ASSIGNMENT – 6

Publisher.java

package pubsub;

import javax.jms.\*;

import org.apache.activemq.ActiveMQConnection;

import org.apache.activemq.ActiveMQConnectionFactory;

public class Publisher {

private static String url = ActiveMQConnection.DEFAULT\_BROKER\_URL;

public static void main(String args[])throws JMSException, InterruptedException{

ConnectionFactory conFact = new ActiveMQConnectionFactory(url);

Connection con = conFact.createConnection();

con.start();

Session session = con.createSession(false, Session.AUTO\_ACKNOWLEDGE);

Topic topic = session.createTopic("PubSubMessage");

MessageProducer producer = session.createProducer(topic);

TextMessage message = session.createTextMessage();

int runs = 0, wickets = 0;

for(int i = 0; i < 120; i++) {

Thread.sleep(5000);

if(wickets == 10) break;

int[] score= {1,2,3,4,6,-1,0};

int index = (int)(Math.random()\*6);

int ball = score[index];

if(ball == -1)

{

wickets++;

}

else

{

runs+=ball;

}

message.setText(runs+"/"+wickets);

producer.send(message);

System.out.println("Match score: " + message.getText());

}

con.close();

}

}

Subscriber.java

package pubsub;

import javax.jms.\*;

import org.apache.activemq.ActiveMQConnection;

import org.apache.activemq.ActiveMQConnectionFactory;

public class Subscriber {

private static String url = ActiveMQConnection.DEFAULT\_BROKER\_URL;

public static void main(String args[]) throws JMSException, InterruptedException{

ConnectionFactory conFact = new ActiveMQConnectionFactory(url);

Connection con = conFact.createConnection();

con.start();

Session session = con.createSession(false, Session.AUTO\_ACKNOWLEDGE);

Topic topic = session.createTopic("PubSubMessage");

MessageConsumer consumer = session.createConsumer(topic);

for(int i = 0; i < 120; i++) {

MessageListener listener = new MessageListener(){

public void onMessage(Message message){

try{

if(message instanceof TextMessage){

TextMessage textMessage = (TextMessage) message;

System.out.println("Score Recieved: "+textMessage.getText());

}else{

System.out.println("No Message Recieved: ");

}

}catch(Exception e){

System.out.println("Exception in subscriber: "+e.toString());

}

}

};

consumer.setMessageListener(listener);

Thread.sleep(5000);

}

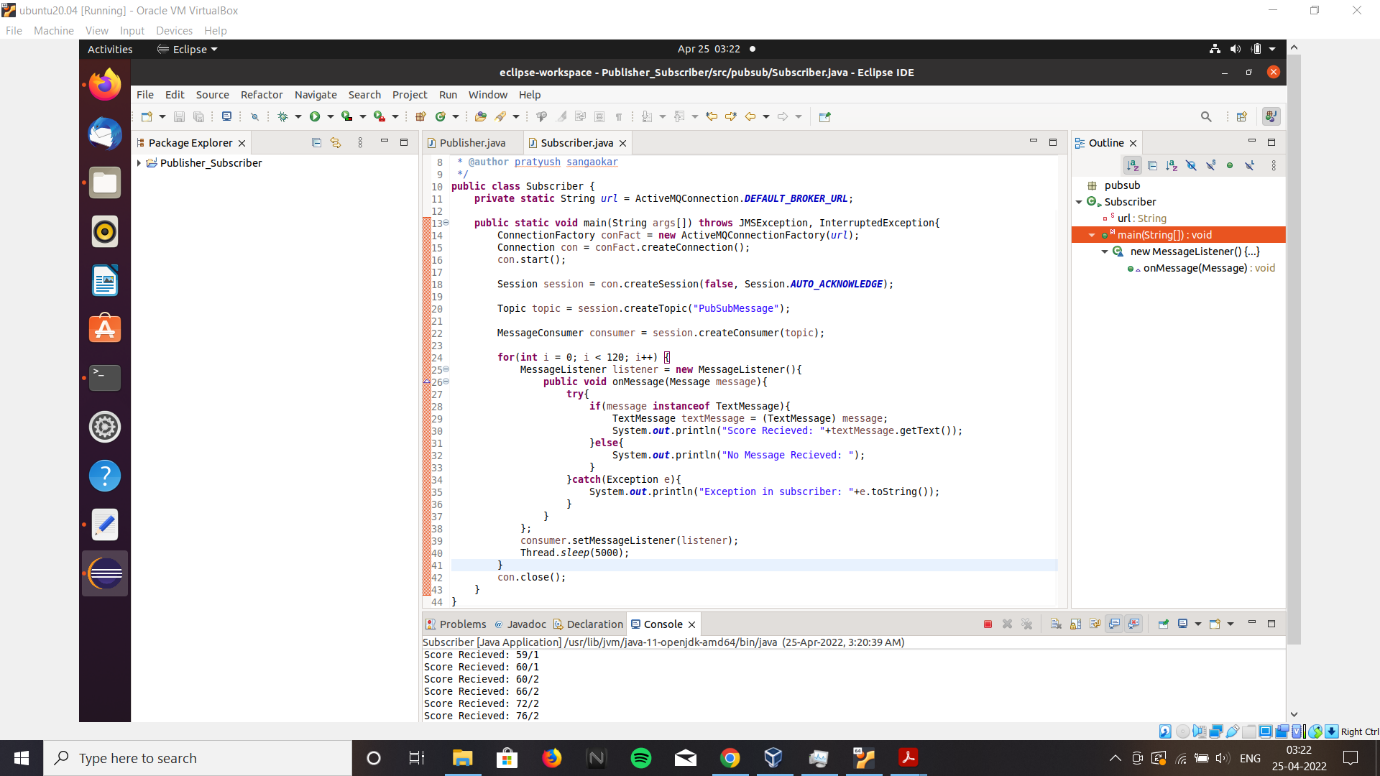
con.close();

}

}

OUTPUT-

Subscriber:



Publisher:

