

Practical No.: 6B

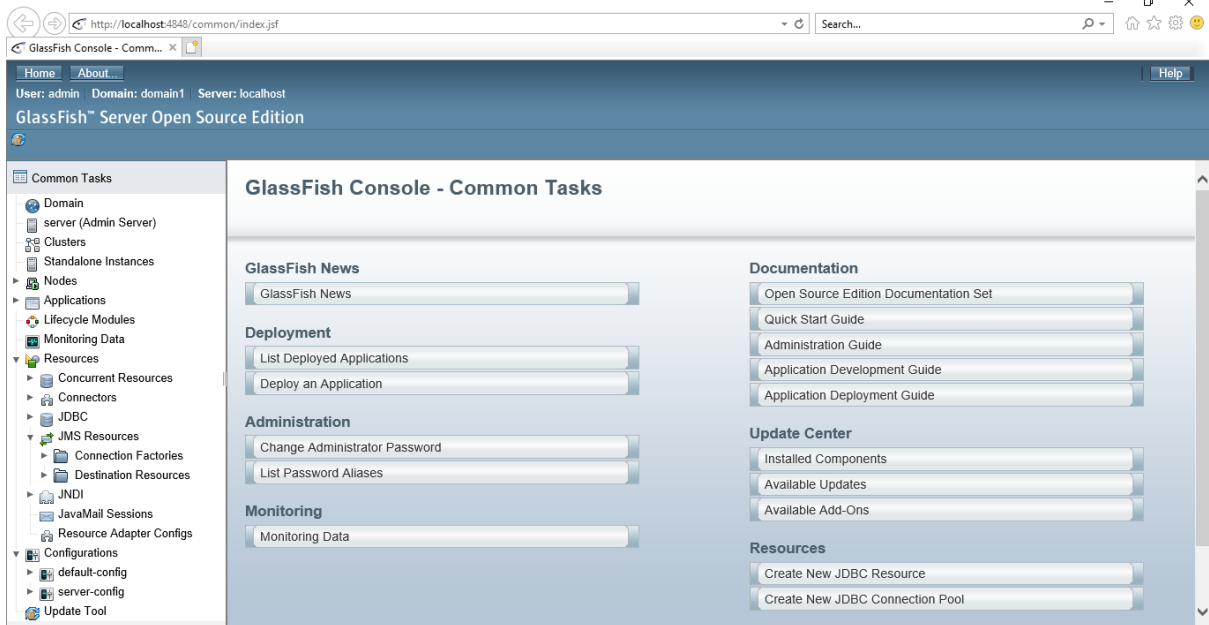
Aim: Write a Program on Message Bean Demonstration.

➔ *Start Glasfish Server under Services Tab*

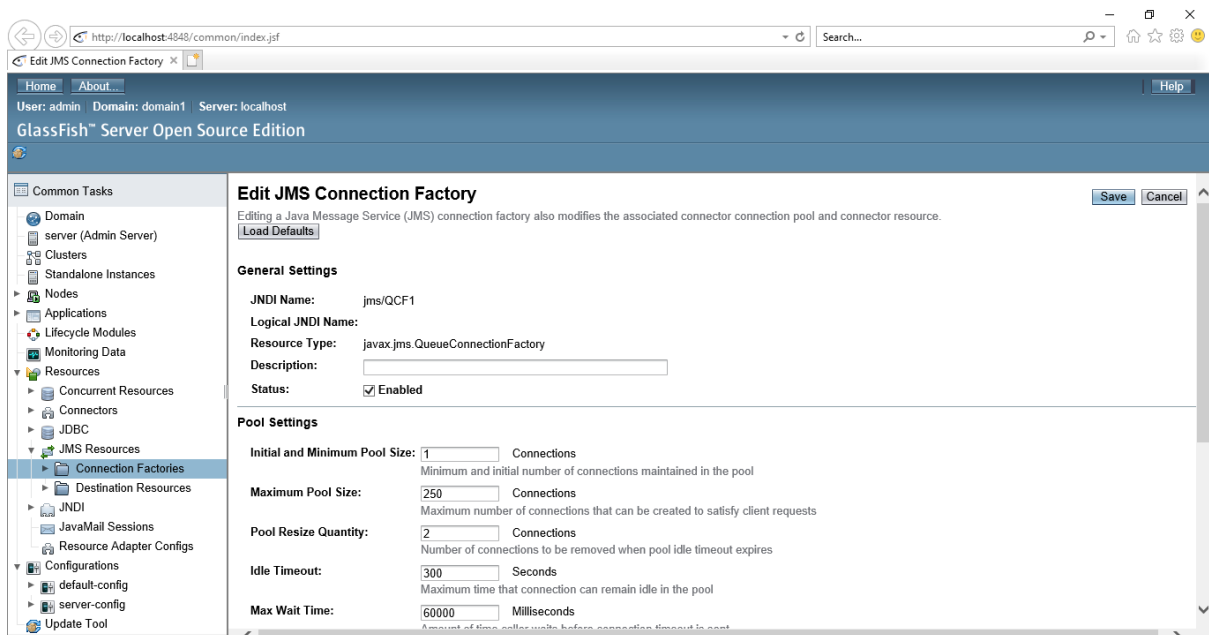
- *Right click on GlasFish Server 4.1.1*
- *Click on Start*

