**1) What is DevOps?**

DevOps is Development and Operation’s Collaboration, it’s a Union of 3Ps – Process, People and Product (working Product) that enable continuous integration and continuous delivery of value to our end users. DevOps accelerate the process to deliver applications and software services at high speed and high velocity. So that organization can learn and adopt to the market at its earliest. Also, it minimizing the risk factor by continuously delivering and getting end users and stakeholders feedback at the early stages.

Learning DevOps can be a rewarding career path that opens doors to exciting opportunities in the tech industry. Here’s a comprehensive guide on how to learn DevOps effectively:

1. **Establish a Strong Foundation in Programming:** Begin by mastering at least one programming language, such as Python, Ruby, or Java. These languages are widely used in DevOps tools and automation scripts.
2. **Understand the Fundamentals of DevOps:** Grasp the core concepts of DevOps, including continuous integration (CI), continuous delivery (CD), infrastructure as code (IaC), and configuration management.
3. **Explore DevOps Tools and Technologies:** Familiarize yourself with essential DevOps tools like Git, Jenkins, Docker, Kubernetes, Terraform, Ansible, and Nagios. Learn how to use these tools for various DevOps tasks.
4. **Gain Hands-on Experience:** Practice DevOps skills by building and deploying applications using DevOps methodologies. Participate in open-source projects, contribute to codebases, and experiment with different DevOps tools.
5. **Pursue Certifications:** Consider pursuing certifications like AWS Certified DevOps Engineer – Professional, Certified Kubernetes Administrator (CKA), or Certified ScrumMaster (CSM) to enhance your credibility and expertise.
6. **Stay Updated with the Latest Trends:** Keep abreast of the latest developments and trends in DevOps. Read industry blogs, attend webinars, and participate in DevOps communities to stay informed.
7. **Network with DevOps Professionals:** Connect with experienced DevOps professionals to seek mentorship, guidance, and insights into the industry. Attend DevOps conferences and meetups to expand your network.

**2) What is the need for DevOps?**

In Traditional software development, after completing the development part, the code deployment time was huge. And many times, we heard the common fights between the Development Team and Operations Team or deployment team that it works fine on our system, it’s the sever causing problem and operation team defenses it’s not your server it’s your code, Right? Well, DevOps solves the Traditional Dev and Ops fights by breaking the wall of confusion.

**3) How DevOps Works?**

DevOps is the practice of operations and development engineers that work together in the entire project lifecycle, from the design and development process to production releases and support.

Starting from design and development to testing automation and from continuous integration to continuous delivery, the team works together to achieve the desired goal. People having both development and operations skill sets working together and use various tools for CI-CD and Monitoring to respond quickly to customer’s need and fix issues and bugs.

**4) What are the benefits of DevOps?**

The main benefits of implementing DevOps are:

* Customer Satisfaction.
* More engaged and Collaborative Development and Operation teams.
* DevOps Deploy code faster in the market through Continuous Integration and Continuous Delivery.
* Faster Operational Support.
* Strong Infrastructure Performance and IT performance.
* Less failures and continuous improvement.
* Transparency between the team.
* Constant Monitoring and better adaption.
* Increase efficiency

**5) What are the DevOps tools?**

To Implement DevOps, Automation plays a major role and we defiantly need some tools for Implementation. Following are the major areas:

* Planning
* Code management
* Build and Testing
* Release management
* Deploy and Monitor

Here’s a list of tools that can help you meet your DevOps requirement perfectly.

**6) Which tools are useful for Continuous Integration?**

Azure Pipelines has support for all the platform like Linux, macOS, and Windows also we can consider following tools for the Continuous Integration.

Jenkins

TeamCity

Travis CI

Bamboo

GitLab CI

CircleCI

Codeship

**7) Which tools are useful for Continuous Deployment?**

Following are few useful Continuous Deployment tools

* Azure Pipelines for Deployment.
* Jenkins.
* TeamCity.
* Bamboo.
* ElectricFlow.
* Octopus Deploy.
* AWS CodeDeploy.
* DeployBot.
* Shippable.

**8) What is InfrastructureConfiguration Which tools are useful for InfrastructureConfiguration?**

In today’s fast and competitive market, many companies demand a faster deployment process and Infrastructure Configuration, so treating Infrastructure as software and manage the processes such as version control, continuous integration, deployment and automated testing will make infrastructure changes more rapidly and reliably.

Following are the most popular tools for Infrastructure Configuration.

* Chef
* Puppet
* Ansible

**9) What is Continuous Testing? What is the use of Test Automation in DevOps?**

DevOps is not about jobs or tools, it’s about people, culture and automation. and to implement DevOps, continuous testing plays a very important role where writing scripts for software testing and make it auto executable so that we can automate the testing and do the frequent releases using the delivery pipelines.

