### ****1. What is DevOps?****

Of course, this is one of the first Microsoft Azure DevOps interview questions that you will be asked. The full form of **[DevOps](https://intellipaat.com/blog/what-is-devops/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank)** is Development and Operation’s Collaboration. It focuses on the 3Ps—Process, People, and (working) Product that stands for continuous integration and continuous delivery of value to end-users.

In simple terms, DevOps, essentially, speeds up the process of delivery of applications and software services. This ability to continuously deliver minimizes the risk factor. This is possible through stakeholder and end-user feedback collection.

### ****2. Why use DevOps?****

This is a common Azure DevOps interview question. Traditional software development always had a slow code deployment time after completion of development. And oftentimes, the Development Team and Operations Team or deployment team would get into arguments regarding its status blaming the server or the code for the issues. This is where DevOps enters with a solution.

DevOps facilitates the delivery of smaller features to clients in a quick and efficient manner and allows seamless software delivery.

### ****3. How does DevOps work?****

DevOps is the process of Operations and Development Engineers who work in collaboration in a complete project lifecycle, from the design and development to product releases and support.

### ****4. What is Azure DevOps?****

The process includes testing automation, continuous integration, as well as continuous delivery.  
People with both development and operations skills work together and implement various tools for CI-CD and monitoring for quick response to customer’s requirements and fix issues and bugs.

[Azure DevOps](https://intellipaat.com/blog/what-is-azure-devops/) is the new name for Microsoft VSTS (Visual Studio Team Services) and an application lifecycle management tool. It helps in project planning through Agile tools and templates. Other functionalities include version control of source code, management and execution of test plans, and management of branches.

Additionally, Azure DevOps also helps in solution deployments across various platforms with the help of Azure Pipelines and allows continuous integration and continuous deployment.

### ****5. What are the benefits of DevOps?****

The major **[benefits of DevOps](https://intellipaat.com/blog/benefits-of-devops/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank)** are:

* Customer satisfaction
* More engagement and collaboration between Development and Operation teams
* Faster code deployment in the market through continuous integration and delivery
* Faster operational support
* Higher efficiency
* Strong infrastructure and IT performance
* Continuous improvements and reduced failures
* Transparency between teams/li>
* Constant monitoring and better adaption

窗体底端

### ****6. Name a few DevOps tools.****

Some popular [DevOps tools](https://intellipaat.com/blog/top-devops-tools/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) are:

* Git
* Selenium
* Jenkins
* Puppet
* Chef
* Ansible
* Nagios
* Docker

****Keep this**[GIT Cheat Sheet](https://intellipaat.com/blog/tutorial/devops-tutorial/git-cheat-sheet/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank)**handy while studying GIT.****

### ****7. What are the popular DevOps tools for Continuous Integration and Continuous Deployment?****

Azure Pipelines support macOS, Windows, and Linux. Following are a few popular tools for Continuous Integration:

* Jenkins
* TeamCity
* Codeship
* GitLab CI
* CircleCI
* Travis CI
* Bamboo

Following are some of the popular Continuous Deployment tools:

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

### ****8. What are Azure Boards?****

Azure Boards is an Azure DevOps service. It helps manage the work in software projects and provides a wide range of functionalities like native support for Kanban and Scrum, customizable dashboards, and integrated reporting. Azure Boards include features like boards, sprints, work items, dashboards, backlogs, queries, etc.

### ****9. What is Azure Repos?****

This is a basic Azure DevOps interview question but it can be difficult to answer. Azure Repos is a version control system that can manage the different versions of a code and the code itself throughout the lifecycle of development. It is easy to track any changes in the code made by different teams. It also keeps a detailed record of these changes and the history for better coordination within the team. The changes are then merged at a later stage.

Azure Repos offers a centralized version control system (Team Foundation Version Control) as well as a distributed version control system (Git).

### ****10. What are containers?****

Containers provide the means to package software code, its configurations, dependencies, and packages into a single unit or object. Multiple containers can run on the same machine and share OS with other containers for running fast, reliable, and consistent deployments anywhere.

