

BMC AMI DevX Code Pipeline 22: Fundamentals Administering



[Learning Path >](#)

Course Code: CPIW-FADM-F2201

Modality	Duration	Applicable Versions	Target Audience
Instructor-Led Training (ILT)	3 Days	BMC AMI DevX Code Pipeline 22.01	Administrators

Course Overview

BMC AMI DevX Code Pipeline is a modern mainframe CI/CD tool that can ensure your mainframe code pipelines are secure, stable, and streamlined throughout the DevOps lifecycle. With Code Pipeline, you'll be confident that you can quickly and safely build, test, and deploy mainframe code. As a Code Pipeline Administrator, you will use Code Pipeline to maintain the REXX Execs, JCL, ISPF Panels, Messages, and Skeletons used by Code Pipeline.

This three-day class combines classroom instruction with laboratory exercises to guide students through Code Pipeline Administering concepts. Students will develop skills in product configuration, effective administration of Code Pipeline, using maintenance functions, understanding how to troubleshoot problems, and more.

Prerequisites

- NA

Recommended Trainings

- BMC AMI DevX Code Pipeline (ISPW) 22.x: Fundamentals Developing (WBT)

Learning Objectives

- Describe the key features and benefits of BMC AMI DevX Code Pipeline
- Describe the types of Maintenance functions
- Describe the Code Pipeline architecture and its components
- Explain the Code Pipeline change lifecycle
- Define a Code Pipeline stream
- Describe how to use M-Tables
- Understand how generate processing is defined
- Describe generate variables and their parameters
- Illustrate how to use Code Pipeline skeletons
- Create new exits
- Explain how to make the Component Reference functional
- Understand Set Containers and their attributes
- Explain how the approval process works
- Describe the deploy maintenance functions
- Illustrate the deployment process
- Explain Batch API or External Call Interface and its uses
- Explain how to prepare for implementing security
- Explain how to troubleshoot dataset migration issues
- Describe troubleshooting when multiple component types are used in the same library
- Understand how to troubleshoot a generate

Course Modules

Module 1: Getting Started with BMC AMI DevX Code Pipeline Administration

- Describe key features and benefits of BMC AMI DevX Code Pipeline
- Explain the differences and uses of Code Pipeline test and production instances
- Describe the types of Maintenance functions
- Explain user-based licensing

Module 2: BMC AMI DevX Code Pipeline Configuration

- Describe the Code Pipeline architecture and its components
- Explain the different types of datasets defined in Code Pipeline
- Illustrate how to configure Topaz with Code Pipeline

Module 3: BMC AMI DevX Code Pipeline Change Lifecycle

- Explain the Code Pipeline change lifecycle
- Define a Code Pipeline stream

Module 4: M-Tables

- Describe the features of M-tables
- Describe the reference data options
- Explain Technical Support tables
- Describe the functionality of each of the Technical Support tables

Module 5: Generate Processing

- Explain the use of generate processing
- Describe generate variables and their parameters
- Explain parts registration and build map
- Illustrate how to use ISPF Skeleton JCL in Code Pipeline

Module 6: Exit Processing

- Describe Code Pipeline exit processing
- Explain the types of exit processing
- Create new exits

Module 7: Component Reference

- Describe Component Reference features
- Explain how to make the Component Reference functional
- Understand the use of parser User Exits

Module 8: Set Processing

- Describe Set Containers and their attributes
- Explain how the approval process works
- Explain the set scheduling system

Module 9: Deployment Processing

- Explain the deployment concepts
- Describe the deploy maintenance functions
- Explain the steps in Code Pipeline deploy
- Illustrate the deployment process

Module 10: APIs

- Explain the Batch API or External Call Interface and its uses
- Understand the different APIs available
- Describe Code Pipeline Rest API

Module 11: BMC AMI DevX Code Pipeline Security

- Describe how Code Pipeline security works
- Explain how to prepare for implementing security
- Define security rules
- Explain how to preview security rules
- Define security objects and methods

Module 12: Troubleshooting

- Describe the importance of monitoring changes made in the Code Pipeline and the ISPF systems
- Explain how to troubleshoot dataset migration issues
- Describe troubleshooting when multiple component types are used in the same library
- Understand how to troubleshoot a failed generate

Discount Options

Have multiple students? Contact us to discuss hosting a private class for your organization.

Contact us for additional information 