

Here are 30 Azure DevOps scenario-based interview questions and answers for a DevOps Engineer, along with real-time examples.

1. How do you set up a CI/CD pipeline in Azure DevOps for a microservices

architecture?

Scenario: You have multiple microservices in a repository and need to create an automated build and release pipeline.

Answer:

- Use Azure Pipelines with a YAML configuration for CI/CD.
- Define separate build and deployment stages

for each microservice.

- Example YAML:
trigger:
branches: include: -
main jobs: - job: Build
steps: - script: | dotnet
restore dotnet build --
configuration Release
displayName: 'Build
microservices'
- job:
Deploy steps: - script: |
az login --service-
principal -u <client-id>

```
-p <client-secret> --  
tenant <tenant-id> az  
webapp deploy --name  
my-microservice --  
resource-group my-rg --  
src-path myapp.zip  
displayName: 'Deploy  
to Azure'
```

2. How do you implement blue- green deployment

in Azure DevOps?

Scenario: Your team wants to minimize downtime while deploying a new version of an application.

Answer:

- Use Azure App Service Deployment Slots (e.g., staging and production).
- Deploy new code to

staging, run tests, and then swap slots for near-instant deployment.

- Example YAML:

```
- task:  
  AzureRmWebAppDeployment@4  
  inputs:  
    ConnectionType: 'AzureRM'  
    azureSubscription: 'MyAzureServiceConnection'  
    appType: 'webApp'
```

WebAppName: 'my-app'

DeployToSlotOrASE:

true

ResourceGroupName:

'my-rg' SlotName:

'staging'

3. How would you

configure

Infrastructure as

Code (IaC) using

Azure DevOps and

Terraform?

Scenario: You need to provision Azure resources using Terraform via Azure DevOps.

Answer:

- Store Terraform files in **Azure Repos.**
- Use a **Service Connection** to authenticate to Azure.

- Define a pipeline with Terraform steps:

```
- task: TerraformTaskV2@2
  inputs:
    command: 'init'
    backendType: 'azurerm'

- task: TerraformTaskV2@2
  inputs:
    command: 'apply'
    environmentServiceNameAzureRM:
      'MyAzureServiceConnec
```

tion'

- Use Terraform State stored in Azure Storage.

4. How do you manage secrets in Azure DevOps securely?

Scenario: You need to store API keys, database passwords securely in an Azure DevOps pipeline.

Answer:

- Use Azure Key Vault and link it to Azure DevOps via a variable group.
- Example YAML:

```
- task: AzureKeyVault@2
  inputs:
    azureSubscription: 'MyAzureServiceConnection'
    KeyVaultName: 'my-keyvault'
```

SecretsFilter: '*'

5. How do you implement automated rollback in Azure DevOps?

Scenario: Your deployment failed, and you need to revert to the previous version automatically.

Answer:

- Use Release Gates in Azure DevOps to validate the deployment.
- Rollback with a previous successful deployment via PowerShell:
- script: | az webapp deployment slot swap --resource-group my-rg --name my-app --slot staging

displayName: 'Rollback
to previous version'

6. How do you perform canary deployments using Azure DevOps?

Scenario: You want to
gradually roll out a new
release to a subset of
users before full
deployment.

Answer:

- Deploy to a separate slot (canary) and use **Traffic Routing** to direct a percentage of users:-
script: | az webapp traffic-routing set --name my-app --resource-group my-rg --distribution staging=50
displayName: 'Canary Deployment with 50%

traffic'

7. How do you implement security scanning in Azure DevOps?

Scenario: You want to scan your application code for vulnerabilities before deployment.

Answer:

- Use SonarQube,

WhiteSource, or Trivy for scanning.

- Example YAML:

```
- task:  
  SonarQubePrepare@5  
  inputs:  
    SonarQube:  
      'SonarQube Service Connection'  
    scannerMode: 'CLI'  
    configMode: 'manual'  
    cliProjectKey:  
      'MyProject'
```

8. How do you

handle zero-downtime deployments in Azure DevOps?

Scenario: You need to update your application without any downtime.

Answer:

- Use Rolling Updates with multiple instances in Azure Kubernetes

Service (AKS): - kubectl
rollout status
deployment my-app

9. How do you enforce branch policies in Azure Repos?

Scenario: Developers
should not merge code
without passing all tests.

Answer:

- Use Branch Policies to enforce:
 - Required reviewers
 - CI Build Validation
 - Work item linking

**10. How do you
automate database
migrations in Azure
DevOps?**

Scenario: Your team
needs to apply SQL

migrations automatically during deployments.

Answer:

- Use Entity Framework Migrations in the pipeline: - script: dotnet ef database update

11-30. Other Key

Azure DevOps

Scenarios

Would you like deeper explanations for any of

these?

Here's a detailed explanation of questions 11-30, with real-world examples and best practices:

11. How do you implement Helm charts in Azure DevOps for

Kubernetes?

Scenario: You need to package and deploy Kubernetes applications using Helm charts.

Solution:

- Store Helm charts in Azure Container Registry (ACR) or a separate repository.
- Use the Helm task in Azure DevOps pipelines.

- Example YAML:

```
- task:  
  HelmInstaller@1  
  inputs: helmVersion:  
    'latest'  
  - task:  
    HelmDeploy@0  
    inputs:  
      azureSubscription:  
        'MyAzureServiceConnection'  
      kubernetesCluster:  
        'my-aks-cluster'  
      command: 'upgrade'  
      chartType: 'FilePath'  
      chartPath: 'helm/
```

```
mychart' releaseName:  
'my-app'
```

12. How do you optimize pipeline execution time in Azure DevOps?

Scenario: Your pipelines take too long to complete, delaying releases.

