Page 1 of 22

Detailed example of running IBM MQ Explorer to setup JMS Administered objects in the JNDI .bindings file and test with sample JmsJndiProducer for queue and topic

https://www.ibm.com/support/pages/node/6956770

Date last updated: 26-Jun-2023

Angel Rivera IBM MQ Support <u>https://www.ibm.com/products/mq/support</u> Find all the support you need for IBM MQ

+++ Objective +++

Detailed example of running IBM MQ Explorer to setup JMS Administered objects in the JNDI .bindings file and test with the MQ JMS sample JmsJndiProducer for putting a message into a queue and for publishing a message to a topic.

Note 1: About using MQ Explorer

If you have installed MQ Explorer in the host where you want to have your JDNI .bindings file, then it is far easier to use MQ Explorer than the JMSAdmin tool. For more information on using the MQ Explorer for this task, see the online manual: <u>https://www.ibm.com/docs/en/ibm-mq/9.3?topic=explorer-creating-configuring-jms-administered-objects</u> IBM MQ / 9.3

Creating and configuring JMS administered objects

You can use IBM® MQ Explorer to configure the JMS administered objects that enable communication between Java™ applications and IBM MQ.

Note 2: For Jakarta Messaging 3.0

- You cannot administer JNDI using IBM MQ Explorer.

- JNDI administration is supported by the Jakarta Messaging 3.0 variant of JMSAdmin, which is **JMS30Admin**.

+ Related tutorial for JMS Admin:

If you want to use the command line utility JMSAdmin see:

https://www.ibm.com/support/pages/node/481027

Detailed example of running IBM MQ JMSAdmin to setup JMS Administered objects in the JNDI .bindings file and test with sample JmsJndiProducer for queue and topic

Page 2 of 22

++ Requirements ++

You need:

- The MQ Java fileset (MQSeriesJava)

- The MQ Samples fileset (MQSeriesSamples), which provides the sample JmsJndiProducer

- A Java Runtime Environment (JRE). You can use the JRE shipped with MQ.

- The MQ Explorer.

- Note: MQ Explorer 9.3 has been removed from the main product installers (Linux x86-64 and Windows), that is, is no longer included with the MQ server package from IBM Passport Advantage. But it is available as a separate download from IBM Fix Central: https://ibm.biz/mgexplorer

- The MQ samples for JMS are located at:

Linux:

/opt/mqm/samp/jms/samples/JmsJndiProducer.class

rpm fileset: MQSeriesSamples-9.2.0-5.x86_64

Windows:

C:\Program Files\IBM\MQ\tools\jms\samples

- It is necessary to have a Java Runtime Environment (JRE) installed in the machine. MQ already provides one, if you install the following components:

Linux:

\$ rpm -qa | grep MQSeriesJRE
Example output:
MQSeriesJRE-9.3.0-2.x86_64

Windows:

```
C:\> reg.exe query
"HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\IBM\WebSphere
MQ\Installation" /s
```

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\IBM\WebSphere
MQ\Installation\Installation1\Components
 JRE REG SZ Installed

++ Configuration

- Host-1: Queue Manager in Linux RHEL

MQ version.release: 9.3.0.2 LTS Host-1: volterra1 Queue Manager Name:QM93 Port: 1414 Server-Connection Channel: MY.CHANNEL

- Host-2L Client in Linux RHEL

MQ version.release: 9.2.0.6 LTS Host-2: suvetero1 User: mqm

- Host-3: Client in Windows 10

MQ version.release: 9.3.0.2 LTS Host-3: tolteca1 User: Administrator

+++ Procedure +++

++ Step 1: You must use "setmqenv" to setup the MQ environment variables.

Login as MQ Administrator.

It is very important that you always establish the proper setup for MQ via the command "setmqenv" because it will define the environment variables, updated the PATH, and update the CLASSPATH for Java/JMS programs.

In Linux and AIX you must "source" it! (That is, type a dot, then a space, then the setmqenv command and parameters).

The following use the default Installation directories.