### ****11. What containers does Azure DevOps support?****

Azure DevOps provides the following container support:

* [Docker](https://intellipaat.com/blog/tutorial/devops-tutorial/docker-tutorial/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank)
* [Azure Kubernetes](https://intellipaat.com/blog/what-is-azure-kubernetes-service/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) services
* Asp.Net with containers
* [Azure Service Fabric](https://intellipaat.com/blog/what-is-azure-service-fabric/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) application with Docker support

### ****12. What are Azure Pipelines?****

Azure [DevOps Pipeline](https://intellipaat.com/blog/devops-pipeline/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) automatically develops and tests code projects. It is a service on the Azure cloud that works well with most project types and languages. This service helps in improving the availability of code projects to other users.

### ****13. What is the use of Selenium in DevOps?****

Selenium is used for continuous testing in DevOps. It specializes in forms of regression and functional testing.

### ****14. What are Azure Test Plans?****

This is another one of the popular interview questions on Azure DevOps.  
Azure Test Plans is a service provided by Azure DevOps. It provides a browser-based test management solution along with crucial capabilities in user acceptance testing, exploratory testing, and planned manual testing. It also includes a browser extension for the provision of exploratory testing as well as feedback from stakeholders.

Exploratory and manual testing are critical methods that consist of the evaluation of product or service quality.

Azure Test Plans, additionally, focus on DevOps on automated testing. It combines the contributions from developers, managers, testers, product owners, and user experience advocates and enhances the quality of a project.

### ****15. What are some important features of Memcached?****

Memcached offers a wide variety of features like:

* CAS Tokens
* Callbacks
* GetDelayed
* Binary protocol
* Igbinary

### ****16. What is the Dogpile effect and how can you prevent it?****

Dogpile effect or cache stampede indicates the expiry of cache, followed by the website being simultaneously hit by numerous requests. Semaphore lock helps prevent this effect by generating a new value as the cache expires.

### ****17. What is Continuous Testing? What is the use of Test Automation in DevOps?****

DevOps is all about people, culture, and automation. [Continuous testing](https://intellipaat.com/blog/what-is-continuous-testing/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) plays a crucial role in DevOps. Scripts are written for software testing and made auto executable so that testing can be automated and frequent releases are possible using the delivery pipelines.

### ****18. What is Forking Workflow?****

Forking Workflow gives Developers the service-side repositories. It supports open-source projects and is often used together with a Git hosting service like Bitbucket.

### ****19. What are some of the useful plugins in Jenkins.****

Following are a few useful [Jenkins](https://intellipaat.com/blog/tutorial/devops-tutorial/jenkins-tutorial/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) plugins:

* Amazon EC2
* Maven 2 project
* HTML publisher
* Join
* Copy artifact
* Green Balls

### ****20. Can we move or copy Jenkins from one server to another?****

Yes, it is possible. Jenkins can be moved or copied from one server to another. When copying Jenkins, it is possible to move the jobs directory from the older server to the new or the current one. This enables moving an installation by copying in the corresponding job directory.

### ****21. What are Azure DevOps Projects?****

Azure DevOps Project is a simplified way to effectively bring existing code and Git repository for the creation of CI and CD pipeline to Azure.

Let’s move on to the next section of advanced Azure DevOps Interview Questions for experienced professionals.

## **Advanced Azure DevOps Interview Questions**

### ****22. What is the difference between Azure DevOps Services and Azure DevOps Server?****

This is one of the trickier Azure DevOps interview questions. Azure DevOps Services is the cloud service of [Microsoft Azure](https://intellipaat.com/blog/what-is-microsoft-azure/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) and it offers a highly scalable and reliable hosted service that is globally available. DevOps Server, on the other hand, is an on-premise offering that is on a SQL Server back end.

Enterprises go for the on-premise offering when they want their day within their network or when there is a requirement for accessing SQL Server reporting services that are integrated with Azure DevOps data and tools.

