

Introduction to Infrastructure as Code Using Bicep

Duration: 3 days - Remote | **Focus Area:** Upgrade, Migration and Deployment | **Difficulty:** 300 - Advanced

Intended Audience

Primary Audience:

- DevOps / Cloud Engineers
- Cloud Solution Architects
- Infrastructure / Platform Engineers

Secondary Audience:

- Software Engineers
- Site Reliability Engineers

Overview

This workshop focuses primarily on how to use Bicep efficiently to provision Infrastructure components on Azure.

Objectives

After completing this training, you will be able to:

- Understand Infrastructure-as-Code Principles
- Learn Core concepts around Bicep
- Use Bicep to manage Azure Resources
- Deploy resources using Bicep Templates
- Build complex deployments using Bicep

An extended version of this Workshop is also possible by integrating Bicep with tools such as GitHub Actions, Azure DevOps Pipelines and including chalk-and-talk component focusing on customer's problem domain.

Key takeaways

- IaC Overview
- Bicep Introduction
- Bicep Advanced Topics – Loops & Conditions
- Bicep Modules – Local and Remote
- Visual Studio Code Tooling
- DevOps Tools and Bicep
- Git and Version Control

Agenda

- Bicep Fundamentals
- Tooling with Visual Studio Code
- Introduction to Bicep Templates
- Bicep – Parameters
- Bicep – Conditions and Loops
- Bicep – Modules (Local and Remote)
- Git and Version Control (Optional)
- DevOps – Pipelines & GitHub Actions (Optional)

Course details

Module 0: IaC Overview

- Infrastructure as Code – Overview
- Benefits
- Challenges
- Discussion

Module 1: Tooling with Visual Studio Code

- VS Code Tooling overview
- VS Code Installation and setup – Mac & Windows
- VS Code – Deeper Dive & Extensions
- Version Control – an Introduction
- Git Version Control – Overview

Module 2: Introduction to Bicep Templates

- Azure Resource Manager - Overview
- What is Bicep?
- Bicep & ARM
- Building a Bicep Template (Beginner)
- Labs
- Class Discussion

Prerequisites

Before attending this course, it is recommended that you meet the following criteria

- Sound knowledge of the Microsoft Azure platform.
- Be familiar with the Azure portal.

If you are new to these, here are a few references you can complete prior to class:

- [Microsoft Azure on Microsoft Learn](#)

For more information

Contact your Microsoft Account Representative for further details.

Module 3: Build reusable Bicep Templates by using parameters

- Introduction
- Understand Parameters
- Parameter Files
- Secure Parameters
- Parameters and Azure KeyVault
- Labs
- Class Discussion

Module 4: Using Conditions and Loops in a Bicep Template

- Introduction
- Conditional deployments
- Deploy resources using Loops
- Control loop execution and nest loops
- Using variable and output loops
- Labs:
- Class discussion

Module 5: Using Modules in a Bicep Template

- Introduction
- Create and use Bicep modules.
- Add parameters and outputs to modules.
- Local Modules
- Remote modules using Azure Container Registry (ACR)
- Layering a Bicep deployment using Modules
- Labs
- Class discussion

Module 6: Collaborating with Git in Visual Studio Code (Optional)

- Getting Started – Clone and Init
- Branching
- Edit and Commit
- Fetch/Pull/Push
- Pull Requests

Software and Hardware Prerequisites

Software

- Visual Studio Code - v1.6.6 or later
- Bicep extension for Visual Studio Code - v0.5.6 or later
- Azure Tools for Visual Studio Code
- Azure Resource Manager (ARM) extension for Visual Studio Code
- Azure CLI Tools extension for Visual Studio Code
- Azure CLI v2.40 or later
- Azure Bicep v0.5.0 or later
- Git command line tools v2.32.0 or later
- PowerShell – v7.2 or later
- Azure Subscription with some credits – Free Tier - <https://azure.microsoft.com/en-us/pricing/free-services/>
- Contributor Permissions to the subscription
- Azure Storage account (for CloudShell)

Hardware

- 1.6 GHz or faster processor
- 1 GB of RAM or more

Operating Systems

- VS Code is supported on the following platforms:
- MacOS OS X El Capitan (10.11+) or Later
- Windows 8.0, 8.1 and 10, 11 (32-bit and 64-bit)
- Linux (Debian): Ubuntu Desktop 16.04, Debian 9
- Linux (Red Hat): Red Hat Enterprise Linux 7, CentOS 7, Fedora 34