Linux:

```
. /opt/mqm/bin/setmqenv -n Installation
AIX:
. /usr/mqm/bin/setmqenv -n Installation
```

Windows:

```
"C:\Program Files\IBM\MQ\bin\setmqenv" -n Installation1
```

Page 5 of 22

+ Highly recommended: Create a script that invokes setmqenv and adds samples directory

+ Linux

You can create a shell script and invoke it in your .bashrc file that issues the setmqenv command and adds the samples directories into the PATH and your current directory into the CLASSPATH, in that way, it will be easier to run the samples. Note that the "export PATH" is a very long single line!

```
# Name: set-mq-inst1
# Purpose: to setup the environment to run MQ in Installation1
. /opt/mqm/bin/setmqenv -n Installation1
# Additional MQ directories for the PATH
export
PATH=$PATH:$MQ_INSTALLATION_PATH/java/bin:$MQ_INSTALLATION_PATH/samp/
bin:$MQ_INSTALLATION_PATH/samp/jms/samples:
# Add local directory for running Java/JMS programs
export CLASSPATH=$CLASSPATH:.
# Display the full fix pack level
dspmqver -f 2
# end
```

+ Windows

You can create a batch command file and invoke it in your Windows terminal, that issues the setmqenv command and adds the samples directories into the PATH and your current directory into the CLASSPATH, in that way, it will be easier to run the samples. Note that the "SET PATH" is a very long single line!

```
REM Setup the environment to run MQ from Installation1
CALL "C:\Program Files\IBM\MQ\bin\setmqenv" -n Installation1
REM Adding Samples to the path
SET
PATH=%PATH%;%MQ_FILE_PATH%\Tools\c\Samples\Bin;%MQ_FILE_PATH%\Tools\c
\Samples\Bin64;%MQ_FILE_PATH%\Tools\jms\samples;%MQ_JAVA_INSTALL_PATH
%\bin\
dspmqver -f 2
SET CLASSPATH=%CLASSPATH%;.
```

++ Step 2: Creation of physical objects (for example, a Queue) in the queue manager

Note:

This document assumes that the queue manager is already created and that the proper security/authorizations are in place.

For more details, see the following tutorial:

https://www.ibm.com/support/pages/node/1135522

Configuring IBM MQ to use a dedicated Listener, Channel and Queue in Linux

The following is an example:

Host-1: volterra1.x.com Queue Manager name: QM93 Port: 1414

Ensure to have the following test queue:

```
$ runmqsc QM93
DEFINE QL(Q1)
DEFINE TOPIC(T1) TOPICSTR('TOPIC1')
END
```

++ Step 3: Create a directory where the JMS configuration objects will be located in a file name ".bindings".

Login as an MQ Administrator.

You can create the following subdirectory in the same directory where the other MQ objects are stored (variable: MQ_DATA_PATH) JNDI-Directory

- Linux: mkdir /var/mqm/JNDI-Directory

- Windows: mkdir %MQ_DATA_PATH%\JNDI-Directory

+ File permissions and ownership

The permissions and ownership must allow read-write permission to the user "mqm" and read-execute for group and others:

\$ ls -dl /var/mqm/JNDI-Directory drwxr-xr-x 2 mqm mqm 91 Jun 26 05:52 /var/mqm/JNDI-Directory

\$ ls -al -rw-r--r-- 1 mqm mqm 15880 Jun 23 05:56 .bindings

If the file permission / ownership are not correct, then at runtime, JMSAdmin may issue the following error, which is very vague and generic.

Unable to bind object

Page 8 of 22

++ Step 4: Launch MQ Explorer to add JMS Administered objects

- Launch MQ Explorer
- On the left navigation panel, scroll down to the bottom.
- Select:

JMS Administered Objects

QM931
 QMMI1 on 'suvereto1.fyre.ibm.com(142
 Queue Manager Clusters
 JMS Administered Objects
 Managed File Transfer
 Service Definition Repositories
 Right click and select:

Add Initial Context...

QMMI1 on 'suvereto1.fyre.ibm.com(142			
궏 Queue Manager Clusters	Right Click		
JMS Administered Objects			
😂 Managed File Transfer	Add Initial Context		
Service Definition Reposito	ries		

- You will see a dialog that says "Connection details" Let's look at the top part about:

Where is the JNDI namespace located and the JNDI Service Provider

Connection details

Bindings directory does not contain an existing JNDI namespace and contains other files and/or directories.

JMS administered objects are stored in Java Naming and Directory Interface (JNDI) namespaces. An Initial context defines the root of a JNDI namespace and is used to access the JMS objects that are stored in the namespace.

Where is the JNDI namespace located?