➔ *Creating Queue and Destination*

- *Right click on GlasFish Server 4.1.1*
- *Click on View Domain Admin Console*



Click on JMS Resources ➔ Connection Factories ➔ New



Click on OK

Click on JMS Resources → Destination Factories → New

Common Tasks

- Domain
 - server (Admin Server)
 - Clusters
 - Standalone Instances
 - Nodes
 - Applications
 - Lifecycle Modules
 - Monitoring Data
 - Resources
 - Concurrent Resources
 - Connectors
 - JDBC
 - JMS Resources
 - Connection Factories
 - Destination Resources**
 - JNDI
 - JavaMail Sessions
 - Resource Adapter Configs
 - Configurations
 - default-config
 - server-config
 - Update Tool

Edit JMS Destination Resource

Editing a Java Message Service (JMS) destination resource also modifies the associated admin object resource.

[Load Defaults](#)

JNDI Name: jms/QCF1Dest

Physical Destination Name: * QCF1Dest
Destination name in the Message Queue broker. If the destination does not exist, it will be created automatically when needed.

Resource Type: * javax.jms.Queue

Deployment Order: 100
Specifies the loading order of the resource at server startup. Lower numbers are loaded first.

Description:

Status: ☒ Enabled

Additional Properties (0)

[Add Property](#) [Delete Properties](#)

Select	Name	Value	Description
No items found.			

[Save](#) [Cancel](#)

Click on OK

Under Netbeans → New Project → Java EE → Enterprise Application → Project Name (Message Bean)

Right click on EJB Project Module (MessageBean-ejb) → Message Driven Bean → Finish

New Message-Driven Bean

Steps

- Choose File Type
- Name and Location**
- Activation Config Properties

Name and Location

EJB Name: NewMessageBean

Project: MessageBean-ejb

Location: Source Packages

Package: MessageBeans

☐ Project Destinations: [Add...](#)

☒ Server Destinations: jms/QCF1Dest

[< Back](#) [Next >](#) [Finish](#) [Cancel](#) [Help](#)

NewMessageBean.java

```
package MessageBeans;

import java.util.logging.Level;
import java.util.logging.Logger;
import javax.ejb.ActivationConfigProperty;
import javax.ejb.MessageDriven;
import javax.jms.JMSEException;
import javax.jms.Message;
import javax.jms.MessageListener;
import javax.jms.TextMessage;

/**
 *
 * @author TechnoBoy
 */
@MessageDriven(activationConfig = {
    @ActivationConfigProperty(propertyName = "destinationLookup", propertyValue =
"jms/QCF1Dest"),
    @ActivationConfigProperty(propertyName = "destinationType", propertyValue =
"javax.jms.Queue")
})
public class NewMessageBean implements MessageListener {

    public NewMessageBean() {
    }

    @Override
    public void onMessage(Message message) {
        try {
            TextMessage msg=(TextMessage) message;
            System.out.println("Welcome to Message Bean, "+msg.getText());
        } catch (JMSEException ex) {
            Logger.getLogger(NewMessageBean.class.getName()).log(Level.SEVERE, null, ex);
        }
    }
}
```

Right click on Web Project (MessageBean-war) → New → Servlet → Class Name: showMsg

New Servlet

Steps

1. Choose File Type
2. **Name and Location**
3. Configure Servlet Deployment

Name and Location

Class Name:

Project:

Location:

Package:

Created File:

< Back Next > Finish Cancel Help

showMsg.java

```
package MessageBeans;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.annotation.Resource;
import javax.inject.Inject;
import javax.jms.Connection;
import javax.jms.ConnectionFactory;
import javax.jms.JMSConnectionFactory;
import javax.jms.JMSContext;
import javax.jms.JMSException;
import javax.jms.MessageProducer;
import javax.jms.Queue;
import javax.jms.Session;
import javax.jms.TextMessage;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author TechnoBoy
 */
public class showMsg extends HttpServlet {

    @Resource(mappedName = "jms/QCF1Dest")
    private Queue qCF1Dest;

    @Resource(mappedName = "jms/QCF1")
    private ConnectionFactory queue;

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            /* TODO output your page here. You may use following sample code. */
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet showMsg</title>");
            out.println("</head>");
            out.println("<body>");
            String messageData=request.getParameter("name");
            sendJMSMessageToQCF1Dest(messageData);
            out.println("<h1>Name is Queued</h1>");
            out.println("</body>");
            out.println("</html>");
        }
    }
}
```

```

private void sendJMSMessageToQCF1Dest(String messageData) {
    try {
        Connection con=queue.createConnection();
        Session s = con.createSession();
        MessageProducer mp=s.createProducer(qCF1Dest);
        TextMessage tm=s.createTextMessage();
        tm.setText(messageData);
        mp.send(tm);

    } catch (JMSException ex) {
        Logger.getLogger(showMsg.class.getName()).log(Level.SEVERE, null, ex);
    }
}
}

```

MessageBean-war → New → JSP → File Name(*newjsp.jsp*)

newjsp.jsp

<%--

Document : newjsp

Created on : 20 Oct, 2019, 8:05:26 PM

Author : TechnoBoy

--%>

<% @page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Message Bean</title>

</head>

<body>

<form action="showMsg">

<input type="text" name="name">

<input type="submit" value="send">

</form>

</body>

</html>

Output:

Deploy MessageBean-ejb

Run MessageBean-war → newjsp.jsp

