

VMware vRealize Automation Orchestration and Extensibility [V8.6]

Duration 5 Days

COURSE DESCRIPTION

During this five-day course, you focus on using VMware vRealize® Orchestrator™ to extend the functionality of VMware vRealize® Automation™. You learn how to provide XaaS (Anything as a Service) and implement Machine Lifecycle Extensibility using the VMware vRealize® Automation™ Event Broker. You also learn how to create vRealize Orchestrator workflows and vRealize Automation ABX actions. You learn about various features, including basic scripting implementation along with logic processing to implement a variety of functions to use in your environment. This course teaches implementing debugging, loops, conditions, and user interactions in vRealize Orchestrator.

The course introduces the new vRealize Orchestrator HTML 5 interface, along with API calls and REST functions, to give you the groundwork to implement a variety of plugins and scripts. This course is designed to give you the tools to craft custom solutions in the product.

COURSE OBJECTIVES

By the end of the course, you should be able to meet the following objectives:

- Describe the features, benefits, components, and integration of vRealize Orchestrator and vRealize Automation
- Describe the role of vRealize Orchestrator workflows and content elements in automation
- Use the vRealize Orchestrator client to access and navigate the vRealize Orchestrator platform, and to import and run library vRealize Orchestrator workflows
- Design, develop, and run custom reusable vRealize Orchestrator workflows
- Use VMware APIs in the vRealize Orchestrator workflow
- Integrate vRealize Automation with vRealize Orchestrator instances to extend infrastructure as a service (IaaS) system life cycle processes and to enhance and deliver custom IT services
- Use anything as a service (XaaS) to extend vRealize Automation into other enterprise systems
- Use the vRealize Automation event broker service to trigger specific vRealize Orchestrator workflows

COURSE OUTLINE

1. Course Introduction

- Introductions and course logistics
- Course objectives

2. Overview of vRealize Automation and vRealize Orchestrator

- Explain the features of vRealize Automation
- Identify the differences between vRealize Automation 8 and vRealize Automation Cloud
- List the services provided by vRealize Automation
- Explain the purpose of the cloud template designer
- Explain the features of vRealize Orchestrator
- Explain the architecture of vRealize Orchestrator
- List the vRealize Orchestrator plug-ins
- Run a vRealize Orchestrator workflow
- Use a workflow run

3. Using Schema Elements

- Explain the scriptable task
- Specify the programming language that is used in scriptable tasks
- Compare using a scriptable task with an action
- List the schema elements used to start workflows
- Supply the inputs and capture the outputs of a workflow
- Explain modular programming
- Call an action from a workflow
- Supply inputs to an action
- Rename the actionResult variable

4. Working with Variables

- Determine when to use an input, output, or variable
- List the different data types supported by vRealize Orchestrator
- Select valid JavaScript identifiers
- Adhere to the camel case naming convention
- Explain the purpose of binding
- Determine whether to use an inward or outward binding
- Pass data between schema elements in a workflow
- Use wrapping to simplify an overly complex workflow
- Use wrapping to add functionality to a workflow
- Use wrapping to protect your workflow and infrastructure from users
- Describe an API
- Consult the API documentation for VMware APIs
- Use vRealize Orchestrator API Explorer
- Define the inputs and output of a vRealize Orchestrator action
- Organize vRealize Orchestrator actions into modules
- Apply good modular programming habits when creating vRealize Orchestrator actions
- Customize a workflow's input form
- Make an input mandatory
- Ensure that end users supply a valid input
- Explain the use cases for User Interactions
- Specify the approvers for a User Interaction
- Respond to a User Interaction

5. Handling Exceptions, Logging, and Debugging

- Identify the function of exception handling
- Create an exception handler
- Use various methods to throw an exception within a workflow
- Configure the logging level in Control Center
- Access vRealize Orchestrator logs
- Send logging messages from a workflow
- Set debugging breakpoints
- Run a workflow through the vRealize Orchestrator debugger
- Monitor variable values

6. Branching and Looping

- Use the decision element in a workflow
- Use the decision activity element in a workflow
- Use the switch element in a workflow
- Create a looping workflow by using a Foreach element
- Create a looping workflow by using a decision element

7. Working with Assets

- Create a configuration element and add variables to it
- Create a workflow that uses variables to link to the configuration element
- Use a Configuration Element type variable and JavaScript to extract the values from a configuration element
- Create a resource element
- Create a workflow that uses a resource element
- Import a package
- Create a package
- Export a package

8. Working with Plug-Ins

- Understand what vRealize Orchestrator plug-ins do
- Download the vRealize Orchestrator plug-ins
- Locate the plug-in configuration workflows
- Configure the SSH plug-in
- Remotely execute a command on an SSH server
- Copy files from the vRealize Orchestrator server to an SSH server
- Copy files to the vRealize Orchestrator server from an SSH server
- Add a REST server to vRealize Orchestrator
- Add REST operations to vRealize Orchestrator
- Invoke a REST operation
- Add the vRealize Automation hosts to the inventory
- Run the workflows to manage vRealize Automation
- Run the REST operations to manage vRealize Automation
- Search the vRealize Automation API
- Add Windows PowerShell hosts to the inventory
- Invoke scripts on Windows PowerShell hosts
- Create Windows PowerShell workflows

9. Working with Versioning and Git

- List when new versions are created
- Increase the version number
- Compare two versions
- Revert to a version
- Create a Git Repository
- Push Changes to a Git Repository
- Pull Changes from a Git Repository

10. Working with Time-Related Schema Elements

- Schedule a workflow manually or programmatically
- Cause a workflow to sleep using JavaScript code
- Cause a workflow to sleep using a pre-defined schema element
- Cause a workflow to wait until a specific time and date
- Cause a workflow to wait upon an event

11. Introduction to Extensibility

- Explain what VMware means by the term extensibility
- Explain the Event Broker Service
- Explain Life Cycle Integration

12. Using the Event Broker

- Explain what the Event Broker Service does
- Explain what an event subscription is

- List the supported event topics
- Explain what an event subscription does
- Create an event subscription
- Monitor invocations of runnable items
- Explain what a payload is
- Explain what an event topic parameters are
- Extract the contents of a payload

13. Using ABX Actions

- Describe an ABX action
- Describe Function As A Service (FaaS)
- Explain when to use an ABX action and when to use a vRealize Orchestrator workflow
- Transmit information between vRealize Automation and an ABX action script
- Create an ABX action that is called by a vRealize Automation Subscription

14. Working with Services, Custom Resources, and Resource Actions

- Describe the Service Broker service cloud templates
- Create Content Source from the vRealize Orchestrator workflows
- Share content from the new Content Source
- Request the shared content from Catalog
- Create custom resources
- Create resource actions

15. Capturing vSphere Client Code

- Recognize how to capture code in the vSphere Client
- Use code in a vRealize Orchestrator workflow or PowerCLI script

WHO SHOULD ATTEND

Experienced VMware administrators, automation and orchestration specialists, system integrators, and private cloud and public cloud administrators

PREREQUISITES

This course requires the following prerequisites:

- Knowledge of VMware vSphere®
- Working knowledge of scripting or programming using JavaScript, Windows PowerShell, Perl, Java, Python, or similar languages. All code is provided during class.