○ LDAP server		
• File system		
○ Other		
- INDI Sanica Providar		
JNDI Service Frovider		
Factory class:	com.sun.jndi.fscontext.RefFSContextFactory	
Factory class:	com.sun.jnui.iscontext.netrsContextractory	

Page 9 of 22

This top part is the equivalent of the following attribute in the JMSAdmin.config file: INITIAL_CONTEXT_FACTORY=com.sun.jndi.fscontext.RefFSContextFactory

This means that the JNDI will be a FILE, named: .bindings

The bottom part specifies the full path of the directory where the .bindings file will be located.

JNDI Namespace Location	n	
Bindings directory:	C:\ProgramData\IBM\MQ\JNDI-Directory	Browse
Provider URL:	RL: file:/C:/ProgramData/IBM/MQ/JNDI-Directory/	

For this tutorial is: C:\ProgramData\IBM\MQ\JNDI-Directory

This is the equivalent following attribute in the JMSAdmin.config file: PROVIDER_URL

- For Windows:

Notice only 1 forward slash after "file:" and notice the drive letter "C:/" and that the slash is FORWARD SLASH:

For example:

file:/C:/ProgramData/IBM/MQ/JNDI-Directory/

 For UNIX: Notice the 3 forward slashes after "file:" For example: file:///var/mgm/JNDI-Directory

Follow the prompts and click on "Finish"

Page 10 of 22

The MQ Explorer will show the entry for the new "Initial Context"

🗟 MQ Explorer - Navigator 🛛 👘 🖻 🔋 🗖	Image: MQ Explorer - Content ≅		배 🏟 🔛 🖇 🗖 🗖
✓ ֎ IBM MQ	IMS Administered Objects		
👻 🗁 Queue Managers			
FR1 on 'riggioni1.fyre.ibm.com(1421)'	 Name 	Status	Provider URL
FR2 on 'suvereto1.fyre.ibm.com(1422)'	"₣ file:/C:/ProgramData/IBM/MQ/JNDI-Directory/	Connected	file:/C:/ProgramData/IBM/MQ/JNDI
PR3 on 'riggioni1.fyre.ibm.com(1423)'			
PR4 on 'suvereto1.fyre.ibm.com(1424)'			
PR5 on 'suvereto1.fyre.ibm.com(1425)'			
🖗 QM931			
QMMI1 on 'suvereto1.fyre.ibm.com(1420)'			
🖻 Oueue Manager Clusters			
🗸 🗁 JMS Administered Objects			
> 🖷 file:/C:/ProgramData/IBM/MQ/JNDI-Directory/			
🗁 Managed File Transfer			
🗁 Service Definition Repositories			

In the future, you may need to "manually connect" to this Initial Context.



Expand the folders for the Initial Context



You will see 2 folders and at this time they are empty: Connection Factories Destinations

Х

Let's add our first Connection Factory.

Select the folder "Connection Factories", right click on "New" and then click on "Connection Factory..."

 JMS Administered Objects Image: file:/C:/ProgramData/IBM/MQ/JNDI-Directory/ 	
Connection Fa Destinations New	Connection Factory
⇔ Managed File Transf∢ Tests >	

In the next dialog, enter the name:

CF1

New Connection Factory

Create a Connection Factory

Enter the details of the connection factory

Name:		
CF1		
Messaging provider:		
IBM MQ		\sim
Use IBM MQ as the me the IBM MQ Publish/Su	essaging provider if the JMS client application uses point-to-point ubscribe engine.	messaging or

Page 12 of 22

Accept the default that this will be a "Connection Factory" (JMS 1.1 standard)

Note: The very old JMS 1.0 standard had "Queue Connection Factory" and "Topic Connection Factory".

Select the type of connection factory
Name:
CF1
Туре:
Connection Factory
Support XA transactions
This creates an object of type 'com.ibm.mq.jms.MQConnectionFactory'. Select this option if the JMS client application uses both point-to-point messaging and publish/subscribe messaging.
Just click "Next"

In the next screen you need to select for the "Transport" type: MQ Client

Name:	
CF1	
Transport:	
Bindings	
Bindings	
MQ Client	

Click Next. Because we do not have another Connection Factory so far, click Next.

In the next page you will see a notebook with several tabs.

The first tab is "General".