Solution:

- Use parallel jobs for faster execution.
- Enable caching for dependencies:
- task:
Cache@2
inputs: key:
'npm | "\$(Agent.OS)" |
package-lock.json'
path: '\$(Build.SourcesDirectory)
/node_modules'
- Run tests selectively based on changes

(pathFilters).

13. How do you manage multiple environments in Azure DevOps?

Scenario: You need different configurations for dev, test, and production environments.

Solution:

- Use variable groups in

Azure DevOps.

- Example environment-based variable group:
variables: - group: 'Production-Config'
- Define **stages** in YAML:
stages: - stage: Dev
jobs: - job: Deploy -
stage: Prod jobs: - job:
Deploy dependsOn: Dev
condition: succeeded()

14. How do you

trigger a pipeline based on external GitHub commits?

Scenario: You want an Azure DevOps pipeline to trigger automatically when a commit is pushed to a GitHub repository.

Solution:

- Configure a service connection to GitHub.
- Use a GitHub trigger in

```
the pipeline: trigger:  
branches: include: -  
main paths: include: -  
src/*
```

15. How do you integrate Azure DevOps with Jira?

Scenario: Your team uses Jira for issue tracking and wants to link it with Azure DevOps.

Solution:

- Install the Azure DevOps for Jira extension.
- Map Jira issue IDs to Azure DevOps work items.
- Use commit messages with Jira issue references (e.g., DEV-123: Fix login bug).

16. How do you perform load

testing in Azure

DevOps?

Scenario: You need to verify application performance under heavy traffic.

Solution:

- Use Azure Load Testing service.

- Example using Apache JMeter: - task:
AzureLoadTest@1

```
inputs:  
serviceConnection:  
'MyAzureLoadTest'  
testPlanName:  
'PerformanceTest'  
testRunTitle: 'Load  
Test'
```

17. How do you
enforce security
compliance for
Azure DevOps

pipelines?

Scenario: You need to ensure security policies are followed in pipelines.

Solution:

- Use Microsoft Defender for DevOps to scan code.
- Implement OWASP ZAP scanning in pipelines:
 - task: OWASPZAP@1
 - inputs:

```
targetUrl: 'https://my-  
app.com'
```

18. How do you implement feature flags in Azure DevOps?

Scenario: You want to enable/disable new features dynamically without redeploying.

Solution:

- Use Azure App Configuration to store feature flags.
- Example code to check feature flags: if (Configuration["Feature Flags:NewFeature"] == "true") {
 EnableNewFeature(); }

19. How do you deploy a .NET Core application to

Azure App Service?

Scenario: You need an automated deployment pipeline for a .NET Core app.

Solution:

- Use Azure WebApp

Deployment task: -

task: AzureWebApp@1

inputs:

azureSubscription:

'MyAzureServiceConnec

```
tion' appName: 'my-dotnet-app' package: '$(Build.ArtifactStagingDirectory)/app.zip'
```

**20. How do you
rotate secrets
dynamically in
Azure DevOps?
Scenario:** You want to
rotate credentials
periodically for security

reasons.

Solution:

- Store secrets in Azure Key Vault.
- Use Azure DevOps Variable Groups to link Key Vault secrets.

21. How do you set up monitoring and logging in Azure DevOps?

Scenario: You need to track application performance and errors.

Solution:

- Use Application Insights for logs.
- Example YAML:
- script:
 | az monitor app-insights metrics show --app my-app --metric "ServerResponseTime"

22. How do you

automate Azure DevOps agent scaling?

Scenario: Your self-hosted agents should scale dynamically based on workload.

Solution:

- Use Azure Virtual Machine Scale Sets (VMSS) to auto-scale agents.

23. How do you implement A/B testing in Azure DevOps?

Scenario: You want to test different versions of a feature with real users.

Solution:

- Use Azure Front Door to split traffic.
- Example command: az

```
afd route update --  
name my-route --  
distribution 50-50
```

24. How do you secure Docker images in Azure DevOps?

Scenario: You need to
scan container images
for vulnerabilities.

Solution:

- Use Azure Container Registry (ACR) security scanning.
- Example YAML:

```
- task:  
  Docker@2  
  inputs:  
    command: 'scan'  
    imageName:  
      'myimage:latest'
```

25. How do you
create an
automated rollback

pipeline in Kubernetes?

Scenario: A new deployment fails, and you need to revert to the previous version.

Solution:

- Use Helm rollback: -
script: helm rollback
my-app 1

26. How do you

enforce pipeline approvals before production deployment?

Scenario: You want manual approval before deploying to production.

Solution:

- Use manual intervention tasks in YAML.

27. How do you deploy

Infrastructure using

Bicep in Azure

DevOps?

Scenario: You need to provision Azure resources using Bicep.

Solution:

- Use Azure CLI task: -
task: AzureCLI@2

```
inputs: scriptType:  
'bash' scriptLocation:  
'inlineScript'  
inlineScript: 'az  
deployment group  
create --resource-group  
my-rg --template-file  
main.bicep'
```

28. How do you

integrate

ServiceNow with

Azure DevOps?

Scenario: You want automatic ServiceNow ticket creation for failed deployments.

Solution:

- Use ServiceNow DevOps extension in Azure DevOps.

29. How do you ensure high

availability for Azure DevOps services?

Solution:

- Use Geo-replication in Azure DevOps and multi-region deployments.

30. How do you reduce cloud costs in Azure DevOps?

Solution:

- Optimize pipeline runs, use spot VMs, and clean up unused resources.

Let me know if you need deep dives on specific topics!