

IBM App Connect Enterprise 12 Application Development II

WM687 (Classroom)

ZM687 (Self-paced)

Course description

This course provides instruction in using IBM App Connect Enterprise to connect to IBM MQ and reference databases. The first part of this course covers traditional use cases for IBM App Connect Enterprise including using App Connect Enterprise in conjunction with JMS, web services, and IBM MQ. The second half extends topics discussed in prior courses including using the Graphical Data Mapping editor to create a DFDL model and using ESQL in a Compute node. The course also covers referencing a database in a message flow application.

For information about other related courses, see the IBM Training website:

**ibm.com**/training

General information

Delivery method

Classroom or self-paced virtual classroom (SPVC)

Course level

ERC 1.0

Product and version

IBM App Connect Enterprise V12

Audience

This course is designed for experienced integration specialists and senior-level developers with experience in application development and messaging middleware applications interested in becoming an IBM App Connect Enterprise Developer.

Learning objectives

After completing this course, you should be able to:

* Build a message flow that manages workload distributions between two integration nodes
* Develop a message flow that manages workload distributions between two integration nodes by using IBM MQ
* Design a message flow that reads from and writes to a Java Messaging Service destination queue
* Create a message flow that transforms an XML input into a JSON Array output structure using a Mapping node
* Create a Data Format Description Language (DFDL) message model schema in a shared library and test it by parsing the input data
* Develop an application that serializes a message with an XML message as input
* Transform messages using the Compute or a JavaCompute node
* Design a message flow that reads from and writes to a database by using Open Database Connectivity (OBDC) and Java Database Connectivity (JDBC) connections

Prerequisites

Before taking this course, you should have taken the following course:

* WM686: IBM App Connect Enterprise 12 Application Development I

Duration

2 days

Skill level

Intermediate

Notes

The following unit and exercise durations are estimates, and might not reflect every class experience. If the course is customized or abbreviated, the duration of unchanged units will probably increase.

This course is a new course

Course agenda

|  |
| --- |
| Course introductionDuration: 15 minutes |

|  |
| --- |
| Unit 1. Connecting to IBM MQDuration: 1 hour and 30 minutes |
| Overview | In this unit, you learn how to process IBM MQ messages in a message flow. You also learn about the connection and topology options for IBM MQ. |
| Learning objectives | After completing this unit, you should be able to:* Describe the IBM MQ connection options
* Examine the properties of the IBM MQ nodes
* Predict the location of the message if a runtime error is encountered during message flow processing
* Attach an MQEndpoint policy to one or more IBM MQ nodes in a message flow to control connection details at run time
* Use tools to test a message flow that connects to IBM MQ
 |

|  |
| --- |
| Exercise 1. Connecting to IBM MQDuration: 1 hour and 30 minutes |
| Overview | In this exercise, you create an IBM MQ queue manager and IBM App Connect Enterprise integration nodes that use the same queue manager as a default queue manager. Next, you create and test a simple flow that gets a message from an input queue and puts a message to an output queue. Finally, you test that the workload is evenly distributed between the integration nodes by sharing a queue manager and putting multiple messages to the input queue. |
| Learning objectives | After completing this exercise, you should be able to:* Create an integration node that uses a default IBM MQ queue manager
* Share a default IBM MQ queue manager with multiple integration nodes
* Create a message flow that gets a message with an MQInput node and puts a message with an MQ Output node
* Edit a BAR file
* Manually deploy a BAR file
* Verify that the integration nodes that share a queue manager also share the workload
 |

|  |
| --- |
| Unit 2. Processing JMS, HTTP, and web service messagesDuration: 1 hour |
| Overview | This unit introduces IBM App Connect support for HTTP, JMS, and web services. |
| Learning objectives | After completing this unit, you should be able to:* Describe how to use message flow applications with JMS
* Describe how message flow applications can support Hypertext Transfer Protocol (HTTP) and SOAP messages
* Explain how the Web Services Definition Language (WSDL) file is used to develop web services message flows
 |

|  |
| --- |
| Exercise 2. Java Message Service processingDuration: 30 minutes |
| Overview | In this exercise, you create two simple message flows. You create a message flow that receives a message over HTTP and sends the message to a JMS destination queue. You create a second message flow that uses a JMS input node to read the message from the JMS queue. You wire the JMS input node to a Trace node and configure the Trace node to write out the value received from the JMS destination queue. In order to enable the JMS communication, you define an MQ listener for the ACEQM queue manager. You also create a JNDI directory to house the bindings. Using a JMS administration tool provided by IBM, you create the queue connection factory and configure the bindings. After deploying the applications, you test them using the Flow exerciser and verify the JMS message by viewing the results of the trace file. |
| Learning objectives | After completing this exercise, you should be able to:* Define a listener for queue manager
* Create a queue connection factory and JNDI bindings
* Build a message flow to write to a JMS destination queue
* Build a message flow to read from a JMS destination queue
 |