New Connection Factory

Change properties

Change the properties of the new Connection Factory

General	General		
Reconnection	Name:	CF1	
Channels SSI	Description:		
Exits	Class name:	MQConnectionFactory	
Broker	Messaging provider:	IBM MQ	
lemporary queues Temporary topics	Transport:	Client	
Subscriber	Provider version: *	• unspecified	
Extended Advanced tuning		0	
5	Client identifier:		

Ensure that you have the name (CF1) and transport type (CLIENT)

Select the tab: Connection

General	Connection		
Reconnection	Base queue manager:	QM93	
Channels	Broker queue manager:		
Exits	Connection list:	volterra1.fyre.ibm.com(1414)	

Specify the queue manager (QM93) and the connection list [volterra1.fyre.ibm.com(1414)]

Page 14 of 22

Select the tab: Channels

General	Channels			
Connection Reconnection	Channel:	MY.CHANNEL		
Channels SSL	Client channel definition table URL:			

Specify the server-connection channel (MY.CHANNEL)

All these selections are the equivalent of the following command for JMSAdmin:

```
DEF CF(CF1) QMGR(QM93) CHANNEL(MY.CHANNEL) TRANSPORT(CLIENT)
HOSTNAME(volterral.fyre.ibm.com) PORT(1414)
```

Click on "Finish"

You will see now our 1st entry in the right panel of the Connection Factories:

✓ ֎ IBM MQ	Connecti	on Factori	es		
🕆 🗁 Queue Managers		Filter: Standard for JMS Connection Factory			
FR1 on 'riggioni1.fyre.ibm.com(1421)'	Filter: Stand				
FR2 on 'suvereto1.fyre.ibm.com(1422)'	Anne Description Class name Messaging provider			Transport	
PR3 on 'riggioni1.fyre.ibm.com(1423)'	li li GF1		MQConnectionFactory	IBM MQ	Client
PR4 on 'suvereto1.fyre.ibm.com(1424)'			,		
PR5 on 'suvereto1.fyre.ibm.com(1425)'					
₽ QM931					
QMMI1 on 'suvereto1.fyre.ibm.com(1420)'					
눧 Queue Manager Clusters					
 > JMS Administered Objects 					
file:/C:/ProgramData/IBM/MQ/JNDI-Directory/					
Connection Factories					
🗁 Destinations					
🗁 Managed File Transfer					

Let's proceed to define our Destination Queue Q1.

Select the folder "Destinations", right click "New" and select "Destination..."



In the next dialog, enter the name: Q1 ... and specify that the type is: Queue

Create a Destination

Enter the details of the object you wish to create

Name:	
Q1	
Messaging provider:	
IBM MQ and Real-time	~
A destination that is created in MQ Explorer can be used with both IBM MQ and Real-time messaging providers.	
\frown	
Туре:	
Queue	~
Select this option if the JMS application uses point-to-point messaging. The destination will represent a queue.	

Click Next.

You will see a notebook with Tabs. We are only interested in the tab: General

General	
Name:	Q1
Description:	
Class name:	MQQueue
Messaging provider:	IBM MQ and Real-time
Queue manager:	QM93
Queue: *	Q1
	General Name: Description: Class name: Messaging provider: Queue manager: Queue: *

You need to enter the queue manager name (QM93) and the name of the queue (Q1).

These options are the equivalent of the following JMSAdmin command:

DEF Q(Q1) QMGR(QM93) QUEUE(Q1)

Page 16 of 22

You will see now our 1st entry (queue Q1) in the Destinations section, on the right panel.



Let's proceed to create the topic.

On the left panel, select "Destinations", right click "New", then "Destination".

Enter the name T1 and ensure to specify the type to be "Topic"



You will see a notebook dialog with several tabs. We are only interested in the tab: General

General	General	
Broker	Name:	Т1
Producers Consumers	Description:	
Extended	Class name:	MQTopic
	Messaging provider:	IBM MQ and Real-time
	Topic: *	TOPIC1

Ensure that we have the name "T1" and the topic "TOPIC1"

These options are the equivalent of the following JMSAdmin command:

```
DEF T(T1) TOPIC('TOPIC1')
```

You will see now our 2nd entry (topic T1) in the Destinations section, on the right panel.

