

JMS and MDB Sample Lab



Contents

JMS AND LIBERTY PROFILE	3
1.1 CREATE AND CONFIGURE LIBERTY SERVER FOR JMS	5
1.2 CREATE NEW WEB PROJECT	7
1.3 CREATE JMSSAMPLEP2P SERVLET.....	8
1.4 CREATE JMSSAMPLEPUBSUB SERVLET.....	11
1.5 CREATE JMS SAMPLE MDB	12
1.6 RUNNING THE SAMPLE JMS APPLICATION.....	14
1.7 CLEAN UP.....	18
1.8 SUMMARY	18
APPENDIX A. NOTICES.....	19
APPENDIX B. TRADEMARKS AND COPYRIGHTS.....	21

JMS and Liberty Profile

Liberty profile supports asynchronous messaging as a method of communication that is based on the Java™ Message Service (JMS) programming interface. The JMS interface provides a common way for Java programs (clients and Java EE applications) to create, send, receive, and read asynchronous requests as JMS messages. With the Liberty profile, you can configure multiple JMS messaging providers, which can be used by the JMS applications.

The following JMS messaging providers are supported in the Liberty profile:

- 1) Liberty embedded messaging engine, as the JMS messaging provider
- 2) Service integration bus, which is the default messaging provider of WebSphere® Application Server
- 3) WebSphere MQ messaging provider, which uses the WebSphere MQ system as the provider

Your applications can use messaging resources from any of the three JMS providers. The choice of the provider is made based either on your messaging requirements or for integration with an existing messaging system. For example, you might want your applications to connect locally to the Liberty messaging provider without the need to configure any external messaging provider, in which case you would use the Liberty embedded messaging engine. Alternatively, you might want to integrate with your existing messaging infrastructure, such as WebSphere MQ. In such cases, you can either connect directly by using the WebSphere MQ messaging provider or configure a service integration bus with links to the WebSphere MQ network and then access the bus through the default messaging provider.

The scope of this lab is limited to **Liberty messaging**, an embedded messaging feature in the Liberty profile. It is a composable, flexible, and dynamic JMS messaging engine that runs within the Liberty profile. Liberty messaging is JMS 1.1 compliant and supports both the point-to-point and publish/subscribe messaging models.

In this lab exercise, you will learn:

- __a. Enabling JMS messaging for the Liberty Profile
- __b. Create a sample JMS P2P servlet
- __c. Create a sample JMS PubSub servlet
- __d. Create a sample JMS MDB
- __e. Run all the sample JMS applications created

As prerequisites, you should:

- __a. Complete Setup lab to set up the lab environment, and learn how to create a server using WebSphere Developer Tools.
- __b. **Optional:** Complete the Simple Development Lab if you need a refresher on how to use eclipse with WebSphere Developer Tools (WDT)

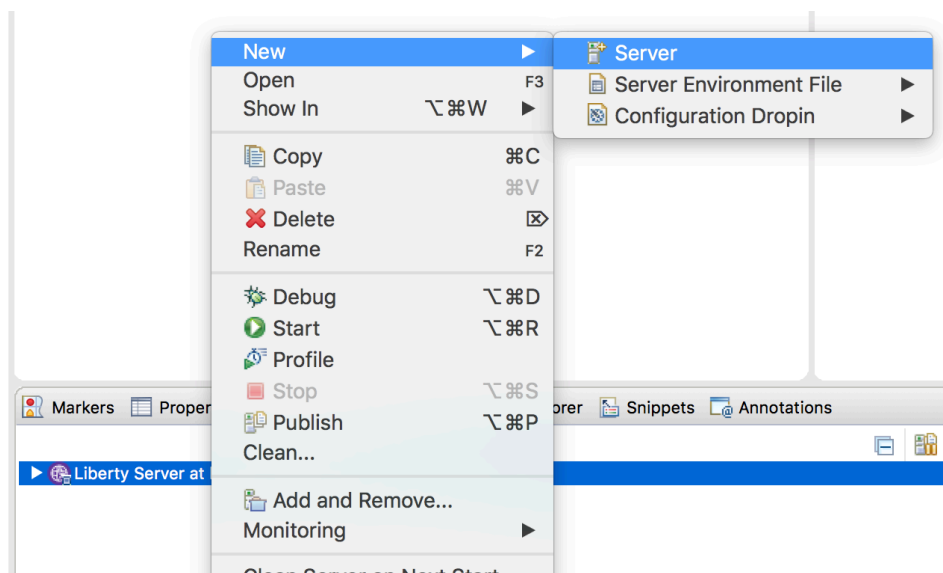
To run this lab, your workstation must meet the following requirements:

- Approximately 3 GB of memory free to run the developer workbench and the server
- Connectivity to the internet is NOT required
- Please refer to the following table for file and resource location references on different operating systems.

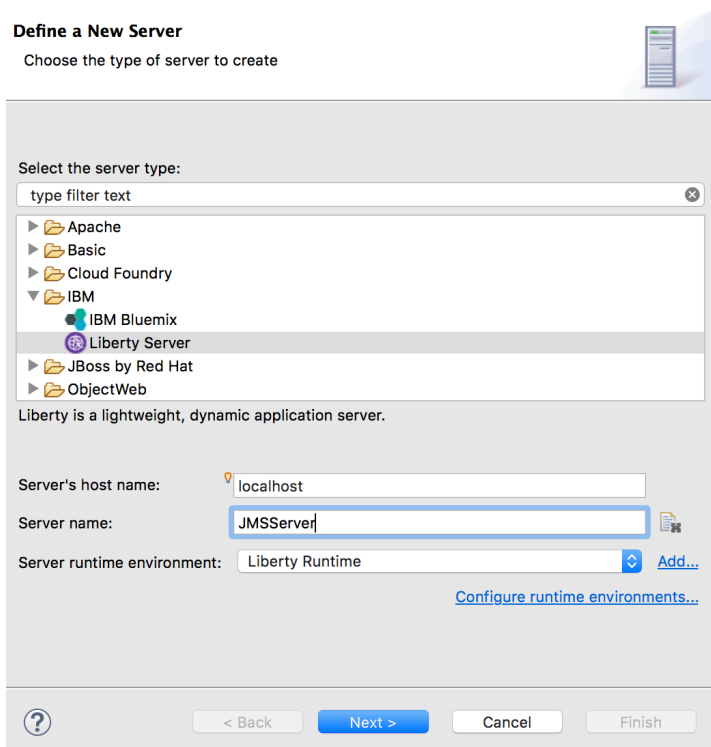
Location Ref.	OS	Absolute Path
{LAB_HOME}	Windows	C:\WLP_<version>
	Linux	~/WLP_<version>
	Mac OSX	

