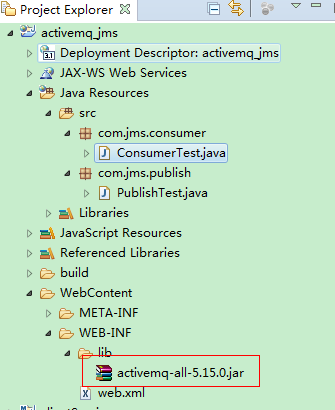
1. 搭建开发环境



点对点的消息模型，只需要一个消息生成者和消息消费者，下面我们编写代码。

1. 编写生产者

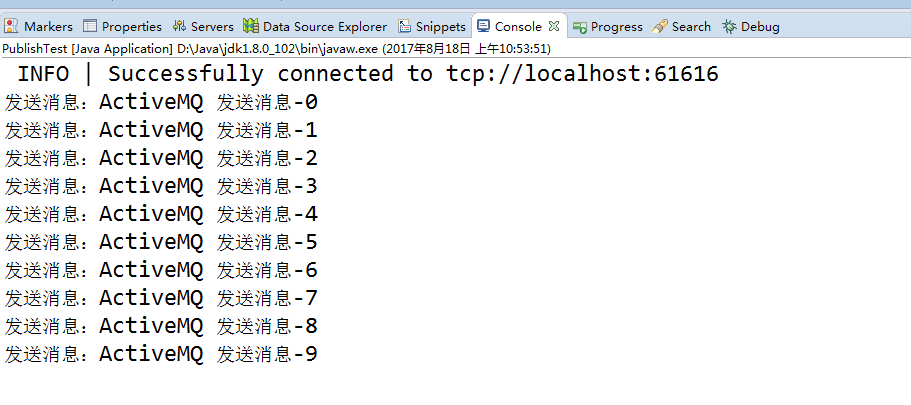
|  |
| --- |
| **package** com.jms.publish;  **import** javax.jms.Connection;  **import** javax.jms.ConnectionFactory;  **import** javax.jms.Destination;  **import** javax.jms.JMSException;  **import** javax.jms.MessageProducer;  **import** javax.jms.Session;  **import** javax.jms.TextMessage;  **import** org.apache.activemq.ActiveMQConnection;  **import** org.apache.activemq.ActiveMQConnectionFactory;  /\*\*  \*  \* **@name** PublishTest  \* **@description** 消息生产者  \* **@author** xupf  \* **@version** 1.0  \* **@date** 2017年8月18日  \*/  **public** **class** PublishTest {    //默认连接用户名  **private** **static** **final** String ***USERNAME*** = ActiveMQConnection.***DEFAULT\_USER***;    //默认连接密码  **private** **static** **final** String ***PASSWORD*** = ActiveMQConnection.***DEFAULT\_PASSWORD***;    //默认连接地址  **private** **static** **final** String ***BROKEURL*** = ActiveMQConnection.***DEFAULT\_BROKER\_URL***;    //发送消息的数量  **private** **static** **final** **int** ***SENDNUM*** = 10;    **public** **static** **void** main(String[] args) {  //连接工厂  ConnectionFactory connectionFactory;    //连接  Connection connection = **null**;    //会话 接受或者发送消息的线程  Session session;    //消息的目的地  Destination destination;    //消息生产者  MessageProducer messageProducer;    //实例化连接工厂  connectionFactory = **new** ActiveMQConnectionFactory(PublishTest.***USERNAME***,PublishTest.***PASSWORD***,PublishTest.***BROKEURL***);    **try** {  //通过连接工厂获取连接  connection = connectionFactory.createConnection();  //启动连接  connection.start();  //创建session  session = connection.createSession(**true**, Session.***AUTO\_ACKNOWLEDGE***);  //创建一个名称为HelloWorld的消息队列  destination = session.createQueue("HelloWorld");  //创建消息生产者  messageProducer = session.createProducer(destination);  //发送消息  *sendMessage*(session,messageProducer);    session.commit();      } **catch** (JMSException e) {  e.printStackTrace();  }    }  /\*\*  \* 发送消息  \* **@param** session  \* **@param** messageProducer  \* **@throws** JMSException  \*/  **private** **static** **void** sendMessage(Session session, MessageProducer messageProducer) **throws** JMSException {  **for**(**int** i=0;i<PublishTest.***SENDNUM***;i++){  //创建一条文本信息  TextMessage messge = session.createTextMessage("ActiveMQ 方式消息-"+i);  System.***out***.println("发送消息：ActiveMQ 发送消息-"+i);  //通过消息生产者发出消息  messageProducer.send(messge);  }  }  } |

