



MQ Training

MQGem Software supplies online training modules about IBM® MQ.

Format

The modules have a format where you read an introduction about the subject at hand, and then do some worked exercises on your own machine to try out what you have learned. You can choose to use command line MQSC commands or GUI tools (such as MQ Explorer or MO71) to interact with your queue manager, depending on your preference.

Full instructions are provided for MQSC, MQ Explorer, MO71, IBM i 5250 Panels, and ISPF 3270 Panels; and for UNIX®, Windows®, IBM i and z/OS® platforms, so you can learn in the same environment where you will use the skills going forward. You choose the path through the module, by selecting the tool or platform you want to use, no paging forward through instructions that are not relevant to you. Modules are short lessons, designed to take between 2 and 4 hours to complete.

The modules are provided on a Web based learning platform to which you will be enrolled on the modules you choose. The dashboard shows your progress as you work through the modules and remembers where you are in the instructions if you leave and later come back, so if you need to dip in and out of the module as time is available, you can do that – one of the advantages of online self-paced training.

Online training provides many advantages over classroom based education, such as:-

- Going at your own pace
- Being able to do it piece meal, an hour here and an hour there, as your current job schedule allows
- No need to have the whole department away from their jobs at the same time
- Considerably cheaper than classroom based education

Each module finishes with a short quiz so you can test yourself to see how much you remember of what you were taught. While a score is given, credit is given for completing the module regardless of the quiz score.

Pricing

With the exception of module MQG-101-00 which is FREE, modules are priced at 80 NZD per person per module (unless stated otherwise). You can either purchase an individual module on the learning platform website through PayPal just before you decide to do it, or you can purchase several modules (and for several individuals) in a block by getting in touch at learn@mqgem.com. We can invoice you for a block and, once paid, provide logons to the learning platform for each individual to run through the module(s).

Modules

The following modules are currently available:

MQG-101-00 - First steps with MQ – available for FREE!

In this module you will be introduced to some of the basic constructs in IBM MQ; how to use the commands available; the samples provided; and how to get information about any errors emitted by the product. It is offered free to ensure that you have this basic pre-requisite level of skills to do the later modules, and to allow you to try out an MQGem training module and see what the format is like.

Part 1: Getting Started

You get started with IBM MQ by creating a queue manager, and a queue, and learning how to run the MQ supplied sample programs to put and get messages using your queue. You learn about the reason code that IBM MQ returns to applications to indicate when there is a problem, and you try out changing an attribute on a queue that causes a non-zero reason code when you try to use the queue again.

Part 2: A closer look at a message

In part two you take a closer look at an MQ message. You learn about fields in the Message Descriptor, and how to browse messages on a queue as well as how to destructively get those messages off the queue. You learn about priority messages, and create alias queues that cause different default priorities to be set. You learn how that can change the order of the messages returned to a getting application.

Part 3: Running Commands from a Script

The final brief part of this module teaches you how to run an MQSC script from the tool you are making use of to administer your queue manager. You will need this skill in later modules, so you learn it now to be ready.

Platform Coverage: This module provides instructions for IBM MQ on UNIX, Windows, IBM i and z/OS.

Module Duration: It is expected that this module will take you 1 - 1½ hours.

MQG-101-05 - Remote Messaging

In this module you will be introduced to the basic constructs for sending messages from one queue manager to another – remote messaging. After setting up a channel and learning how to diagnose channel errors, you will also cover automatically starting and ending channels; and learn what happens to undelivered messages.

Part 1: Basic Definition

In this first part you learn how to set up a basic sender and receiver channel to join together two queue managers. You learn how the channel behaves when an error occurs, and how the errors are reported for you to read. You test your investigative skills with a selection of failures, to correct them and get all the channels up and running. You define a remote queue and use it to send a message across your channel to the target system.

Part 2: Automatic Start and End

In part 2 you learn how to have channels automatically start when messages arrive on their transmission queue to be delivered by using IBM MQ Triggering, and also how to have them gracefully end when there are no more messages for them to send by utilising the Disconnect Interval.

Part 3: Undeliverable Messages

In this third part you will be introduced to the problems of undeliverable message and how IBM MQ channels deal with these messages. You will learn about Dead-letter queues, and how to choose to use it or not, and about the message retry facility on receiving channels.

Platform Coverage: This module provides instructions for IBM MQ on UNIX, Windows, IBM i and z/OS.

Module Duration: It is expected that this module will take you 2-4 hours.

MQG-101-06 - Configuring Clients for Connectivity

In this module you will be introduced to a method of running an MQ application that allows it to run on a different machine from where the queue manager, and therefore the queues, reside. This mechanism is called a Client Connection. You will learn what needs to be configured on the queue manager in order for client connections to run, and how to use several client connection techniques, including MQSERVER environment variable and Client Channel Definition Table (CCDT). Lastly we look briefly at direct use of connection information from an application, although this is not a programming course, so no coding experience is required in this module.