1.1 Create and Configure Liberty Server for JMS

- __1. Start **Eclipse** (eclipse.exe) from **{LAB_HOME}/eclipse** directory
- __2. Create a new Liberty Server called **JMSServer**.
 - __a. Right Click on the **Servers** tab and Select **New → Server**




- __b. Select **Liberty Runtime** and Click **Next**



- __c. Click **New** and enter the server name as **JMSServer**, click **Finish** then **Finish** again.

New Liberty Server
Specify the name of the new server.



User directory: Liberty Runtime

Server name: JMSServer

Template: defaultServer

? < Back Next > Cancel Finish

- __3. Expand the definition of the newly created **Liberty Server**. Open (double-click) the **server.xml** of the configuration and click on the **Source** tab.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <server description="new server">
3
4   <!-- Enable features -->
5   <featureManager>
6     <feature>jsp-2.3</feature>
7   </featureManager>
8
9   <!-- To access this server from a remote client add a host attribute to the following element, e.g. host="*" -->
10  <httpEndpoint id="defaultHttpEndpoint"
11               httpPort="9080"
12               httpsPort="9443" />
13
14  <!-- Automatically expand WAR files and EAR files -->
15  <applicationManager autoExpand="true"/>
16
17 </server>

```

Design Source

- __4. Replace the contents of **server.xml** with the lab content at **{LAB_HOME}/labs/development/3_JMS_<timestamp>/server.xml**
- __5. This **server.xml** contains the **JMS resource definitions** that you will need to run the sample application.
- __6. Review the changes in the **server.xml** that includes :
- __a. The features required for JMS, MDB, and built-in message engine
 - __b. Queue and topic configuration for the built-in message engine
 - __c. Activation Specification configuration for MDBs
 - __d. Configurations for queue and topic connection factories

- __e. Association of JMS queues and topics with built-in message engine queues and topics.

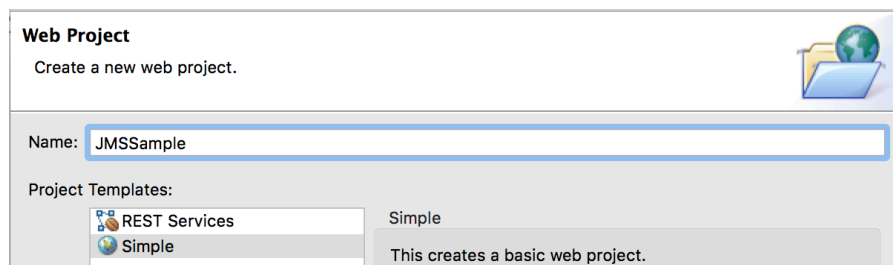
1.2 Create New Web Project

In this section we will create a new web project called **JMSSample** to deploy and run the JMS P2P and JMSPubSub servlets.

- __1. Create a **Web Project** called “**JMSSample**”

- __a. Select **File → New → Web Project**

- __b. Enter **JMSSample** as the name

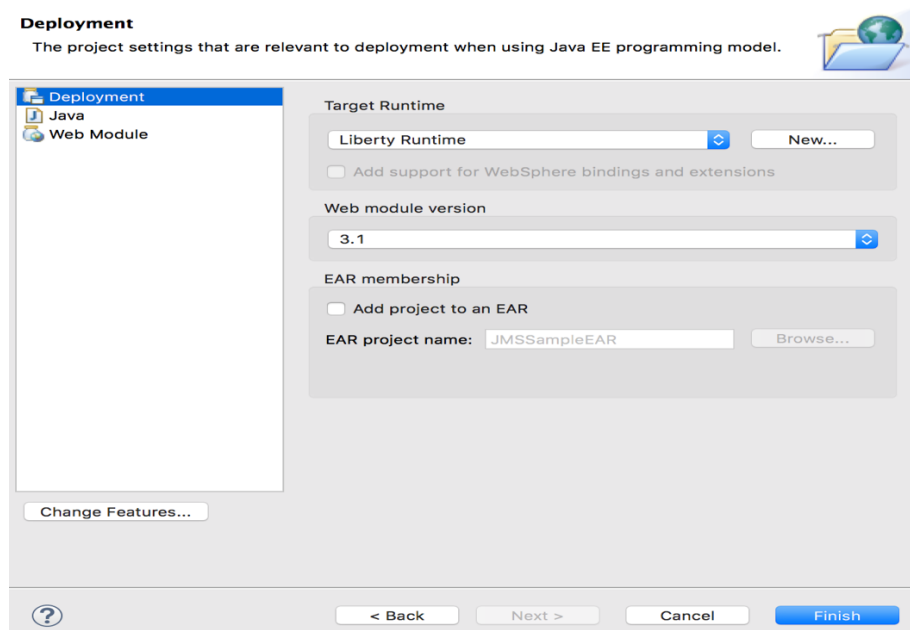


- __c. Click **Next**

- __d. Under **Target Runtime** select **WebSphere Application Server Liberty Profile**

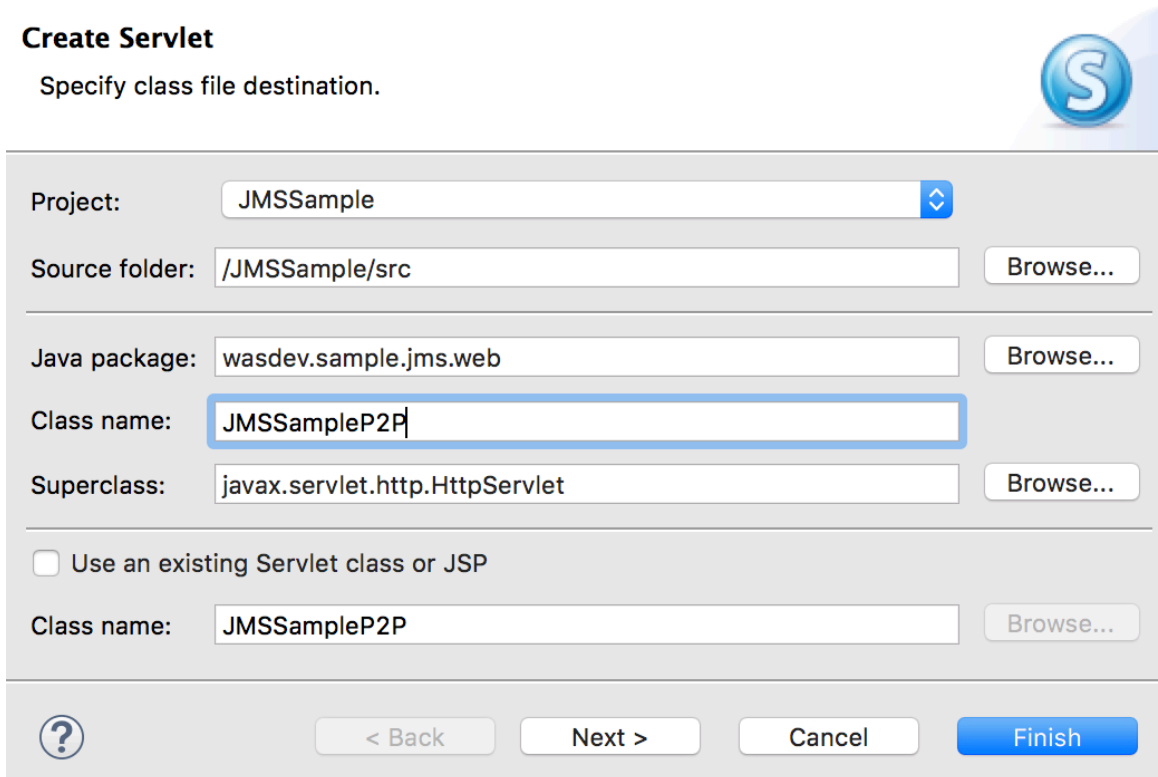
- __e. Deselect **Add project to EAR**