1. 编写消费者

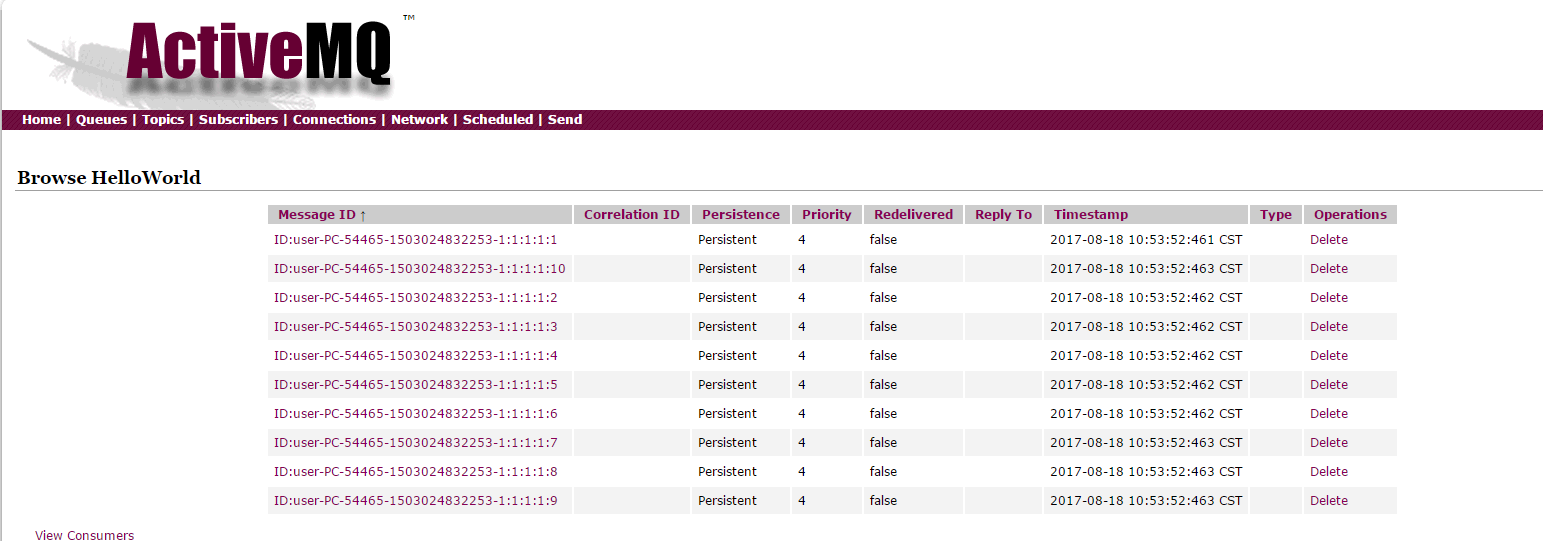
|  |
| --- |
| **package** com.jms.consumer;  **import** javax.jms.Connection;  **import** javax.jms.ConnectionFactory;  **import** javax.jms.Destination;  **import** javax.jms.JMSException;  **import** javax.jms.MessageConsumer;  **import** javax.jms.Session;  **import** javax.jms.TextMessage;  **import** org.apache.activemq.ActiveMQConnection;  **import** org.apache.activemq.ActiveMQConnectionFactory;  /\*\*  \*  \* **@name** ConsumerTest  \* **@description**  \* **@author** xupf  \* **@version** 1.0  \* **@date** 2017年8月18日  \*/  **public** **class** ConsumerTest {  // 默认连接用户名  **private** **static** **final** String ***USERNAME*** = ActiveMQConnection.***DEFAULT\_USER***;  // 默认连接密码  **private** **static** **final** String ***PASSWORD*** = ActiveMQConnection.***DEFAULT\_PASSWORD***;  // 默认连接地址  **private** **static** **final** String ***BROKEURL*** = ActiveMQConnection.***DEFAULT\_BROKER\_URL***;  **public** **static** **void** main(String[] args) {  ConnectionFactory connectionFactory;// 连接工厂  Connection connection = **null**;// 连接  Session session;// 会话 接受或者发送消息的线程  Destination destination;// 消息的目的地  MessageConsumer messageConsumer;// 消息的消费者  // 实例化连接工厂  connectionFactory = **new** ActiveMQConnectionFactory(ConsumerTest.***USERNAME***, ConsumerTest.***PASSWORD***,  ConsumerTest.***BROKEURL***);  **try** {  // 通过连接工厂获取连接  connection = connectionFactory.createConnection();  // 启动连接  connection.start();  // 创建session  session = connection.createSession(**false**, Session.***AUTO\_ACKNOWLEDGE***);  // 创建一个连接HelloWorld的消息队列  destination = session.createQueue("HelloWorld");  // 创建消息消费者  messageConsumer = session.createConsumer(destination);  **while** (**true**) {  TextMessage textMessage = (TextMessage) messageConsumer.receive(100000);  **if** (textMessage != **null**) {  System.***out***.println("收到的消息:" + textMessage.getText());  } **else** {  **break**;  }  }  } **catch** (JMSException e) {  e.printStackTrace();  }  }  } |

1. 结果展示

4.1生产者结果展示



ActiveMQ展示



4.2消费者结果展示