We have to write unit testing to achieve Continuous Testing.

**10) Which tools are useful for Continuous Testing?**

For test Automation there are many open source tools are available, following are few names

* Selenium
* JMeter
* JUnit
* AntUnit
* Cucumber
* SoapUI
* Tricentis Tosca

**11) What is Continuous Monitoring?**

DevOps and Agile are all about inspection and adaption to make continuous improvement in our process, and for that, we must have to monitor continuously the process, application performance and infrastructures.

By doing continuous infatuates monitoring, we can visualize the process and get the early alerts in real time. By analyzing that data, we can take decision wisely and adapt the things and process that best suits for the business.

**12) Which tools are useful for Continuous Monitoring?**

For continuous monitoring Nagios, SysDig and Zabbix are the famous open source tools available in the market. Infrastructure as code(laC) vendors like Amazon and Google have tools like AWS CloudWatch and StackDriver. Also, New Relic is a good option for continuous monitoring.

**13) What is the container and What containers Azure DevOps support?**

The container will provide a way to package your software code, its configurations, Packages and its dependencies into a single unit or object.

We can have multiple containers that can run on the same machine and share the operating system with other containers so that we can run anywhere fast and reliable and consistent deployments.

Azure DevOps has the following container support.

Docker

Asp.Net with containers.

Azure Kubernetes services.

Azure Service Fabric application with Docker support.

**14) What is Azure DevOps? What is the difference between Azure DevOps and VSTS Online?**

Microsoft Visual Studio Team Services, now known as Azure DevOps having excellent application lifecycle management tool.

We can plan a project with Agile tools and templates, manage and run test plans, Version control source code and manage the branches, deploy the solution across all platform using Azure Pipelines, by implementing Continuous Instigation and Continuous Deployment.

**15) What services Azure DevOps Provides?**

Azure DevOps provides full application lifecycle management from planning to coding, and from testing to build and deploy.

**16) List out some famous tools of DevOps ?**

* Jenkin
* Travis CI
* Bamboo
* Hudson
* TeamCity
* CircleCI
* JIRA
* Slack
* Zoom
* Clarizen
* Asana
* SVN
* Git
* Bitbucket
* GitHub

**17) Mention various phases of DevOps ?**

The given below are the different phases of DevOps

Plan: plan is essential for any application which requires the development, preparing plan for the process of development is the best practice.  
Code: The code of application is designed according to the needs of users.  
Build: with the help of different codes that are generated in earlier steps, the build of the application created.  
Test: Test plays an essential role in the development of the It application, which helps to test the application and it re-built the application when required.  
Integrate: It is used to collaborate the various codes, which are from multiple programers.  
Deploy: It helps for the future use by forming the environment of the cloud from the deployed codes, and it maintains continuous functioning even though the new changes take place in the website with a high traffic.  
Operate: When it is necessary operations take place on code.  
Monitor: For the customer needs, the performance of the application is monitored.  
IMAGE

**18) What are the advantages of Azure DevOps ?**

The given below are some important advantages of the Azure DevOps

* It delivers the software continuously.
* It simply solves the problems which are complex.
* It identifies the problems quickly and solves them with high speed.
* It is also used for quick features transport.
* It maintains the secured operating environment.
* It is used to develop the collaboration of the various teams.

**19) Describe about azure DevOps ?**

It is a new version of Microsoft VSTS, it is used for the project planning by using the templates and the tools of Azure. Its functionality helps to manage and run the plans of the test, it is also used for the solution deployment through various platforms to leverage the pipelines. It provides CI and CD quick deployment.

**20) Define Azure boards ?**

It is a DevOps service which is used to manage the projects of software, It offers various sets of abilities like dashboards, reporting, scrum and kanban. Its essential feature contains queries, backlogs, sprints, dashboards and work items.

**21) Describe about Azure repos ?**

It is a system version control which is used for code management and various versions by using the lifecycle of the development. It is also used to track the changes for the code through various teams, the clear list of changes may be used for coordination between teams and collaborate the changes for future use.

One of the essential features of it is centralized version control and distributed version control. Git is an example for distributed version control, TFVC is an example for centralized version control.

**22) Explain Azure pipelines ?**

It is One of the essential azure cloud services, with which we can build and test the projects code automatically. It works productively with the help of major languages and types of the project, it shares that project code with other customers also.