- __f. Click **Finish**



1.3 Create JMSSampleP2P Servlet

- ___1. From the **Enterprise Explorer**, Right click “**JMSSample**” project, and select **New → Servlet**
- ___2. Use **wasdev.sample.jms.web** as the package, and **JMSSampleP2P** as the class



Create Servlet
Specify class file destination.

Project: JMSSample

Source folder: /JMSSample/src Browse...

Java package: wasdev.sample.jms.web Browse...

Class name: JMSSampleP2P

Superclass: javax.servlet.http.HttpServlet Browse...

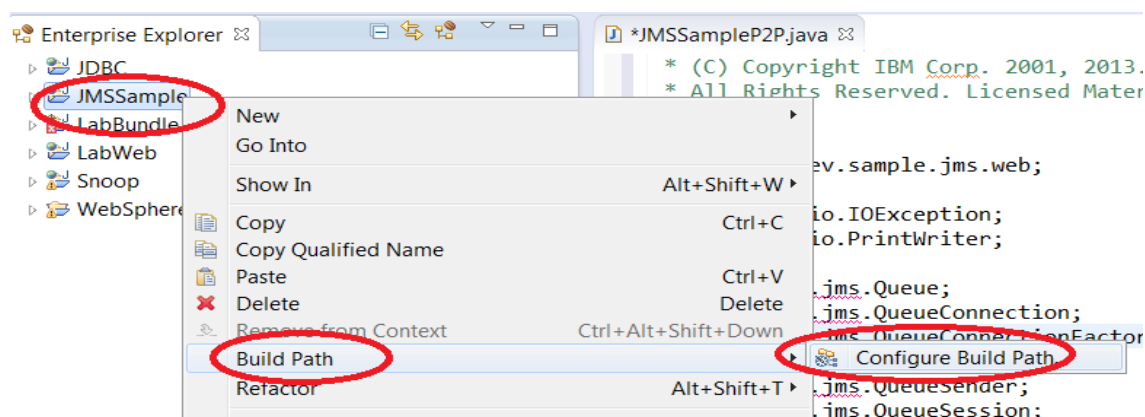
☐ Use an existing Servlet class or JSP

Class name: JMSSampleP2P Browse...

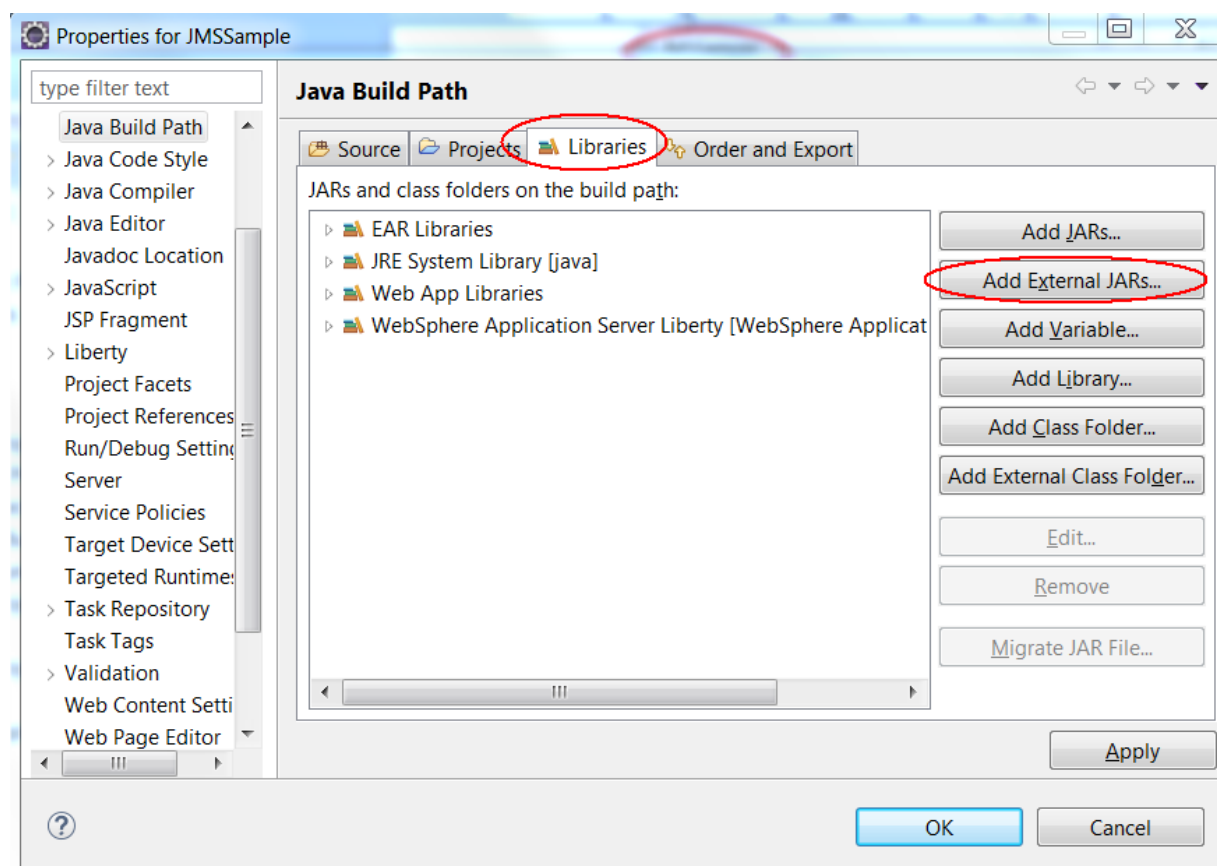
? < Back Next > Cancel Finish

- ___3. Click **Finish**
- ___4. Replace the contents of **JMSSampleP2P.java** with
{LAB_HOME}/labs/development/3_JMS_<timestamp>\JMSSampleP2P.java
- ___5. Review the servlet source code.
- ___6. Skip this step if you don't see compiler errors. If you see compiler errors, you need to configure the build path.

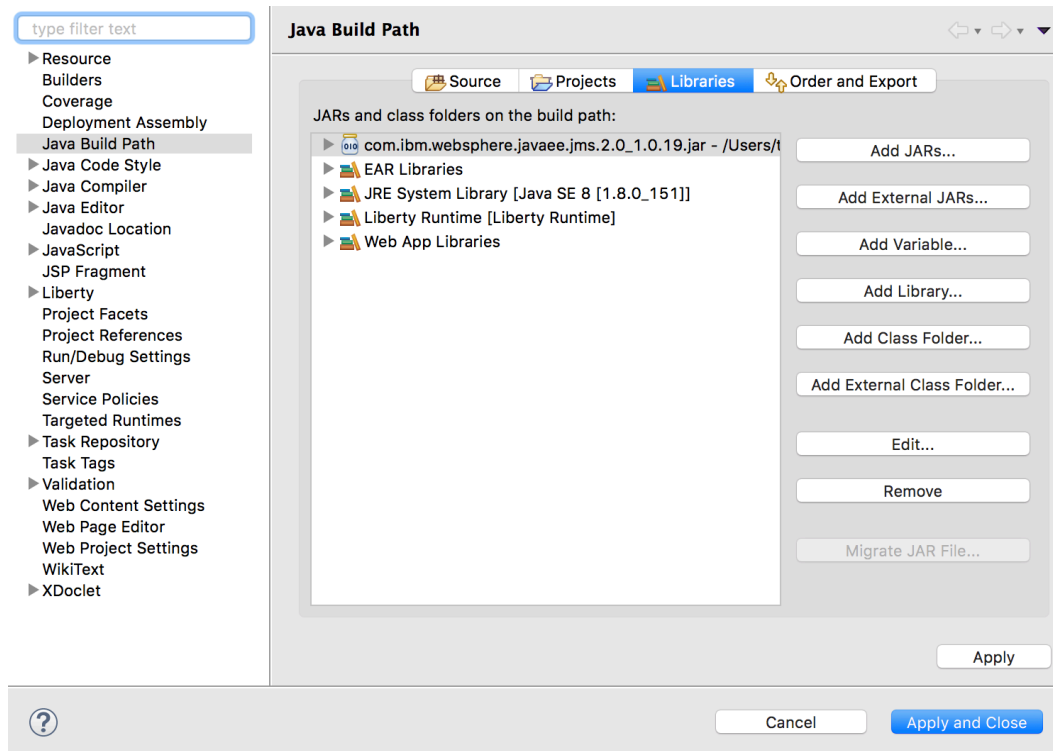
- __a. Right click on the **JMSSample** project then choose
Build Path → Configure Build Path



- __b. Select **Libraries** tab then click **Add External JARs**



- ___c. Navigate to `{LAB_HOME}\wlp\dev\api\spec` and select `com.ibm.ws.javaee.jms.2.0_1.0.19.jar`
