The DatabaseRoute and DatabaseRetrieve nodes construct their JDBC connections by using connection details that are stored in the registry of the integration node, which contain standard DBMS JDBC provider settings.

Use the following instructions to create a JDBC provider entry if you are using an Oracle database. You can use the **mqsicreateconfigurableservice** command, which is provided in the following script, to create a new JDBC provider entry called SIMPLERROUTEDB. This entry is referenced by the **Data Source Name** property field in the database nodes that are used in sample. The following instructions also configure the JDBC provider entry with a user identifier and password for the integration node to use when accessing the database.

1. Create a JDBC provider entry for the database:
   1. Start the integration node and ensure that an integration server has been deployed to the integration node. For example, to start an integration node, open a command console window in which you can run IBM Integration Bus commands, and enter the following command:

mqsistart IB9NODE

* 1. Open a new document in a text editor, copy and paste the following script, which creates the JDBC provider entry SIMPLERROUTEDB to provide JDBC database access for the Simplified Database Routing sample, into a text editor.

You must replace the following place holders in the script with values appropriate to your environment.

* + - <SID> The Oracle System identifier that identifies the database containing the tables used by this sample.
    - <JARS URL> The local directory path, on the system on which the integration node is running, where the JAR file that contains the type 4 driver class is located.
      * On Windows: <ORACLE HOME>\jdbc\lib
      * On Linux: <ORACLE HOME>/jdbc/lib/
    - <PORT NUMBER> The port number on which the database server is listening. The Oracle default is 1521.
    - <SERVER NAME> The name of the host on which Oracle is running; for example localhost.

mqsideleteconfigurableservice IB9NODE -c JDBCProviders -o SIMPLERROUTEDB

mqsicreateconfigurableservice IB9NODE -c JDBCProviders -o SIMPLERROUTEDB

-n connectionUrlFormat,connectionUrlFormatAttr1,description,jarsURL,portNumber,

serverName,type4DatasourceClassName,type4DriverClassName

-v "jdbc:oracle:thin:[user]/[password]@[serverName]:[portNumber]:[connectionUrlFormatAttr1],

<SID>,Simplified Database Routing Sample Database,<JARS URL>,<PORT NUMBER>,

<SERVER NAME>,oracle.jdbc.xa.client.OracleXADataSource,oracle.jdbc.OracleDriver"

* 1. Save the file. The following instructions assume that you have called the file simplifieddbrouting.bat
  2. In the command console, navigate to the folder that contains simplifieddbrouting.bat, and enter the following command to run the script:

simplifieddbrouting.bat

When you run the script, it performs the following tasks:

* + - Deletes any existing JDBCProvider entry named SIMPLERROUTEDB
    - Creates a new JDBCProvider entry named SIMPLERROUTEDB for use by the Simplified Database Routing sample
  1. Wait for the script to finish running. If you are running the script for the first time, the following message is displayed:
  2. BIP8984W: Configurable service object name 'JDBCProviders : SIMPLERROUTEDB' was not found.
  3. The user has tried to delete or modify a configurable service object name that does not exist.
  4. Check the names of the existing configurable services using the mqsireportproperties command.
  5. Then modify the external resource object name and reissue the command.
  6. BIP2087E: Broker IB9NODE was unable to process the internal configuration message.
  7. The entire internal configuration message failed to be processed successfully.
  8. The internal configuration message failed to be processed,
  9. use the messages following this message to determine the reasons for the failure.
  10. BIP8036E: Negative response received.
  11. This command sends an internal configuration message to the broker, the response received
  12. indicated that the internal configuration message was unsuccessful.
  13. Check that the WebSphere MQ transport is available. Check the system log for further information.

Ignore this message. The message is displayed because the script attempts to delete an existing JDBCProvider registry entry called SIMPLERROUTEDB before it re-creates the entry. If you have not run the script before, the entry is not found in the registry.

* 1. You can enter the following command to check that the default integration node registry was correctly updated with the sample SIMPLERROUTEDB JDBCProvider entry:

mqsireportproperties IB9NODE -o SIMPLERROUTEDB -c JDBCProviders -r

If the update was successful, the reported property output from the command matches the following example:

JDBCProviders

SIMPLERROUTEDB

connectionUrlFormat='jdbc:oracle:thin:[user]/[password]@[serverName]:[portNumber]:[connectionUrlFormatAttr1]'

connectionUrlFormatAttr1='orc1'

connectionUrlFormatAttr2=''

connectionUrlFormatAttr3=''

connectionUrlFormatAttr4=''

connectionUrlFormatAttr5=''

databaseName='default\_Database\_Name'

databaseType='default\_Database\_Type'

databaseVersion='default\_Database\_Version'

description='Simplified Database Routing Sample Database'

environmentParms='default\_none'

jarsURL='C:\oracle\oraxx\jdbc\lib'

portNumber='1521'

securityIdentity='default\_User@default\_Server'

serverName='localhost'

type4DatasourceClassName='oracle.jdbc.xa.client.OracleXADataSource'

type4DriverClassName='oracle.jdbc.OracleDriver'

BIP8071I: Successful command completion.

1. Specify the user identifier and password to associate with the JDBC provider SIMPLERROUTEDB:

Use the **mqsisetdbparms** and **mqsichangeproperties** commands to specify a user identifier and password for the integration node to use with JDBC provider SIMPLERROUTEDB. This user identifier must be the same user identifier that you used when you created the database.

* 1. Enter the following command to associate the user identifier and password with a security identity:

mqsisetdbparms IB9NODE -n jdbc::mySecurityIdentity -u <user ID> -p <password>

* 1. Enter the following command to associate the security identity, which you defined in the preceding step, with the **securityIdentity**property of the JDBC provider SIMPLERROUTEDB:

mqsichangeproperties IB9NODE -c JDBCProviders -o SIMPLERROUTEDB -n securityIdentity -v mySecurityIdentity

* 1. You must stop and restart the integration node for the changes to the JDBC provider to become available to a message flow that is already deployed. If you have already deployed the sample, stop and restart the integration node.

For further information, see [Enabling JDBC connections to the databases](https://www.ibm.com/support/knowledgecenter/SSMKHH_9.0.0/com.ibm.etools.mft.doc/ah61300_.htm?view=kc) in the IBM Integration Bus documentation and the "Making the JDBC provider service available to the DatabaseRoute node" section in [DatabaseRoute node](https://www.ibm.com/support/knowledgecenter/SSMKHH_9.0.0/com.ibm.etools.mft.doc/ac37380_.htm?view=kc) in the IBM Integration Bus documentation.