Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра экономической информатики

Лабораторная работа №4

**Технология передачи данных Message driver bean**

Выполнил: Проверил:

Студент: гр. 714301

Александрова Е.И. Кунцевич А.А.

Минск 2020

**Цель –** разработать веб приложение для реализации прослушивания сообщений посредствам технологии MDB.

**Задание:** клиентское приложение отправляет несколько (не менее 10) сообщений в QUEUE. MDB получает и обрабатывает сообщения следующим образом:

Получатель сортирует сообщения в порядке возрастания и таким образом записывает их в текстовый файл.

разработать приложение на базе технологии MDB для реализации рассылки сообщений. При получении сообщения MDB выводит лог: если строка имеет знак «?», сообщение записывается в файл, иначе – нет.

**Текст программы:**

package com.aleksandrova.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.Arrays;  
import java.util.Comparator;  
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("C:/Users/Elizaveta Alex&drova/Downloads/Telegram Desktop/ris4/messages.txt");  
 Scanner scanner = new Scanner(reader)) {  
  
 int i=0;  
 String a[] = new String[10];  
 while (scanner.hasNextLine()) {  
 String str = scanner.nextLine();  
 a[i]=str;  
 i++;  
 }  
 a[i]=message;  
  
 Arrays.*sort*(a, new Comparator<String>() {  
 public int compare(String s1, String s2) {  
 return s1.compareTo(s2);  
 }  
 });  
  
 for (int j =0; j<a.length; j++){  
 if(a[j]!=null && a[j]!="") {  
 messages.append(a[j]).append("\n");  
 }  
 }  
 } catch (IOException e) {  
 e.getMessage();  
 }  
 try (FileWriter writer = new FileWriter("C:/Users/Elizaveta Alex&drova/Downloads/Telegram Desktop/ris4/messages.txt", false)) {  
 writer.write(messages.toString());  
 } catch (IOException e) {  
 e.getMessage();  
 }  
 }  
}

package com.aleksandrova.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:**