✓ ֎ IBM MQ	Destinations								
 Queue Managers FR1 on 'riggioni1.fyre.ibm.com(1421)' 	Filter: Standard for JMS Destination								
 FR2 on 'suvereto1.fyre.ibm.com(1422)' PR3 on 'riggioni1.fyre.ibm.com(1423)' 	✓ Name ¹ Q1	Description	Class name MQQueue	Messaging provider IBM MQ and Real-time	Queue manager QM93	Queue Q1	Торіс		
 PR4 on 'suvereto1.fyre.ibm.com(1424)' PR5 on 'suvereto1.fyre.ibm.com(1425)' QM931 	ার T1		MQTopic	IBM MQ and Real-time			TOPIC1		
□ QMMI1 on 'suvereto1.fyre.ibm.com(1420)' Queue Manager Clusters → ▷ JMS Administered Obiects									
 * file:/C:/ProgramData/IBM/MQ/JNDI-Directory/ Connection Factories Destinations Managed File Transfer 									

You can see the .bindings file in the desired directory:

JNDI-Directory													_	
🕂 New 🗸	/ D		Ĩ		¢	Û	↑↓ Sort	~	≡ Viev	N ~				
$\leftarrow \rightarrow $ ~ \uparrow	•	C:\Prog	ramData	a\IBM\M(Q\JNDI-D	irectory]			~	С	0	Sear	ch JN[
🔶 Quick access		Nam	e		^			Dat	e modifie	d		Туре		
🛄 Desktop	*	. 🗋	bindings					2/1	8/2023 3:	43 PM		BIND	NGS F	ile
🚽 Downloads	*													

++ Step 5: Copy the .bindings file to other hosts (if needed)

Because the creation of the .bindings file is far easier via MQ Explorer, then possibly you could create this file in a Windows host and then copy the .bindings file into another host, such as AIX or Linu.

You will need to store the file in the desired directory in the other host, such as in: /var/mqm/JNDI-Directory

++ Step 6: Run the MQ sample: JmsJndiProducer

The MQ sample JmsJndiProducer can be used to:

- Access an JNDI to look up for the Connection Factory and a Destination (queue or topic)
- Put a message into a queue.
- Or publish a message into a topic.

In Linux, the sample is in: /opt/mqm/samp/jms/samples/JmsJndiProducer.java

In Windows, the sample is in: C:\Program Files\IBM\MQ\tools\jms\samples\JmsJndiProducer.java

+ What to enter into the -I flag (Initial Context, aka JNDI file .bindings)?

The following is a bit tricky. You need to specify exactly the value of the URI from the MQ Explorer:



In Windows: file:/C:/ProgramData/IBM/MQ/JNDI-Directory

In Linux: file:///var/mqm/JNDI-Directory

```
Page 19 of 22
```

```
+ Putting a message into Destination Queue Q1 (from Linux)
```

We are using only Linux to show this example. But the usage from Windows is the same, of course, the flag -i will be different.

Note:

If your only "java" executable is the one provided by MQ, then its full path name is: /opt/mqm/java/jre64/jre/bin/java

```
mqm@suvereto1.fyre.ibm.com: /opt/mqm/samp/jms/samples
```

```
$ cd /opt/mqm/samp/jms/samples/
$ java JmsJndiProducer -i file:///var/mqm/JNDI-Directory -c CF1 -d Q1
Initial context found!
Sent message:
  JMSMessage class: jms text
            null
  JMSType:
  JMSDeliveryMode:
                   2
  JMSDeliveryDelay: 0
  JMSDeliveryTime: 1676754470495
  JMSExpiration:
                  0
 JMSPriority:
                   4
 JMSMessageID:
ID:414d5120514d39332020202020202020eabeef63013f0040
 JMSTimestamp: 1676754470495
 JMSCorrelationID: null
  JMSDestination: queue://QM93/Q1
 JMSReplyTo:
                  null
  JMSRedelivered: false
   JMSXAppID: JmsJndiProducer
   JMSXDeliveryCount: 0
   JMSXUserID: mqm
   JMS IBM PutApplType: 28
    JMS IBM PutDate: 20230218
    JMS IBM PutTime: 21075050
JmsJndiProducer: Your lucky number today is 467
SUCCESS
```

Notice the output line that indicates that the destination is a queue, and provides the name of the queue manager and the queue: JMSDestination: queue://QM93/Q1 ++ Browsing the message.

