**JMS :-**

1. **How To install**

Copy package from:-

<https://www.apache.org/dyn/closer.cgi?filename=activemq/activemq-artemis/2.32.0/apache-artemis-2.32.0-bin.zip&action=download>

Extract the folder

Cd {RootDirectory}/\apache-artemis-2.32.0-bin\apache-artemis-2.32.0\bin ./artemis create {DestinationFolder}/BrokerName

Cd {BrokerNameDirectory}/bin .\artemis.cmd run

1. **How To Code**

Create jndi.properties

java.naming.factory.initial=org.apache.activemq.artemis.jndi.ActiveMQInitialContextFactory

connectionFactory.ConnectionFactory=tcp://localhost:61616

queue.queue/myQueue=myQueue

topic.topic/myTopic=myTopic

queue.queue/requestQueue=requestQueue

queue.queue/replyQueue=replyQueue

queue.queue/expiryQueue=ExpiryQueue

**Dependency Requirement**

**<dependency>**

**<groupId>org.apache.activemq</groupId>**

**<artifactId>artemis-jms-client-all</artifactId>**

**<version>2.6.4</version>**

**</dependency>**

**Basic code for create firstQueue**

InitialContext initialContext = null;

Connection connection = null;

try {

initialContext = new InitialContext();

ConnectionFactory cf = (ConnectionFactory) initialContext.lookup("ConnectionFactory");

connection = cf.createConnection();

Session session = connection.createSession();

Queue queue = (Queue) initialContext.lookup("queue/myQueue");

MessageProducer producer = session.createProducer(queue);

TextMessage message = session.createTextMessage("I am the creator of my destiny");

producer.send(message);

System.out.println("Message Sent: " + message.getText());

MessageConsumer consumer = session.createConsumer(queue);

connection.start();

TextMessage messageReceived = (TextMessage) consumer.receive(5000);

System.out.println("Message Received: " + messageReceived.getText());

} catch (NamingException e) {

e.printStackTrace();

} catch (JMSException e) {

e.printStackTrace();

}finally {

if(initialContext!=null) {

try {

initialContext.close();

} catch (NamingException e) {

e.printStackTrace();

}

}

if(connection!=null) {

try {

connection.close();

} catch (JMSException e) {

e.printStackTrace();

}

}

}

}

ContextModified

public static void main(String[] args) throws NamingException {

InitialContext context = new InitialContext();

Queue queue = (Queue) context.lookup("queue/myQueue");

try(ActiveMQConnectionFactory cf = new ActiveMQConnectionFactory();

JMSContext jmsContext = cf.createContext()){

jmsContext.createProducer().send(queue,"Arise Awake and stop not till the goal is reached");

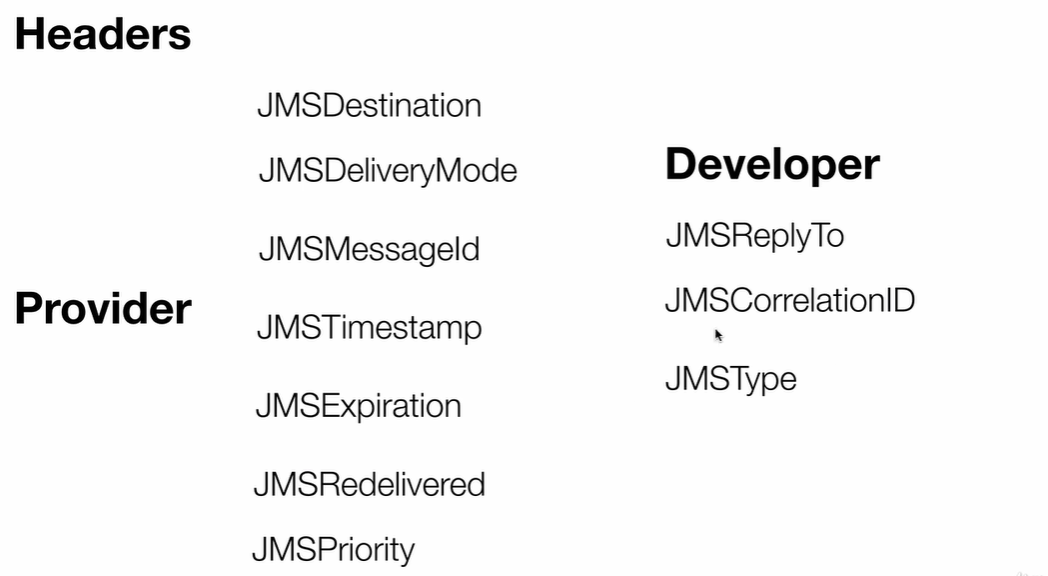
String messageReceived = jmsContext.createConsumer(queue).receiveBody(String.class);

System.out.println(messageReceived);

}

}

Basic of jms Overview





**JMS Set Priorities Methodology**

InitialContext context = new InitialContext();

Queue queue = (Queue) context.lookup("queue/myQueue");

try(ActiveMQConnectionFactory cf = new ActiveMQConnectionFactory();

JMSContext jmsContext = cf.createContext()){

JMSProducer producer = jmsContext.createProducer();

String[] messages = new String[3];

messages[0] = "Message One";

messages[1] = "Message Two";

messages[2] = "Message Three";

//producer.setPriority(3);

producer.send(queue, messages[0]);

//producer.setPriority(4);

producer.send(queue, messages[1]);

//producer.setPriority(9);

producer.send(queue, messages[2]);

JMSConsumer consumer = jmsContext.createConsumer(queue);

for(int i=0;i<3;i++) {

Message receivedMessage = consumer.receive();

System.out.println(receivedMessage.getJMSPriority());

//System.out.println(consumer.receiveBody(String.class));

}

} catch (JMSException e) {

e.printStackTrace();

}

**JMSReply Methodology**

**In this Logic when Lot of application work with each other , We come to know about we need to associate a particular reply to each other**

**CreatePropertiesInJMS**

**Type OF MessageSendAnd Receive**

**JMSFilter**

**Selector** Use Sql Like Query

**JMSAcknowledge**

**JMSTransaction**