Azure DevOps Mock Interview Questions & Answers

Q1: What is Azure DevOps?
A: A cloud-based DevOps platform by Microsoft providing CI/CD, version control, project tracking, testing, and artifact management.

Q2: Difference between Azure DevOps and GitHub Actions?
A: Azure DevOps is enterprise-focused with boards, repos, pipelines, and artifacts; GitHub Actions is event-driven CI/CD tightly integrated with GitHub.

Q3: What services are part of Azure DevOps?
A: Azure Boards, Repos, Pipelines, Test Plans, and Artifacts.

Q4: What is the difference between CI and CD?
A: CI ensures code integration and build verification; CD automates deployment to environments.

Q5: How does DevOps differ from traditional SDLC?
A: DevOps emphasizes automation, continuous delivery, collaboration, and faster feedback loops.

✅ 50 Azure DevOps Mock Interview Questions & Answers

Basic Azure & DevOps

1. **Q: What is Azure DevOps?**
   A: A cloud-based DevOps platform by Microsoft providing CI/CD, version control, project tracking, testing, and artifact management.

2. **Q: Difference between Azure DevOps and GitHub Actions?**
   A: Azure DevOps is enterprise-focused with boards, repos, pipelines, and artifacts; GitHub Actions is event-driven CI/CD tightly integrated with GitHub.

3. **Q: What services are part of Azure DevOps?**
   A: Azure Boards, Repos, Pipelines, Test Plans, and Artifacts.

4. **Q: What is the difference between CI and CD?**
   A: CI ensures code integration and build verification; CD automates deployment to environments.

5. **Q: How does DevOps differ from traditional SDLC?**
   A: DevOps emphasizes automation, continuous delivery, collaboration, and faster feedback loops.

Azure Repos / Git

6. **Q: How do you manage branching strategy in Azure Repos?**
   A: Using GitFlow or trunk-based strategy with feature, develop, release, and hotfix branches.

7. **Q: What is a Pull Request in Azure Repos?**
   A: A request to merge changes into a branch after code review and approvals.

8. **Q: How do you enforce branch policies?**
   A: By enabling branch protection with mandatory code reviews, build validation, and work item linking.

9. **Q: Difference between Git clone and Git fork?**
   A: Clone copies repo locally; fork creates a separate copy in remote for independent contributions.

10. **Q: How do you resolve merge conflicts in Azure Repos?**
    A: Manually editing conflicting files and committing resolved versions.

Azure Pipelines (CI/CD)

11. **Q: What are Azure Pipelines?**
    A: CI/CD pipelines that automate build, test, and deployment.

12. **Q: Difference between Classic Pipeline and YAML Pipeline?**
    A: Classic uses GUI designer; YAML is code-based, version-controlled, and portable.

13. **Q: How do you trigger pipelines automatically?**
    A: Using trigger keyword for branch, pr for pull requests, and schedules.

14. **Q: What is an agent in Azure Pipelines?**
    A: A compute resource that runs pipeline jobs (Microsoft-hosted or self-hosted).

15. **Q: How do you pass secrets in pipelines?**
    A: Using Azure Key Vault integration or pipeline variable groups with secret values.

16. **Q: How do you deploy an app to Azure App Service using pipelines?**
    A: By using built-in Azure Web App deployment tasks or az webapp deploy CLI.

17. **Q: How to implement pipeline stages (Dev, QA, Prod)?**
    A: Define stages with approvals and gates for controlled deployments.

18. **Q: How do you handle rollback in Azure Pipelines?**
    A: Deploying a previous build artifact or using deployment slots.

19. **Q: What is a Multi-stage Pipeline?**
    A: A YAML pipeline that contains build, test, and deploy stages in sequence.

20. **Q: How do you use pipeline templates?**
    A: By reusing YAML templates with parameters across projects.

Infrastructure as Code (IaC)

21. **Q: What is ARM template in Azure?**
    A: JSON-based declarative file to deploy Azure resources.

22. **Q: Difference between ARM Templates and Terraform?**
    A: ARM is Azure-specific; Terraform is multi-cloud and has better modularity.

23. **Q: How do you store Terraform state securely in Azure?**
    A: In Azure Storage account with blob locking via Azure Key Vault.

24. **Q: How do you run Terraform in Azure DevOps?**
    A: Using Terraform tasks or CLI with service connection for authentication.

25. **Q: What is Bicep in Azure?**
    A: A domain-specific language (DSL) that simplifies ARM template authoring.

Azure Kubernetes Service (AKS)

26. **Q: How do you deploy apps to AKS using Azure DevOps?**
    A: Using Helm charts or kubectl commands in pipelines.

27. **Q: How do you authenticate Azure DevOps pipeline with AKS?**
    A: Using Azure Service Principal or Managed Identity with kubeconfig.

28. **Q: What is Helm in Kubernetes?**
    A: A package manager for Kubernetes for templated deployments.

29. **Q: How do you implement blue-green deployment in AKS?**
    A: By running two environments and switching traffic via service routing.

30. **Q: How do you implement canary deployment in AKS?**
    A: By gradually shifting traffic using Ingress controller or service mesh.

Monitoring & Logging

31. **Q: How do you monitor pipelines in Azure DevOps?**
    A: Using pipeline analytics and logs from pipeline runs.

32. **Q: How do you monitor applications in Azure?**
    A: With Azure Monitor, Application Insights, and Log Analytics.

33. **Q: What is Application Insights?**
    A: A service that collects application performance, failures, and telemetry.

34. **Q: How do you collect AKS logs?**
    A: Using Azure Monitor Container Insights and Log Analytics workspace.

35. **Q: How do you set alerts for pipeline failures?**
    A: By configuring service hooks or integrating with Teams/Slack.

Security & Governance

36. **Q: How do you manage secrets in Azure DevOps?**
    A: With Azure Key Vault and secret variables in pipelines.

37. **Q: What is a Service Connection in Azure DevOps?**
    A: A secure way to connect pipelines to Azure subscriptions/resources.

38. **Q: How do you secure pipeline agents?**
    A: By using private agents, restricting permissions, and patching OS regularly.

39. **Q: How do you implement RBAC in Azure DevOps?**
    A: By assigning roles at organization, project, and repo levels.

40. **Q: What is DevSecOps in Azure?**
    A: Integrating security scanning (SAST, DAST, dependency scanning) in pipelines.

Real-world Scenarios

41. **Q: How do you handle zero downtime deployments?**
    A: Using deployment slots, rolling updates, or blue-green strategies.

42. **Q: What do you do if a deployment fails in production?**
    A: Rollback using previous build or slot swap; analyze logs for root cause.

43. **Q: How do you handle multi-region deployments?**
    A: Using Traffic Manager or Front Door with geo-distributed pipelines.

44. **Q: How do you scale an AKS cluster automatically?**
    A: Using Cluster Autoscaler and Horizontal Pod Autoscaler.

45. **Q: How do you optimize pipeline performance?**
    A: Use caching, parallel jobs, pipeline templates, and self-hosted agents.

46. **Q: How do you integrate testing in pipelines?**
    A: By adding unit, integration, and automated UI tests as pipeline tasks.

47. **Q: How do you deploy microservices in Azure?**
    A: Using AKS, Azure Service Fabric, or Container Apps.

48. **Q: How do you ensure compliance in pipelines?**
    A: By adding approval gates, quality checks, and compliance policies.

49. **Q: How do you implement blue-green deployment for Azure App Service?**
    A: Using deployment slots and swapping after validation.

50. **Q: What challenges have you faced in Azure DevOps?**
    A: Examples: long build times, pipeline failures due to infra, securing secrets, managing IaC drift.