****Intellipaat provides a**[SQL Server Course](https://intellipaat.com/microsoft-sql-server-certification-training/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank)**, a great learning opportunity for its learners.****

### ****23. What are the different DevOps solution architectures?****

Multiple tools and technologies can be leveraged with Azure to design solution architectures for the below DevOps scenarios:

* CI/CD for Azure VMs
* CI/CD for Azure Web Apps
* CI/CD for containers
* DevTest image factory
* Using Azure Web Apps and Jenkins for Java CI/CD
* Using Jenkins and Terraform on Azure Virtual Architecture for immutable infrastructure CI/CD
* Using Jenkins and Kubernetes on Azure Kubernetes Service for Container CI/CD

### ****24. What are the reasons to use CI and CD and Azure pipelines?****

Make sure to go through this question as it is one of the important interview questions**.**Implementing [CI and CD](https://intellipaat.com/blog/what-is-ci-cd/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) pipelines ensure quality and reliable code.

[Azure pipelines](https://intellipaat.com/blog/azure-devops-pipelines/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank) are there to ensure a secure, easy, and quick way to automate project development processes and availability. They are entirely free for use in public projects and cost-effective for private ones (free 30 hours per month).

Below are a few reasons for using CI/CD and Azure pipelines:

* Supports any platform or language
* Enables work with open-source project>
* Building on Windows, Mac, and Linux machines
* Enables simultaneous deployment to various types of a target
* Integration with GitHub and Azure deployments

### ****25. What should you do to make a NuGet package available to anonymous users outside your organization alongside minimizing the number of publication points?****

The answer is the introduction of a new feed for the package. Packages that are hosted in Azure Artifacts, find storage in a feed. Sharing packages with higher scalability and according to requirements can be ensured by setting up permissions on the feed. These multiple feeds enable control of access to packages across four levels. These four levels of access are:

* Owners
* Readers
* Contributors
* Collaborators

### ****26. How can you enable communication between members of the development team working in different locations around the world using Azure DevOps?****

For such an application, the isolation of members of different project teams into different communication channels is the first criterion. Additionally, maintaining communication history in the concerned channels is necessary. Effective integration with Azure DevOps is also important for the application and allows one to add external suppliers and contractors to projects. The Microsoft teams have addressed these requirements through their offering of the right capabilities.

Classifying teams allows the creation of different channels by users for the organization of communications according to the topic. Every channel can include a few to thousands of users.

Microsoft Teams provides a guest access feature that allows the invitation of external people to join internal channels for the purpose of meetings,  file sharing, and messaging. It helps in B2B project management and can also directly integrate with Azure DevOps.

### ****27. Which feature can be used for the development of a multi-tier application using Azure App Service web apps as the front end and Azure SQL database as the back end?****

The appropriate option, in this case, is Application Map in Azure Application Insights. This is because it helps in identifying the performance bottlenecks as well as the failure hotspots in different components of the multi-tier applications.

Application components and the related dependencies are represented by the nodes on the map. Furthermore, it is also capable of providing the status for health KPI and alerts.

Become a master of Azure by going through this online [Microsoft Azure Training](https://intellipaat.com/microsoft-azure-training/" \t "https://intellipaat.com/blog/interview-question/azure-devops-interview-questions/_blank)

### ****28. What can you do to improve the quality of code if there are many unused variables and empty catch blocks?****

Selecting “Run PMD” in a Maven build task will do the trick. PMD is a source code analyzer that is capable of identifying common programming errors like unnecessary object creation, unused variables, and empty code blocks.

An Apache Maven PMD Plugin will automatically run the tool on a project’s source code and the detailed code error results are provided on the site report.

### ****29. What are the necessary components for the integration of Azure DevOps and Bitbucket?****

The integration of Azure DevOps and Bitbuckets requires a self-hosted agent and an external Git service connection. Since GitLab CI/CD is compatible with GitHub and Bitbucket. Instead of moving an entire project to GitLab, it is possible to connect the external repository. One can make use of GitLab CI/CD in this manner.

### ****30. Explain Pair Programming with reference to DevOps.****

Pair programming is an engineering practice of Extreme Programming Rules. It involves two programmers working on the same system and on the same design/code/algorithm.

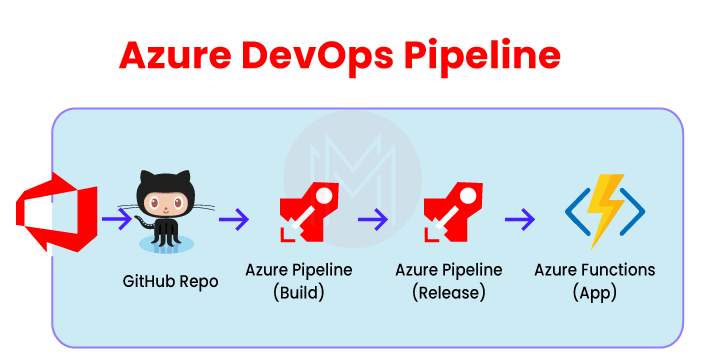
One programmer can be the driver and the other as the observer and constantly monitor the progress of a project and identify problems. The roles can be removed at any moment without prior intimation.

### **1. What do you mean by Azure boards?**

This Azure DevOps service is used for managing the project of software. The various abilities offered by it include reporting, dashboards, kanban, and scrums. Backlogs, queries, dashboards, sprints, and work items are the essential features of Azure boards.

### **2. What do you mean by Azure pipelines?**

It is considered one of the most important cloud services. With the help of Azure pipelines, the project codes can be built as well as tested automatically. With the aid of major languages and the kind of projects, it does its work productively. The project code is shared with other customers as well by this feature.

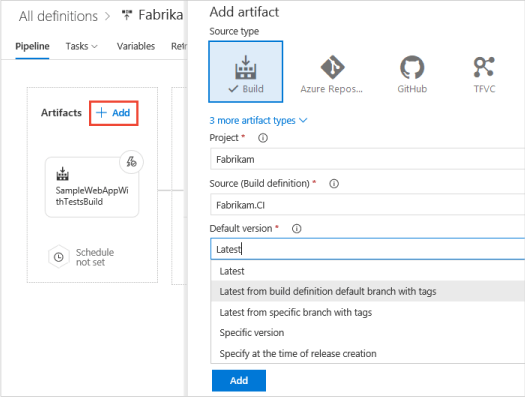


### **3. What do you mean by Azure Repos?**

This system version control is used for managing code and various versions by the use of the lifecycle of the development. It can also track the multiple changes in the code via different teams. The list of changes then can be used for coordination and collaboration between the teams in the future. Centralized version control, as well as distributed version control, is its essential features. An example of distributed version control can be Git. And an example of centralized version control is TFVC.

### **4. What do you mean by Azure artifacts?**

Azure artifact is an extension making it easy to install, discover, and publish the packages such as npm, NuGet, and Maven in [Azure DevOps](https://mindmajix.com/introduction-to-azure-devops" \o "Introduction To Azure DevOps" \t "https://mindmajix.com/_blank). The package management becomes a smooth process in the existing workflows because of its deep integration with other hubs such as Build.



### **5. What do you mean by Azure test plans?**

This Azure DevOps service is used for managing all the test solutions. Exploratory testing, manual planned testing, and customer testing are all provided under this one service. Testing exploration, as well as the stakeholders' feedback storage, is maintained by its extensive browser. Testing explorations and manuals are essential techniques for product development because they are considered the reason for testing automation.

|  |
| --- |
| If you would like to enrich your career in Azure DevOps, then Enrol Our "**[Azure DevOps Training](https://mindmajix.com/azure-devops-training" \t "https://mindmajix.com/_blank)**". This course will help you to achieve excellence in this domain. |

### **6. What do you mean by Multi-stage pipeline?**

CI, CD, or both are allowed by a multi-stage pipeline in a single YAML pipeline code. Being easy to maintain and easy to check the current stage of our deployment is among the many benefits provided by it.

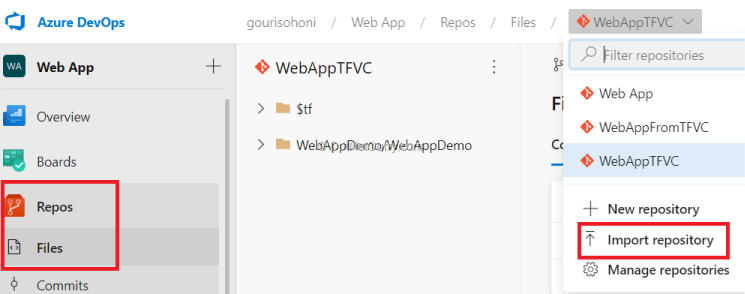
### **7. List some benefits of Azure pipelines.**

Some of the benefits provided by Azure pipelines are-

* Integration with GitHub
* Integration with Azure deployments
* The ability to work with any platform or language
* The ability to deploy and work on multiple targets at the same time
* Configuration of open-source projects
* Build on Windows, Linux, and Mac machines

### **8. What is meant by the pull requests in Azure DevOps Repos?**

Pull requests are used for reviewing as well as merging codes to a Git project. Pull requests enable the team to review the code and also provide feedback on it. Either feature or topic branch, along with a fork, can be the source of pull requests.