|  |
| --- |
| Unit 3. Using a Mapping node to create a JSON array message.Duration: 1 hour |
| Overview | In this unit, you learn how to use the Graphical Data Mapping editor to create and edit graphical data maps. |
| Learning objectives | After completing this unit, you should be able to:* Use a the Graphical Data Mapping editor to map logical messages
* Run message maps within message flows
* Contrast core, structural and database transforms to one another
* Describe JSON Object messages
 |

|  |
| --- |
| Exercise 3. Using a Mapping node to create a JSON Array messageDuration: 1 hour |
| Overview | In this exercise, you learn how to build a message flow that transforms the input from input from an XML structure into a JSON Object output using a Mapping node and sends this back to the HTTP request. |
| Learning objectives | After completing this exercise, you should be able to:* Use a Mapping node to create a new JSON Array message in a message flow.
* Formulate a Concat transform using a nested view in the Graphical Data Mapping editor
 |

|  |
| --- |
| Unit 4. Modeling the dataDuration: 1 hour and 30 minutes |
| Overview | In this unit, you learn about the message modeling options that IBM App Connect Enterprise provides. The unit concentrates on using Data Format Descriptive Language to model the data. |
| Learning objectives | After completing this unit, you should be able to:* Explain the concepts of message models and how they are used to help message transformation
* List the parsers that are available for use within IBM App Connect Enterprise
* Create and modify a DFDL model
* Use importers to create data models
* Choose the appropriate message validation options
* Organize and administer message models
* Reference message models in message flows
 |

|  |
| --- |
| Exercise 4. Creating a Data Format Descriptive Language (DFDL) modelDuration: 1 hour |
| Overview | In this exercise, you create a DFDL message model schema file in a shared library. The DFDL schema that you create defines a delimited text file. You test the model by using the Test Parse Model and Test Serialize Model options that are provided in the DFDL schema editor. |
| Learning objectives | After completing this exercise, you should be able to:* Create a DFDL message model schema file in a shared library
* Define the logical structure and physical properties of the message model elements
* Test a DFDL schema by parsing test input data
* Test a DFDL schema by serializing test data to create an output file
 |

|  |
| --- |
| Unit 5. Using Compute nodes to transform messagesDuration: 1 hour and 30 minutes |
| Overview | In this unit, you learn about runtime errors in message flow applications. You learn how IBM App Connect Enterprise responds to a runtime exception and what happens to the data that is being processed. You also learn how transactions can be coordinated. The unit also introduces some of the tools and techniques that IBM App Connect Enterprise offers for problem determination and debugging, and how to support explicit error handling within a message flow application. |
| Learning objectives | After completing this unit, you should be able to:* Use the TryCatch and Throw nodes to implement explicit error handling within a message flow
* Describe the structure of the ExceptionList component of the message assembly, and the role it plays in runtime error handling
* Use problem determination tools to debug message flows
* Use help resources to learn more about the product and find information about resolving problems
 |

|  |
| --- |
| Exercise 5. Transforming data by using the Compute or Java Compute nodeDuration: 1 hour and 30 minutes |
| Overview | In this exercise, you create a message flow application that uses ESQL and a Compute node or Java and a JavaCompute node to transform message content. |
| Learning objectives | After completing this exercise, you should be able to:* Use a Compute node or JavaCompute node in a message flow application to transform a message
 |

|  |
| --- |
| Unit 6. Referencing a database in a message flow applicationDuration: 1 hour |
| Overview | In this unit, you learn about the database functions and nodes in IBM App Connect Enterprise. You also learn about defining a database service for database assets. |
| Learning objectives | After completing this unit, you should be able to:* Use database message processing nodes to modify messages and control message processing
* Configure database nodes to access user databases
* Describe the differences between ESQL and SQL SELECT
* Create a database definition file
 |

|  |
| --- |
| Exercise 6. Referencing a database in a mapDuration: 1 hour and 30 minutes |
| Overview | In this exercise, you use a Database node in a message flow to store a message in a database. You also import a COBOL Copybook and XML schema to create a data model, and use the Graphical Data Mapping editor to transform the message. |
| Learning objectives | After completing this exercise, you should be able to:* Create a shared library that contains data models that describe the input and output data
* Import a COBOL Copybook to create a DFDL schema file
* Reference a shared library in a message flow application
* Discover database definitions
* Define database connectivity
* Add a Database node to a message flow
* Create the logic to store a message in a database
* Use the Graphical Data Mapping editor to map message elements
* Reference an external database when mapping message elements
 |

|  |
| --- |
| Unit 7. Course summaryDuration: 30 minutes |
| Overview | This unit summarizes the course and provides information for future study. |
| Learning objectives | After completing this unit, you should be able to:* Explain how the course met its learning objectives
* Access the IBM Training website
* Identify other IBM Training courses that are related to this topic
* Locate appropriate resources for further study
 |

For more information

To learn more about this course and other related offerings, and to schedule training, see **ibm.com**/training

To learn more about validating your technical skills with IBM certification, see **ibm.com**/certify