000000B0: 48

+ Using the MQ sample "amqsbcg" Notice the RHF2 (RFH) header, shown in blue.

```
Example:
amqsbcq Q1 QM93
AMQSBCG0 - starts here
MQOPEN - 'Q1'
MQGET of message number 1, CompCode:0 Reason:0
****Message descriptor****
 StrucId : 'MD ' Version : 2
 Report : 0 MsgType : 8
 Expiry : -1 Feedback : 0
 Encoding: 273 CodedCharSetId: 1208
 Format : 'MQHRF2
 Priority: 4 Persistence: 1
 MsqId : X'414D5120514D393357494E202020202D0E7664012B0040'
 BackoutCount : 0
             : '
 ReplyToQ
 ReplyToQMgr : 'QM93
 ** Identity Context
 UserIdentifier : '594079897
 AccountingToken :
  X'16010501000000480366CD527F8B47B07254FAC264847300000000000000000C'
 ApplIdentityData : '
 ** Origin Context
 PutApplType : '11'
 PutApplName
             : 'JmsProducer
 PutDate : '20230530' PutTime : '14562385'
 ApplOriginData : '
 MsgSegNumber : '1'
 Offset : '0'
              : '0'
 MsqFlaqs
 OriginalLength : '-1'
* * * *
                * * * *
      Message
length - 177 of 177 bytes
00000000: 5246 4820 0000 0002 0000 0094 0000 0111 'RFH .....ö....'
00000010: 0000 04B8 4D51 5354 5220 2020 0000 0000 '....MQSTR ....'
00000020: 0000 04B8 0000 0020 3C6D 6364 3E3C 4D73 '..... <mcd><mcd><ms'
00000030: 643E 6A6D 735F 7465 7874 3C2F 4D73 643E 'd>jms text</msd>'
00000040: 3C2F 6D63 643E 2020 0000 0048 3C6A 6D73 '</mcd>
00000050: 3E3C 4473 743E 7175 6575 653A 2F2F 2F51 '><Dst>queue:///Q'
00000060: 313C 2F44 7374 3E3C 546D 733E 3136 3835 '1</Dst><Tms>1685'
00000070: 3435 3835 3833 3834 353C 2F54 6D73 3E3C '458583845</Tms><'
00000080: 446C 763E 323C 2F44 6C76 3E3C 2F6A 6D73 'Dlv>2</Dlv></jms'
00000090: 3E20 2020 5075 7420 5465 7374 204A 4D53 '> Put Test JMS'
000000A0: 204D 6573 7361 6765 2077 6974 6820 5246 ' Message with RF'
```

'Η

+ Using the MQ Explorer

Select the desired queue, then right click to show the context menu. Then select: Browse Messages...

ANGEL.MODEL	Model
⊠ Q1	Compare with
^I √Q1A	
⊠ Q2	Delete
⊠ Q3	Status
⊠Q31	
⊠ Q4	Watch Activity
⊠ Q5	Clear Messages
de_QM93LNX	Put Test Message
AM93LNX	Browse Messages
QNONP	brottse messuges

You will see the messages in the queue. Double click on the desired message.



To see the JMS header, select the tab: Named Properties



```
Page 22 of 22
```

```
++ Publishing a message into Destination Topic T1 (from Windows)
```

We are using only Windows to show this example. But the usage from Linux is the same, of course, the flag -i will be different.

Note:

If your only "java" executable is the one provided by MQ, then its full path name is: "C:\Program Files\IBM\MQ\java\jre\bin\java"

```
C:/> "C:\Program Files\IBM\MQ\java\jre\bin\java" JmsJndiProducer -i
file:/C:/ProgramData/IBM/MQ/JNDI-Directory -c CF1 -d T1
Initial context found!
Sent message:
  JMSMessage class: jms text
                  ทมไไ
  JMSType:
  JMSDeliveryMode: 2
  JMSDeliveryDelay: 0
  JMSDeliveryTime: 1676753963274
  JMSExpiration: 0
  JMSPriority:
                  4
  JMSMessageID:
ID:414d5120514d39332020202020202020eabeef63013c0040
  JMSTimestamp: 1676753963274
  JMSCorrelationID: null
  JMSDestination: topic://TOPIC1
  JMSReplyTo: null
  JMSRedelivered: false
    JMSXAppID: JmsJndiProducer
    JMSXDeliveryCount: 0
    JMSXUserID: mqm
    JMS IBM ConnectionID:
414D5143514D39332020202020202020EABEEF63003B0040
    JMS IBM PutApplType: 28
    JMS IBM PutDate: 20230218
    JMS IBM PutTime: 20592253
JmsJndiProducer: Your lucky number today is 260
SUCCESS
```

Notice the output line that indicates that the destination is a queue, and provides the name of the topic:

JMSDestination: topic://'TOPIC1'

+++ end +++