**23) Explain the test plans of Azure ?**

They are Azure DevOps services which offer solutions for test management, it offers the essential abilities, exploratory testing, customer testing, planned testing manually. It maintains an extensive browser, which offers testing exploration along with stakeholders feedback storage. The essential techniques like testing exploration and manuals are required for the product development, they are the reason for testing automation.

**24) Mention some important components of azure devops integration and the bit bucket ?**

Answer for this question represents the relation between hosted agent and the service of external Git, its continuous integration and continuous deployment is congruent through Github and the server of Git such as bit bucket. Instead of transforming the complete Gitlab project, it is used for external respiratory connection to get the advantages of its continuous integration and the continuous development.

**25) Describe the projects of Azure DevOps ?**

Project of the azure devops is an essential choice for gaining experience, which is helped for code existence and Git repository to design the continuous delivery and integration pipelines of Azure DevOps.

**26) Mention the services provided by Azure DevOps ?**

It offers the complete lifecycle management of applications, that is from plan to code, and also from test to build and deployment.

**27) Describe continuous monitoring ?**

For simultaneous development of our process, agile and devops are used by adopting and spectations. We need to monitor the constant progress and the infrastructure, with the monitoring of simultaneous infatuation, we can do process visualization to gain quick alerts in real time. With the help of data analysis we can select and the process which suits our organization.

**28) List Out some useful tools of continuous monitoring ?**

* Lansweeper
* Spiceworks
* Snort
* Solarwinds
* Nagios
* Tenable
* Ipswitch WhatsUp Gold
* Paessler PRTG
* Rapid7 Insight
* Cisco Identity Services Engine

**29) Name the web application which can be deployed with Azure ?**

The given below are the web applications, that can be deployed with Azure

* ASP.NET
* PHP
* WCF

**30) Mention the service is used to manage the Azure resources.**

Azure resource manager is the one for resource management, that contains various services like deployment, management etc.

**31) What features does Azure DevOps provide?**

Azure DevOps provides many integrated features that you can use based on your project requirement. You can access these features directly in the browser or IDE.

* Azure Boards to manage the PBIs and Tasks.
* Azure Pipelines for the automation of build pipelines and releases.
* Azure Repos allows you to manage your code versions.
* Azure Test Plans to manage the Test artifacts.
* Azure Artifacts manages the public and private package feeds.
* Azure OnePlan provides the capability for enhanced visibility and alignment of the team working on the project.

**32) What is Azure DevOps Services?**

Azure DevOps provides different services:

* Azure Boards
* Azure Pipelines
* Azure Repos
* Azure Test Plans
* Azure Artifacts
* Extensions Marketplace

**33) What are Azure DevOps Boards?**

Azure Boards is a service to manage software projects. You can easily track features, stories, tasks and bugs associated with your projects. It provides capabilities including reporting, dashboards, support for Scrum and Kanban. You can plan, discuss and track work across the teams. For more visit Azure Boards

**34) What is the Azure DevOps Pipeline?**

Azure DevOps Pipeline performs the build and testing of any type of projects with any programming language to make them available to other targets. Azure Pipeline is an automated combination of Continuous Integration and Continuous Delivery to build, test the code project and ship it to deployment target.

**35) What different ways does Azure DevOps offer to define pipelines?**

You should have a basic understanding of key concepts such as triggers, stages, tasks, task group, variables, variable groups, build artifacts to create a release pipeline. Azure DevOps offers two ways to define pipelines:

Using YAML Syntax – Here you can use a YAML file to create your pipeline that you can version with your rest of code. The YAML file ‘azure-pipeline.yml’ contains all the code changes to build and deploy your infrastructure and application. You can also use YAML Editor in Azure DevOps portal to create your pipeline.  
Using Classic Interface – Azure DevOps offers the capability to create releases using highly configurable and manageable pipelines by Classic User Interface. You can create releases to multiple stages with required approvals and checks using these pipelines. For more visit Define Classic Pipeline.

**36) What is a Multi-stage Pipeline?**

Multi-stage Pipeline allows you to define CI, CD or both in your single YAML pipeline code. It offers some benefits such as easy to maintain, you can easily check that in which stage your deployment is currently in, and many more. For more about Multi-stage pipeline visit Multi-stage Pipeline Azure DevOps.

**37) Why should you use Azure DevOps Pipelines?**

Azure pipelines comes with many benefits:

Provides ability to work with any language or platform  
Provides integration with GitHub  
Can configure Open-Source Projects  
Has ability to deploy on different type of targets at same time  
Has integration with Azure Deployments  
Provide build on Linux, Windows and Mac Machines

**38) What is Release in Azure DevOps?**

Releases or Release Pipelines allow you to continuously deliver your software to customers at a faster rate with lower risk. Release pipelines provide capability to fully automate the testing and delivery of your software in multiple stages prior to production. You can control these automated ways by approvals or on-demand deployments.  
Azure DevOps allows you to create release pipelines by navigating to Releases in Pipeline Menu. Once you click on ‘New’ then ‘New Release Pipeline’, It will navigate you to the new release pipeline page where you can specify Build Artifacts, Stages, Pre-deployment conditions, Variables and other Deployment options.