- ___d. Click **Open**
- ___e. Verify the jar has been imported, then click **Apply and Close**



__7. The servlet accepts a request on

<http://<host>:<port>/JMSSample/JMSSampleP2P?action=<action>>

where the action is one of the following :

- __a. **sendAndReceive**: Sends a message defined by the queue **jndi_INPUT_Q**, then receives a message from it.
- __b. **sendMessage**: Sends a message to the queue **jndi_INPUT_Q**
- __c. **receiveAllMessageSelectors**: Sends a message to the queue **jndi_INPUT_Q**, then receives all messages on the queue.
- __d. **mdbRequestResponse**: Sends a message to the **MDBQ**, which will be received by the Sample MDB. The MDB sends a reply on **MDBREPLYQ**, which is then received by this method.

1.4 Create JMSSamplePubSub Servlet

- __1. From the **Enterprise Explorer**, Right click “**JMSSample**” project, and select **New → Servlet**
- __2. Use **wasdev.sample.jms.web** as the package, and **JMSSamplePubSub** as the class
- __3. Replace the contents of **JMSSamplePubSub.java** with the contents of **{LAB_HOME}/labs/development/3_JMS_<timestamp>JMSSamplePubSub.java**.
- __4. This servlet accepts a request on

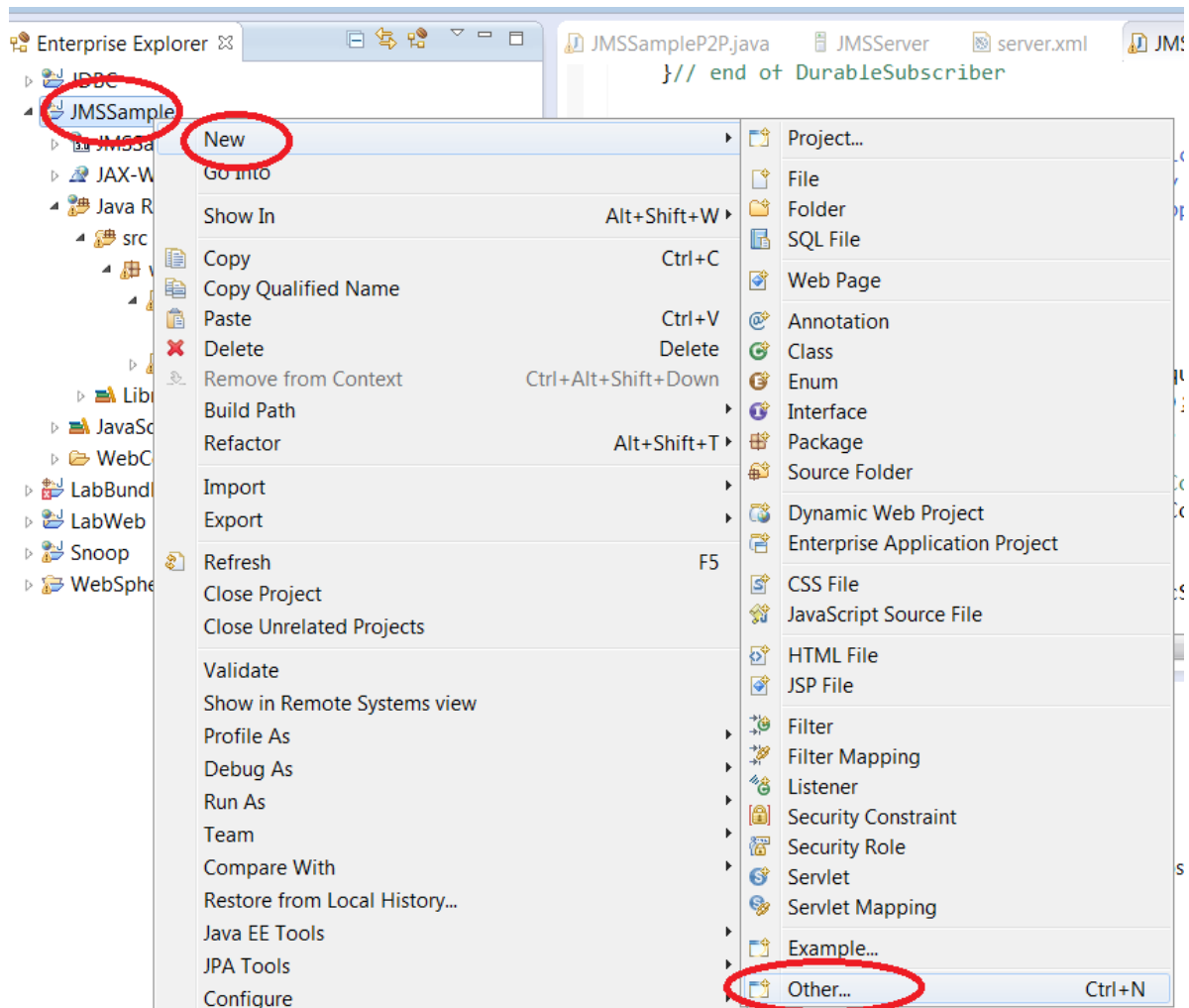
<http://<host>:<port>/JMSSample/JMSSamplePubSub?action=<action>>

