Steps to Implement the JMS Application using NetBeans IDE with JDK 1.7 and above.

To develop JMS queue example, you need to install any application server. Here, we are using glassfish3 server where we are creating two JNDI.

- Create connection factory named myQueueConnectionFactory
- Create destination resource named myQueue

After creating JNDI, create server and receiver application. You need to run server and receiver in different console. Here, we are using Netbeans IDE, it is opened in different console by default.

Create connection factory and destination resource

- Open server admin console by the URL **http://localhost:4848**
- Login with the username and password.
- Click on the **JMS Resource -> Connection Factories -> New**, now write the pool name and select the Resource Type as QueueConnectionFactory then click on ok button.
- Click on the **JMS Resource -> Destination Resources -> New**, now write the pool name and select the Resource Type as Queue and give name “myQueue” then click on ok button.

Create the Sender and Receiver Application

1) In the IDE open the File Menu → New Project → select the Web Application → give the name JMSDemo.
2) Under the Source → create new package → “Demo”.
3) In side Demo Package Create the Java Class name”MyListner” (Responsible for the overriding the onMessage() method).
4) Copy and pest the code written in MyListner.java file attached at last.
5) Create new Servlet under the Demo Package→ name=”Sender”.
6) Copy and pest the code written in Sender.java file attached at last.

7) Create the second servlet under Demo package → name=”Receiver”

8) Copy and pest the code written in Receiver.java file attached at last.

9) Build the application and run the Sender servlet and open the glassfish console and message is printed on the console.

10) Then run the Receiver servlet and again open the glassfish console same message is printed on console again when receiver consume it from the Queue.

-

MySender.Java Class

```
import java.io.BufferedReader;
import java.io.InputStreamReader;
import javax.naming.*;
import javax.jms.*;


      try
      {   //Create and start connection
         InitialContext ctx=new InitialContext();
         QueueConnectionFactory
f=(QueueConnectionFactory)ctx.lookup("myQueueConnectionFactory");
         QueueConnection con=f.createQueueConnection();
         con.start();
         //2) create queue session
         QueueSession ses=con.createQueueSession(false,
Session.AUTO_ACKNOWLEDGE);
         //3) get the Queue object
         Queue t=(Queue)ctx.lookup("myQueue");
         //4)create QueueSender object
         QueueSender sender=ses.createSender(t);
         //5) create TextMessage object
         TextMessage msg=ses.createTextMessage();

         //6) write message

            msg.setText(“Hello World”);

            sender.send(msg);
            System.out.println("Message successfully sent.");
         }
         //8) connection close
         con.close();
     }catch(Exception e){System.out.println(e);}
```

MyReceiver.java Class

```
   import javax.jms.*;
   import javax.naming.InitialContext;

         try{
            //1) Create and start connection
            InitialContext ctx=new InitialContext();
            QueueConnectionFactory
f=(QueueConnectionFactory)ctx.lookup("myQueueConnectionFactory");
            QueueConnection con=f.createQueueConnection();
            con.start();
            //2) create Queue session
            QueueSession ses=con.createQueueSession(false,
Session.AUTO_ACKNOWLEDGE);
            //3) get the Queue object
            Queue t=(Queue)ctx.lookup("myQueue");
            //4)create QueueReceiver
            QueueReceiver receiver=ses.createReceiver(t);

            //5) create listener object
            MyListener listener=new MyListener();

            //6) register the listener object with receiver
            receiver.setMessageListener(listener);

            System.out.println("Receiver1 is ready, waiting for messages...");
            System.out.println("press Ctrl+c to shutdown...");

         }catch(Exception e){System.out.println(e);}
```

–---

MyListener.java

```
import javax.jms.*;
public class MyListener implements MessageListener {

    public void onMessage(Message m) {
        try{
        TextMessage msg=(TextMessage)m;

        System.out.println("following message is received:"+msg.getText());
        }catch(JMSException e){System.out.println(e);}
```