

VMware vRealize Automation: Orchestration & Extensibility

Duration: 5 Days Course Code: VMRAOE Version: 8.6

Overview:

During this vRealize Automation 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.

Product Alignment: VMware vSphere® 7.0 VMware vRealize Automation 8.6 VMware vRealize Orchestrator 8.6 VMware vRealize® Suite Lifecycle Manager™ 8.6

Target Audience:

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

Objectives:

- By the end of the course, you should be able to meet the following objectives:
- Describe the features and benefits of integrating 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
- Use the vRealize Orchestrator client to import and run vRealize Orchestrator library workflows
- Design, develop, and run custom reusable vRealize Orchestrator workflows
- Integrate vRealize Automation with vRealize Orchestrator to deliver custom IT services
- Use the vRealize Automation event broker service to trigger specific vRealize Orchestrator workflows or ABX Actions
- Leverage the event broker to extend IaaS (Infrastructure-as-a-Service) machine lifecycle processes
- Use XaaS to extend vRealize Automation into other enterprise systems • Use VMware APIs to run vRealize Orchestrator workflows
- Use the vSphere Client Code Capture feature

Prerequisites:

This course requires the following prerequisites:

- Knowledge of VMware vSphere®
- VMware vRealize Automation: Install, Configure, Manage [V8.3] course or equivalent knowledge
- Working knowledge of scripting or programming using JavaScript, Windows PowerShell, Perl, Java, Python, or similar languages. All code is provided during class.

Content:

1 Course Introduction	• Debugging vRealize Orchestrator workflows	• Using Python
• Introductions and course logistics	6 Branching and Looping	• Using Nodejs
• Course objectives	• Using branching in vRealize Orchestrator workflows	• Using PowerShell
2 Overview of vRealize Automation and vRealize Orchestrator	• Using loops in vRealize Orchestrator workflows	• Using vRealize Automation Lifecycle
• Define the purpose of vRealize Automation	7 Working with Assets	12 Extending vRealize Automation with Event Broker
• Outline the purpose of vRealize Orchestrator	• Using configuration elements in vRealize Orchestrator	• Overview of vRealize Automation Event Broker
• Describe the main components of vRealize Automation	• Using resources in vRealize Orchestrator	• Creating vRealize Automation subscriptions
• Describe the main components of vRealize Orchestrator	• Using packages in vRealize Orchestrator	• Data exchange between vRealize Automation and vRealize Orchestrator
3 Creating Schema Elements	8 Working with Plug-Ins	13 Using ABX Actions
• Invoking JavaScript from a vRealize Orchestrator workflow	• Downloading and installing Plug-Ins	• Overview of Action Based Extensibility (ABX)
• Invoking a vRealize Orchestrator Workflow from a vRealize Orchestrator workflow	• Using the SSH plug-In in vRealize Orchestrator	• Comparison of vRealize Orchestrator and ABX
• Invoking an action from a vRealize Orchestrator workflow	• Using the REST plug-in in vRealize Orchestrator	• Creating ABX Actions scripts, REST, and flows
• Using vRealize Orchestrator workflows both synchronously and asynchronously	• Using the vRealize Automation plug-in in vRealize Orchestrator	• Using Day-2 Actions in vRealize Automation
4 Working with Variables	• Using the PowerShell plug-in in vRealize Orchestrator	• Describe the visualization capabilities of NSX Network Detection and Response
• Defining inputs, outputs, and variables in vRealize Orchestrator workflows	9 Working with Versioning and Git	14 Working with Services, Custom Resources, and Resource Actions
• Binding variables in vRealize Orchestrator workflows	• Using versioning in vRealize Orchestrator	• Using vRealize Orchestrator as a content source in vRealize Automation
• Wrapping vRealize Orchestrator workflows	• Using Git in vRealize Orchestrator	• Creating custom resources in vRealize Automation
• Using APIs and the API Explorer	10 Scheduling, Sleeping, and Waiting	• Creating resource actions in vRealize Automation
	• Using scheduling in vRealize Orchestrator	

<ul style="list-style-type: none"> • Creating actions in vRealize Orchestrator 	<ul style="list-style-type: none"> • Using sleeping in vRealize Orchestrator 	<ul style="list-style-type: none"> • Using Day-2 Actions in vRealize Automation
<ul style="list-style-type: none"> • Using vRealize Orchestrator input forms 	<ul style="list-style-type: none"> • Using waiting in vRealize Orchestrator 	<p>15 Using vSphere Client Code Capture</p>
<ul style="list-style-type: none"> • Handling user interactions in vRealize Orchestrator 	<p>11 Introduction to vRealize Automation Extensibility</p>	<ul style="list-style-type: none"> • Enabling vSphere Client code capture
<p>5 Handling Exceptions, Logging, and Debugging</p>	<ul style="list-style-type: none"> • Introduction to extensibility 	<ul style="list-style-type: none"> • Using vSphere Client code capture to capture code in vRO, Javascript, PowerCLI or other languages.
<ul style="list-style-type: none"> • Handling exceptions in vRealize Orchestrator workflows 	<ul style="list-style-type: none"> • Using ABX actions 	<ul style="list-style-type: none"> • Using the captured code in vRealize Orchestrator workflows or actions.
<ul style="list-style-type: none"> • Using logs in vRealize Orchestrator workflows 		

Further Information:

For More information, or to book your course, please call us on 00 966 92000 9278

training@globalknowledge.com.sa

www.globalknowledge.com/en-sa/

Global Knowledge - KSA, 393 Al-Uroubah Road, Al Worood, Riyadh 3140, Saudi Arabia