javax.servlet.ServletException;  
import javax.servlet.annotation.WebServlet;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
import java.io.IOException;  
  
@WebServlet(urlPatterns = {"/messages-servlet"})  
public class MessageServlet extends HttpServlet {  
 @Resource(mappedName = "jms/ConnectionFactory")  
 private ConnectionFactory connectionFactory;  
  
 @Resource(mappedName = "jms/Queue")  
 private Queue queue;  
  
 @Override  
 protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {  
 try {  
 Connection connection = connectionFactory.createConnection();  
 Session session = connection.createSession(false, Session.*AUTO\_ACKNOWLEDGE*);  
 MessageProducer messageProducer = session.createProducer(queue);  
  
 String userMessage = req.getParameter("message");  
  
 TextMessage message = session.createTextMessage();  
 message.setText(userMessage);  
 messageProducer.send(message);  
  
 } catch (JMSException e) {  
 e.getMessage();  
 }  
 }  
}

**Скриншоты работы:**