Part 1: Connecting as a client

To start with, you learn the configuration needed on the queue manager to allow a client connection to be used, and you use the most basic of client configuration, the MQSERVER environment variable. You investigate different things that can change on the client or queue manager machine, and learn where to look for error reporting on both machines to discover what went wrong.

Part 2: Fully Configurable client

The client configuration you used in Part 1 is very basic, so in this part you learn how to fully configure a client channel using the Client Channel Definition Table (CCDT). You learn that you can have multiple channels defined in a CCDT and how the IBM MQ Client chooses which one to use including variations on the choosing algorithm that allow different spreads of client connections across the targets.

Part 3: Client configuration from inside an application

In this part you learn that it is also possible to fully configure the client details from within application code. You learn how this over-rides what you have done in the earlier parts.

Platform Coverage: This module provides instructions for IBM MQ on UNIX, Windows, IBM i and z/OS (although when using IBM i or z/OS the instructions only cover those platforms as queue managers, so another platform will be required as well as those to act as the client for this module).

Module Duration: It is expected that this module will take you 2-4 hours.

MQG-101-10 - Starting out with Queue Manager Clustering

In this module you will be introduced to the basic constructs for queue manager clustering, allowing auto-discovery of queues within the cluster, and automatic connectivity between queue managers. You will set up a cluster of queue managers, learn how queues are advertised round the cluster, and try out workload balancing of messages.

Part 1: Basic Definition

In this part you set up a basic cluster with three queue managers, learning what happens if a definition is wrong, and how to spot it using the various display commands that show you your cluster state.

Part 2: Clustering Features - Automatic object discovery

In Part 2 you look into one of the major reasons to make use of an IBM MQ Queue Manager cluster, that of automatic object discovery. You learn how cluster defined resources are propagated round the cluster on a need to know basis, and what happens if the full repository is not available when a change is made that needs to be propagated. You also learn how the full repository is not a single point of failure all the time for a cluster, but only when changes are needed.

Part 3: Clustering Features - Workload Balancing of Messages

In Part 3 you look into the other major reason to make use of an IBM MQ Queue Manager cluster, which is the workload balancing of messages. You define cluster queues of the same name on members of your cluster and see how the messages are delivered. You investigate the different bind types and learn how they behave and how to configure them, either administratively or directly in an application.

Platform Coverage: This module provides instructions for IBM MQ on UNIX, Windows, IBM i and z/OS.

Module Duration: It is expected that this module will take you 2-4 hours.

MQG-101-30 - Getting going with Authorization

In this module you will be introduced to authorisation features in IBM MQ and how to use those to allow an application to access MQ resources (e.g. put to a queue), or a user to issue certain MQ commands (e.g. change a queue definition). Take module MQG-101-32 "Using MQ Authentication" to learn about that side of securing your queue manager.

Part 1: What's My user id?

This very short introductory part shows you how to determine the user id that the queue manager knows you as.

Part 2: Granting Authorities

In this part you learn how to grant authorities to non-privileged user ids to allow applications that they run to access IBM MQ resources such as queues. You learn about the various patterns you can use to name resources when using generic profiles, and how the queue manager reports authorisation failures, both in the error log and with event messages. You also learn what happens when you have overlapping profiles that could both match the user id, resource and entity in question.

Part 3: Administrative authorities

In this part you learn about the authorities that can be granted to allow non-privileged user ids to issue MQ commands, for example ALTER QLOCAL or START CHANNEL. You learn how to set up a read-only user, that can look but not change anything, and you test out your investigative skills to diagnose some authorization failures.

Platform Coverage: This module provides instructions for IBM MQ on UNIX, IBM i and Windows. It does not cover z/OS.

Module Duration: It is expected that this module will take you 2½ - 5 hours.

MQG-101-31 - Securing channels using Digital Certificates (SSL/TLS)

In this module you will be introduced to creating Certificate Authority (CA) signed digital certificates and configuring the queue manager and MQ Channels to make use of those certificates to secure the channels. This module will cover channels between two queue managers, and Client channels (both 'C' and Java) using these digital certificates.

Part 1: Channels and Certificates

In this part you set up two queue managers with a channel between them, and create a certificate for the receiving queue manager. You learn various errors that you might encounter when setting up these certificates on the queue manager, and find out where the queue manager writes errors to tell you what happened. You also learn what channel status output tells you about your connection.

Part 2: Mutual Authentication

In this part you continue with your queue managers from Part 1 and set up mutually authenticated certificates, that is both the sending and receiving queue managers have a certificate. This is the expected setup you would have on queue manager channels, but it useful to learn to set up certificates in a piece meal fashion to make it easier to debug when it doesn't work. You also learn about Distinguished Name matching in this part, and learn to diagnose various channel set up errors.