where the action is one of the following:

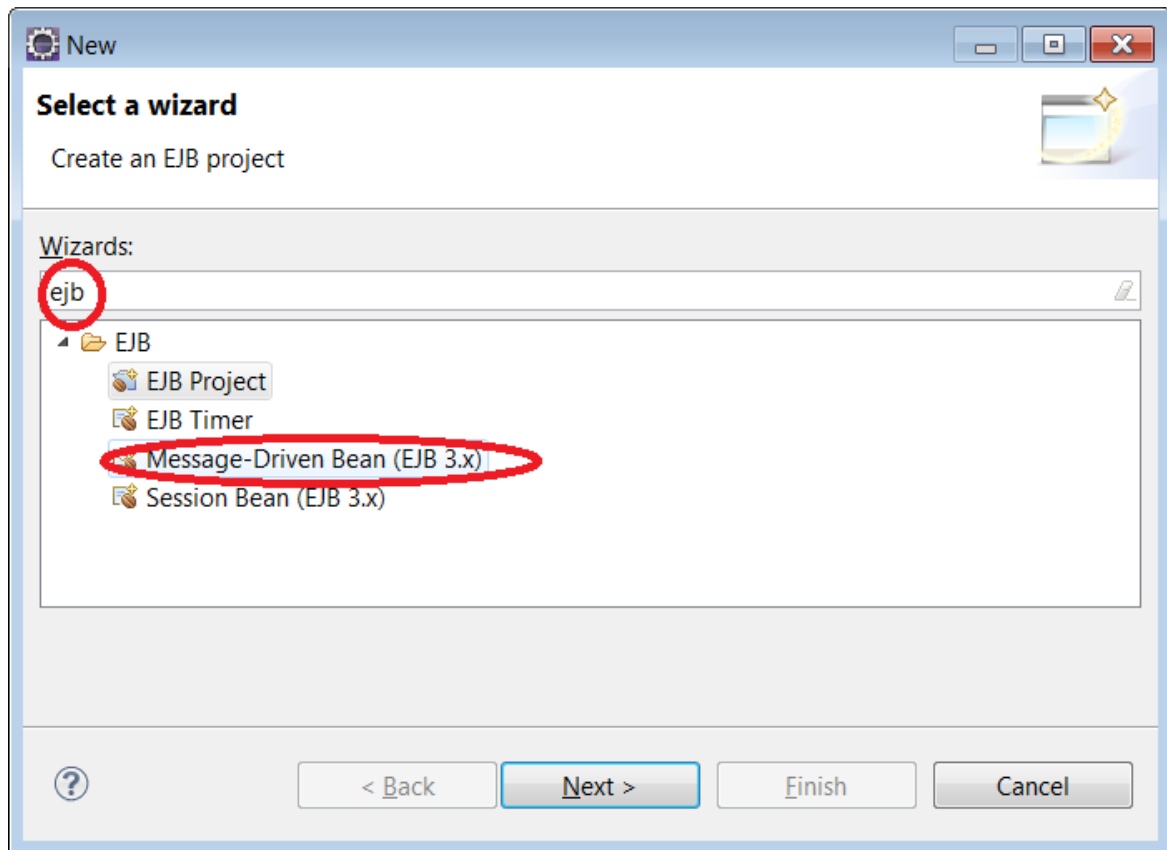
- __a. **nonDurableSubscriber**: Creates a non-durable subscriber for topic **jmsTopic**, sends a message, then receives message.
- __b. **durableSubscriber**: Creates a durable subscriber for topic **jmsTopic**, sends then receives message.
- __c. **publishMessages**: Publishes 5 messages to topic **jmsTopic**.
- __d. **unsubscribeDurableSubscriber**: Opens durable subscriber **DURATEST**, receives all messages, then un-unsubscribe.

1.5 Create JMS Sample MDB

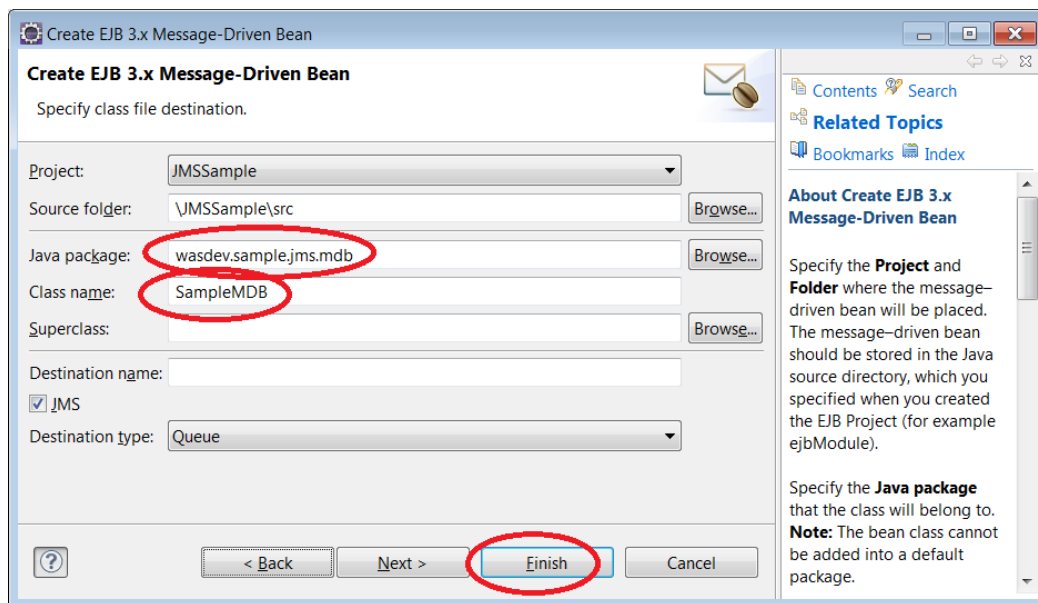
- ___1. From the **Enterprise Explorer**, Right click “**JMSSample**” project, and select **New → Other**



- ___2. Type **ejb** to filter, then choose **Message Driven Bean (EJB 3.x)**.



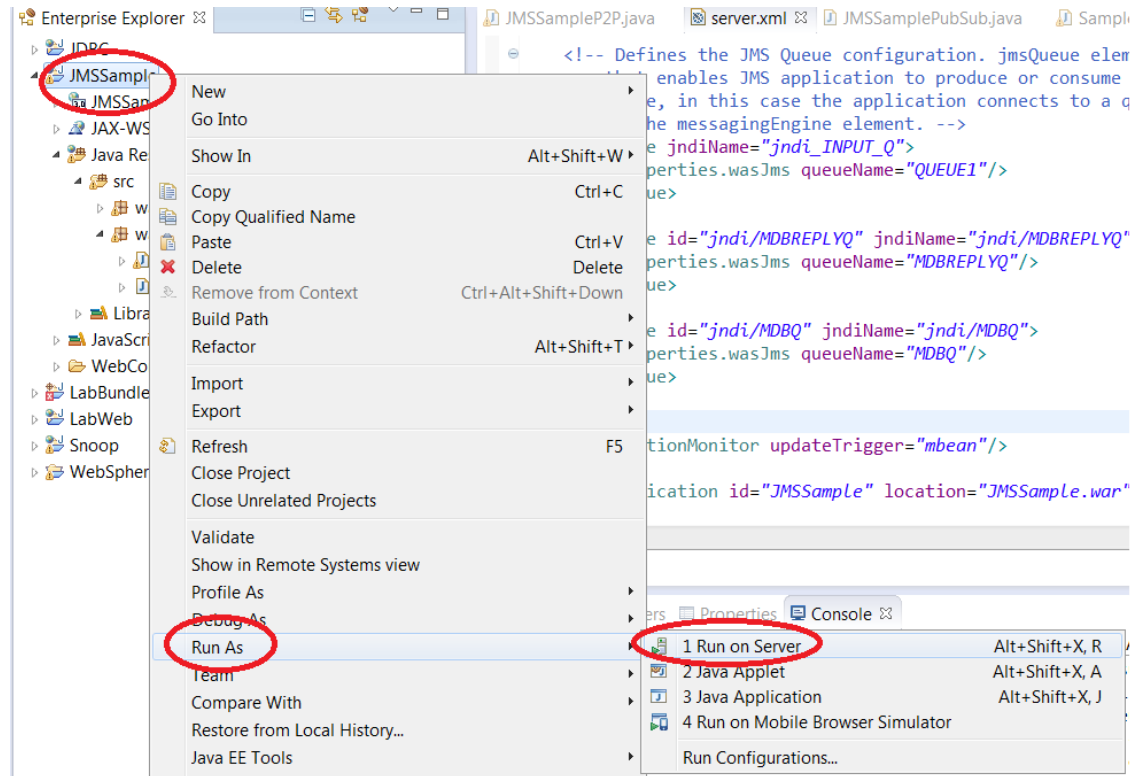
- __3. Click **Next**.
- __4. Enter **wasdev.sample.jms.mdb** for package name, and **SampleMDB** for class name.
- __5. Click **Finish**.



- ___6. Replace the contents of **SampleMDB.java** with the contents of
{LAB_HOME}/labs/development/3_JMS_<timestamp>\SampleMDB.java
- ___7. This MDB is to be configured to process messages on **MDBQ**, and send a reply via **MDBREPLYQ**.

1.6 Running the Sample JMS Application.