**39) Explain variable and variable groups in Azure DevOps.**

Variables allow you to store some data that can be used across pipelines. All variables are mutable and stored as strings.  
Variable Groups provide the ability to use variables across multiple pipelines. You can store secrets in variable or variable groups.

**40) What are Azure DevOps artifacts?**

Azure Artifacts allows you to create and share npm, NuGet and Maven package feeds from private and public sources with a team. Azure artifacts has capability to manage all types of packages like npm, NuGet, Maven etc.  
You can easily add fully integrated Package management to your Continuous Integration/Continuous Delivery CI/CD pipelines with a single click or via ARM functionality. For more about Azure Artifacts visit Azure Artifacts.

**41) What are Azure DevOps Test Plans?**

Azure Test Plans is a planned and exploratory testing solution to improve code quality and provides three types of test management artifacts -test plans, test suites and test cases in Azure DevOps server. For more about Azure Test Plans you can refer Azure Test Plans.

**42) What are Azure DevOps Repos?**

Azure Repos is a set of version control tools by Azure DevOps. Azure repos are used to manage the code. It provides an unlimited number of private repositories, pull requests and code search for your projects. You can create a branch as per your requirement and can push the code from any IDE, GIT client or editor. Azure repos support two types of version control as below.

Git – is a distributed version control system and most widely used.  
TFVC – Team Foundation Version Control – a centralized version control

**43) What is Pull Request in Azure DevOps Repos?**

Pull requests are used to review and merge code to a Git project. Pull requests allow your team to review your code and provide feedback on your code. Pull requests can come from either feature or topic branches in the same repository or from a fork.

**44) What is Forks?**

Forks provide a way to isolate confidential, risky and experimental changes from an original codebase. A fork is a complete new copy of code repository including branches, commits and files. A fork looks like someone cloned a repository and pushed to a new, empty repository. Once a Fork is created, you can not share fork files, branches with the original codebase unless pull request carries them along.

**45) What are the major areas of DevOps tools?**

Candidates could face this simple question, among other common Azure DevOps interview questions. The answer implies that automation plays a major role in the implementation of DevOps. Therefore, DevOps tools are highly dominant in the areas of planning, code management, building and testing, and release management. In addition, DevOps tools also have functionalities in deployment and monitoring tasks in the DevOps ecosystem.

**46) What are the popular DevOps tools for continuous integration and continuous deployment?**

The notable DevOps tools for continuous integration include Jenkins, GitLab CI, TeamCity, Bamboo, Codeship, CircleCI, and Travis CI. The popular DevOps tools for continuous deployment include Azure Pipelines for Deployment, Jenkins, Bamboo, DeployBot, Shippable, ElectricFlow, and TeamCity.

**47) What is continuous testing and the ideal DevOps tools for the same?**

Candidates could expect to face this entry in frequently-asked Azure DevOps interview questions. First of all, you need to understand that DevOps is not about tools or process improvements. DevOps focuses on people, automation, and culture changes. Therefore, automated testing through writing scripts to execute the testing process automatically enables frequent releases. Many opensource tools for test automation can help in achieving the DevOps objective of continuous testing. Some of the notable DevOps tools for continuous testing are Selenium, JMeter, AntUnit, JUnit, SoapUI, and Cucumber.

**48) What is the difference between Azure DevOps Services and Azure DevOps Server?**

Candidates would generally face this entry as one of the tricky Azure DevOps interview questions. Azure DevOps Services is the cloud service of Microsoft Azure with a highly scalable, reliable, and globally available hosted service. On the other hand, DevOps Server is an on-premises offering, built on a SQL Server back end.

Enterprises choose the on-premises option when they need their day within their network. Another scenario for choosing on-premises involves the need for accessing SQL Server reporting services integrating effectively with Azure DevOps data and tools. Both Azure DevOps Services and Azure DevOps Server offer similar basic services, albeit with certain added benefits of the former. Here are the additional advantages of Azure DevOps Services.

* Simpler server management.
* Better connectivity with remote sites.
* Faster access to new and productive features.
* Transition in focus from capital expenditures on servers and infrastructure towards operational expenditures on subscriptions.

