Microsoft Community Workshop

Introduction to Infrastructure as Code Using Bicep

Duration: 2 days – Remote | **Focus Area:** Upgrade, Migration and Deployment | **Difficulty:** 300 - Advanced

Intended Audience

Primary Audience:

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

Optional:

- Software Engineers
- Site Reliability Engineers
- Workload Owner

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.

Agenda

- Bicep Fundamentals
- Tooling with Visual Studio Code
- Introduction to Bicep Templates
- Bicep – Parameters
- Bicep – Conditions and Loops
- Bicep – Modules (Local and Remote)

Optional Content

Customers can opt-in for an **additional day** to cover the following content:

- Git and Version Control
- DevOps – Pipelines or GitHub Actions
- Customer driven Proof Of Concept (limited)
- Deliver infrastructure securely (Defender for DevOps)

This data sheet is for informational purposes only.
MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY

Course details

Module 0: IaC Overview

- Infrastructure as Code – Overview
- Benefits
- Challenges

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)

Module 3: Build reusable Bicep Templates by using parameters

- Introduction
- Understand Parameters
- Parameter Files
- Secure Parameters

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

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

Module 6: Migrate Azure Resources & ARM Templates to Bicep

- Guidance on migrating ARM templates to Bicep
- Converting resources in the portal to Bicep
- Testing Bicep modules and reviewing changes

Module 7: Deploy resources to Subscriptions & Management Groups

- Understanding deployment scopes
- Reviewing deployment commands

Module 8: Publish reusable code with Template Specs

- Introduction to Template Specs
- Comparison on Template Specs to Bicep
- Creating Template Specs

Module 9 (Optional): Collaborating with Git in Visual Studio Code

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

Module 10 (Optional): DevOps – Pipelines or GitHub Actions

- Introduction
- Authentication to Azure
- Basic deployment to Azure

Module 11 (Optional): Customer Driven POC (limited)

- Discuss a project or scenario where IaC might be used
- Start building our and structuring the IaC based on the scenario

This data sheet is for informational purposes only.
MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY

Microsoft

Prerequisites

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

- Foundational 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:

Master the basics of Azure: Learning Path

Access Requirements

During this course students will be deploying resources into Azure. All students either need an MSDN subscription or a sandbox subscription provided by the Organization with the Contributor RBAC permissions.

- Azure Subscription with some credits Free Tier

Software & Hardware

Software

- Visual Studio Code
- Bicep extension for Visual Studio Code
- Azure Tools for Visual Studio Code
- Azure Resource Manager (ARM) extension for Visual Studio Code
- Azure CLI
- Azure Bicep
- PowerShell
- Git command
- Azure Storage account (for CloudShell)

Hardware

- Bring Your Own Device (BYOD)
- Ensure you have permissions to install tools and extensions

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

This data sheet is for informational purposes only.
MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY

Microsoft