- ___1. Right click the **JMSSample** project, and choose **Run As → Run on Server**



__2. Choose the **JMSServer** (if you have multiple servers created), then click **Next**.

Run On Server

Select which server to use

How do you want to select the server?

☒ Choose an existing server

☐ Manually define a new server

Select the server that you want to use:

Server	State
Cloud	
localhost	
JMSServer	Stopped

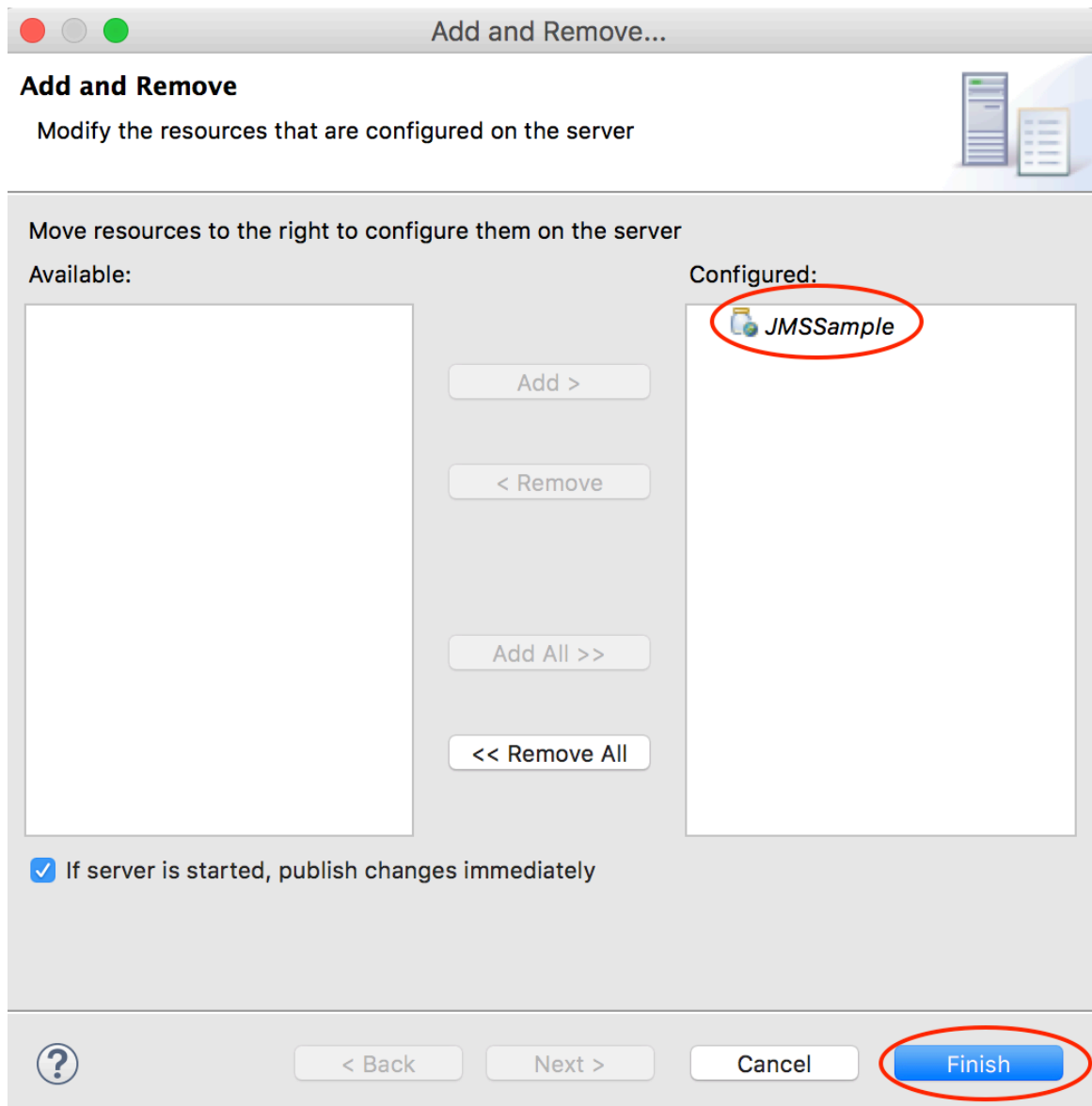
WebSphere Application Server Liberty is a lightweight, dynamic application server.

☐ Always use this server when running this project

[Columns...](#)

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

- ___3. If you have multiple projects, just select **JMSSample** as the project to run, and then click **Finish**



- ___4. Test point to point messaging by pointing your browser to the following URLs:
- ___a. <http://localhost:9124/JMSSample/JMSSampleP2P?ACTION=sendAndReceive>
 - ___b. <http://localhost:9124/JMSSample/JMSSampleP2P?ACTION=sendMessage>
 - ___c. <http://localhost:9124/JMSSample/JMSSampleP2P?ACTION=receiveAllMessages>
 - ___d. <http://localhost:9124/JMSSample/JMSSampleP2P?ACTION=receiveAllMessagesSelectors>
 - ___e. <http://localhost:9124/JMSSample/JMSSampleP2P?ACTION=mdbRequestResponse>