Part 3: Securing a Client Connection

In this part you learn how to set up certificates for client applications. The concepts are the same, but the syntax and/or configuration is different. You do this for a 'C' application and a Java application, learning that the tools, syntax and configuration are different in each of these environments.

Platform Coverage: This module provides instructions for IBM MQ on UNIX, Windows, IBM i and z/OS.

Module Duration: It is expected that this module will take you 2-4 hours.

MQG-101-32 - Using MQ Authentication

In this module you will be introduced to authentication features in IBM MQ and how to use those and assign a user ID to an application connection, or an administrative user, in order for your authorization commands to be tied to that user ID. Take module MQG-101-30 "Getting going with Authorization" to learn about that side of securing your queue manager.

Part 1: What's My user id

This very short introductory part shows you how to determine the user id that the queue manager knows you as.

Part 2: Connection Authentication

This part introduces you to the feature in IBM MQ where the queue manager can be configured to check passwords when an application connects before it is allowed to continue. You learn that there are independent configuration controls for client connections and local connections.

Part 3: Client application authentication - setting an MCAUSER

In this part you learn how to set an MCAUSER to be the identity used by a channel when it runs. You look initially at a statically defined MCAUSER and then at the Channel Authentication (CHLAUTH) feature of IBM MQ and how it can be used to block connections and also to allow them in, setting an MCAUSER at the same time. You learn how CHLAUTH can be combined with Connection Authentication that you looked at in Part 2.

Part 4: Other channel authentication rules

In this part you look at some of the other CHLAUTH rule types, namely BLOCKADDR and BLOCKUSER. You learn how it is possible to allow in a privileged user over a single client channel while continuing to ban such access on all other channels.

Part 5: Channel Authentication rules for QMgr Channels

In the final part you learn that CHLAUTH rules also apply to queue manager channels. In the previous parts of the module you were focussed on client channels.

Platform Coverage: This module provides instructions for IBM MQ on UNIX, Windows, IBM i and z/OS.

Module Duration: It is expected that this module will take you 2½ - 5 hours.

MQG-101-60 z/OS Queue Manager - Getting Started

While many of the features of the IBM MQ product are the same across all platforms, for example, the MQSC commands or the MQ API programming verbs, there are some marked differences when getting to know your queue manager between the distributed platforms and the z/OS platforms. This module aims to introduce you to the z/OS queue manager and show you some of the helpful hints and tips that will enable you to do a number of the other modules without feeling that the z/OS queue manager is too different and hard to learn.

Part 1: A tour of your z/OS Queue Manager - Started Tasks

This part introduces you to the two main address spaces of a z/OS queue manager, the MSTR and the CHIN. It looks at the started task JCL and the various things referenced in there. It also looks at the input datasets and how to run commands at start-up of a task, and how certain MQ commands have to be run from certain places.

Part 2: A tour of your z/OS Queue Manager - Parameter Module

This part introduces you to the ZPARM of the queue manager. It shows you where you can see the values, the sample scripts they are built from, and the commands that can over-ride them. It does not teach what each attribute in ZPARM is, they will be covered as required by other modules on specific subjects. Instead it teaches how to make an update to the ZPARM module, and how to see and over-ride your change.

Part 3: A tour of your z/OS Queue Manager - Page Sets and Buffer Pools

This part introduces you to page sets and buffer pools and the various commands that allow you to look at their usage. You learn about storage classes, and create a queue using a non-default storage class. You experiment with persistent and non-persistent messages and look at page set and buffer pool usage as a result of the messages you put on the queue.

Part 4: A tour of your z/OS Queue Manager - Channel Initiator Address Space

This part is a brief tour of the CHIN address space and the tasks that make it up. The worked exercise will get you to generate a SNAP dump by pointing your web-browser at the listener.

Platform Coverage: This module provides instructions for IBM MQ on z/OS.

Module Duration: It is expected that this module will take you 1-2 hours.

The following are in-progress to be available soon:

MQG-101-71 – Application Triggering

In this module you will be introduced to the basic concepts for automatically starting IBM MQ applications when messages turn up on a queue needing to be processed.

Platform Coverage: This module provides instructions for IBM MQ on UNIX and Windows.

There are a number more planned, and if you have suggestions for other subjects, please get in touch.



© Copyright MQGem Software Ltd 2017, 2018

Email learn@mqgem.com
Web www.mqgem.com/training.html
Blog mqgem.wordpress.com
Twitter [@MQGem](https://twitter.com/MQGem)

IBM and z/OS are trademarks of International Business Machines Corporation in the United States, other countries or both.

Windows is a trademark of Microsoft Corporation in the United States, other countries, or both.

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