**49) Which factors should I consider for choosing one from Azure DevOps Services and Azure DevOps Server?**

Candidates could find this entry as one of the advanced Azure DevOps interview questions. Most important of all, you can get follow-up questions regarding each factor in response to this question. The important factors to consider before making the choice of a platform between Azure DevOps Services and Azure DevOps Server are:

* Scope and scale data
* Authentication requirements
* Users and groups
* Management of user access
* Security and data protection precedents
* Process customization
* Reporting

**50) What are the different DevOps solution architectures?**

You can leverage multiple tools and technologies with Azure for the following DevOps scenarios to design solution architectures.

* CI/CD for Containers
* Java CI/CD using Jenkins and Azure Web Apps
* Container CI/CD using Jenkins and Kubernetes on Azure Kubernetes Service
* Immutable Infrastructure CI/CD using Jenkins and Terraform on Azure Virtual Architecture
* DevTest image factory
* CI/CD for Azure VMs
* CI/CD for Azure Web Apps

**51) What are Azure boards?**

Azure Boards is an Azure DevOps service that helps in the management of work in software projects. Azure Boards provide a diverse set of capabilities such as customizable dashboards, integrated reporting, and native support for Kanban and Scrum. The core features of Azure Boards include work items, boards, backlogs, sprints, dashboards, and queries.

**52) How does DevOps work?**

DevOps is a continuous process that involves the following steps:

1. **Plan:** Define the requirements for the software
2. **Develop:** Create the software code
3. **Build:** Compile the software code into a deployable artifact
4. **Test:** Test the software for quality and functionality
5. **Release:** Deploy the software to production
6. **Monitor:** Monitor the software for performance and availability

**53) What is Azure DevOps?**

Azure DevOps is a cloud-based platform that provides a set of tools for DevOps practices. It includes tools for planning, development, testing, release, and monitoring.

**54) What are the benefits of Azure DevOps?**

Azure DevOps provides a number of benefits, including:

* **Reduced time to market:** Azure DevOps can help you to release software faster and more frequently.
* **Improved software quality:** Azure DevOps can help you to improve the quality of your software by providing tools for automated testing and continuous integration.
* **Reduced downtime:** Azure DevOps can help you to reduce downtime by providing tools for monitoring and alerting.
* **Improved collaboration:** Azure DevOps can help you to improve collaboration between development and operations teams by providing a shared platform for work.

**55) What are the key components of Azure DevOps?**

The key components of Azure DevOps are:

* **Azure DevOps Pipelines:** A tool for automating the software development lifecycle
* **Azure Repos:** A tool for storing and managing source code
* **Azure Boards:** A tool for tracking work items and managing projects
* **Azure Artifacts:** A tool for storing and managing build artifacts
* **Azure Test Plans:** A tool for creating and managing test plans and test suites
* **Azure Releases:** A tool for managing software releases

**56) What are the different deployment approaches in Azure DevOps?**

The different deployment approaches in Azure DevOps are:

* **Manual deployment:** This is the most basic deployment approach, and it involves manually deploying the software to production.
* **Blue-green deployment:** This approach involves deploying the new version of the software to a staging environment, and then switching traffic to the new version if there are no issues.
* **Canary deployment:** This approach involves deploying the new version of the software to a small subset of users, and then gradually rolling it out to the rest of the user base if there are no issues.

**57) How do you handle rollbacks in Azure DevOps?**

Rollbacks in Azure DevOps can be handled using the following methods:

* **Manual rollback:** This involves manually rolling back the software to the previous version.
* **Automated rollback:** This involves using a tool to automate the rollback process.

**58) How do you monitor the health of your application in Azure DevOps?**

The health of your application can be monitored in Azure DevOps using the following methods:

* **Application Insights:** A tool for collecting and analyzing telemetry data from your application.
* **Azure Monitor:** A tool for monitoring the health and performance of your Azure resources.

**59) How do you secure your Azure DevOps environment?**

Your Azure DevOps environment can be secured using the following methods:

* **Role-based access control (RBAC):** This allows you to control who has access to what resources in Azure DevOps.
* **Network security:** This involves using firewalls and other security controls to protect your Azure DevOps environment from unauthorized access.

**60) What are some of the best practices for using Azure DevOps?**

Some of the best practices for using Azure DevOps are:

* **Use pipelines for all deployments:** This will help you to automate the software development lifecycle and reduce the risk of errors.
* **Use version control for all source code:** This will help you to track changes to your code and revert to previous versions if necessary.
* **Use automated testing:** This will help you to improve the quality of your software by identifying and fixing bugs early in the development process.
* **Use monitoring:** This will help you to identify and resolve issues with your application before they cause downtime or other problems.