```

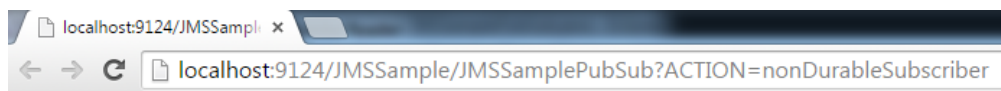
localhost:9124/JMSSample x localhost:9124/JMSSample x
localhost:9124/JMSSample/JMSSampleP2P?ACTION=sendAndReceive

SendAndReceive Started
Message sent successfully
Received Message Successfully :
  JMSMessage class: jms_text
  JMSType: null
  JMSDeliveryMode: 2
  JMSExpiration: 0
  JMSPriority: 4
  JMSMessageID: ID:4962e5ed0ec43d8cecebbb1a110a134f000000000000001
  JMSTimestamp: 1434966288649
  JMSCorrelationID: null
  JMSDestination: queue://QUEUE1
  JMSReplyTo: null
  JMSRedelivered: false
  JMSXDeliveryCount: 1
  JMS_IBM_System_MessageID: 26E3A403C30E9A2E_1
  JMSXUserID:
  JMSXAppID: WebSphere Embedded Messaging
Liberty Sample Message
SendAndReceive Completed

```

__5. Test pub sub by pointing your browser to:

- __a. <http://localhost:9124/JMSSample/JMSSamplePubSub?ACTION=nonDurableSubscriber>
- __b. <http://localhost:9124/JMSSample/JMSSamplePubSub?ACTION=durableSubscriber>
- __c. <http://localhost:9124/JMSSample/JMSSamplePubSub?ACTION=publishMessages>
- __d. <http://localhost:9124/JMSSample/JMSSamplePubSub?ACTION=unsubscribeDurableSubscriber>



```

localhost:9124/JMSSample x
localhost:9124/JMSSample/JMSSamplePubSub?ACTION=nonDurableSubscriber

NonDurableSubscriber Started
Received message for non-durable subscriber
  JMSMessage class: jms_text
  JMSType: null
  JMSDeliveryMode: 2
  JMSExpiration: 0
  JMSPriority: 4
  JMSMessageID: ID:e27efc5a331e40ca6f584b2b110a134f000000000000001
  JMSTimestamp: 1434966495663
  JMSCorrelationID: null
  JMSDestination: topic://SPORTS
  JMSReplyTo: null
  JMSRedelivered: false
  JMSXDeliveryCount: 1
  JMS_IBM_System_MessageID: 26E3A403C30E9A2E_2
  JMSXUserID:
  JMSXAppID: WebSphere Embedded Messaging
Liberty PubSub Message
NonDurableSubscriber Completed

```

1.7 Clean up

- ___1. Right click on the **JMSSample** application (on the server) and select **Remove** to remove the application.
- ___2. Right click on the **JMSServer** and select **Stop** to stop the server

1.8 Summary

In this lab you have learned:

- ✓ Enabling JMS messaging for the Liberty Profile
- ✓ Create a sample JMS P2P servlet
- ✓ Create a sample JMS PubSub servlet
- ✓ Create a sample JMS MDB
- ✓ Run all the sample JMS applications created

Appendix A. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have

been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental. All references to fictitious companies or individuals are used for illustration purposes only.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Appendix B. Trademarks and copyrights

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	AIX	CICS	ClearCase	ClearQuest	Cloudscape
Cube Views	DB2	developerWorks	DRDA	IMS	IMS/ESA
Informix	Lotus	Lotus Workflow	MQSeries	OmniFind	
Rational	Redbooks	Red Brick	RequisitePro	System i	
<i>System z</i>	<i>Tivoli</i>	<i>WebSphere</i>	<i>Workplace</i>	<i>System p</i>	

Adobe, Acrobat, Portable Document Format (PDF), and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. See Java Guidelines

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

ITIL is a registered trademark and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Other company, product and service names may be trademarks or service marks of others.

NOTES

NOTES



© Copyright IBM Corporation 2018.

The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. This information is based on current IBM product plans and strategy, which are subject to change by IBM without notice. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

IBM, the IBM logo and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at ibm.com/legal/copytrade.shtml

Other company, product and service names may be trademarks or service marks of others.



Please Recycle