### **9. Explain Forks in Azure DevOps.**

We can separate the risky, confidential, and experimental changes from our original codebase with the help of forks. A fork is a whole new copy of code collections that include commits, branches, and files. A fork seems like a repository that has been cloned and pushed to a new and empty repository. The fork files and branches can't be shared with the original codebase once a fork is created. But a pull request can carry them along.

### **10. List some of the features of Memcached.**

A wide range of features are offered by Memcached, such as-

* Callbacks
* CAS tokens
* Igninary
* GetDelayed
* Binary protocol

### **11. Mention why Azure boards should be used?**

Azure boards should be used because of the following reasons-

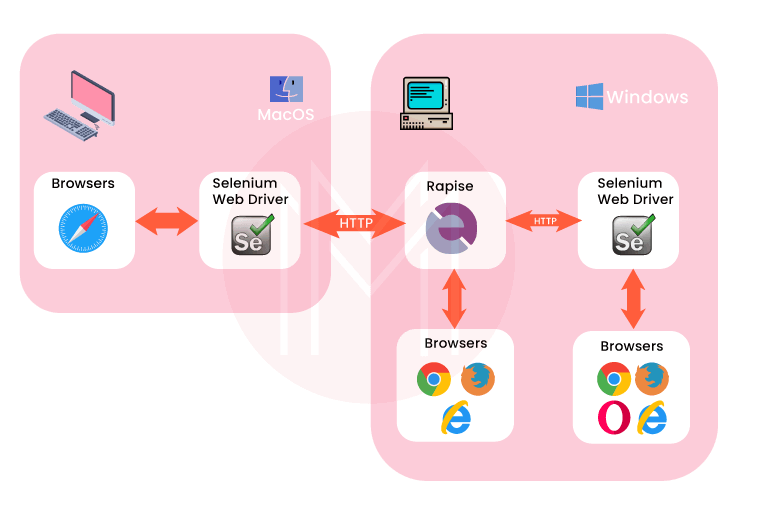
1. Interactive visual tools are provided
2. Scaling makes it easy to begin as levels rise
3. In-built tools enable social communication
4. Extensibility is provided
5. Easy to customize
6. Flexibility and easy storage of collected data
7. Can be begun at no cost
8. Identifying the needs and notifying the changes is easy
9. The progress and status are observed in the dashboard.

### **12. Differentiate between Jira and Azure DevOps.**

Atlassian Jira is also a well-known application life cycle management (ALM) tool. It is especially liked by the agile teams. Originally it was a bug tracking tool. But now, it has grown into a more comprehensive tool that now contains many software development and management tools. The main intention of this tool is to keep things on track. Azure DevOps, on the other hand, not only tracks but extends to building and development as well.

### **13. Why is Selenium used in DevOps?**

The use of Selenium in DevOps is for continuous testing. Its specializations include functional and regression testing.



### **14. What is meant by roles?**

Roles are the type of servers that help us towards common aim achievement. These are managed as well as loaded as virtual machines.  
The three types of roles Azure provides are:

* Worker Role
* Web role
* VM role

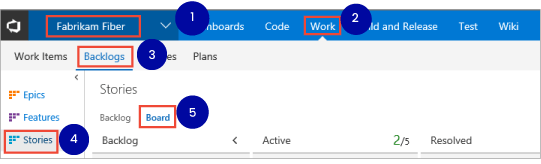
### **15. What different types of backlogs and board options are available in Azure boards?**

Three classes of backlogs are available to manage the work:

* Product
* Portfolio
* Sprint

And two types of boards are available-

* Kanban
* Taskboards



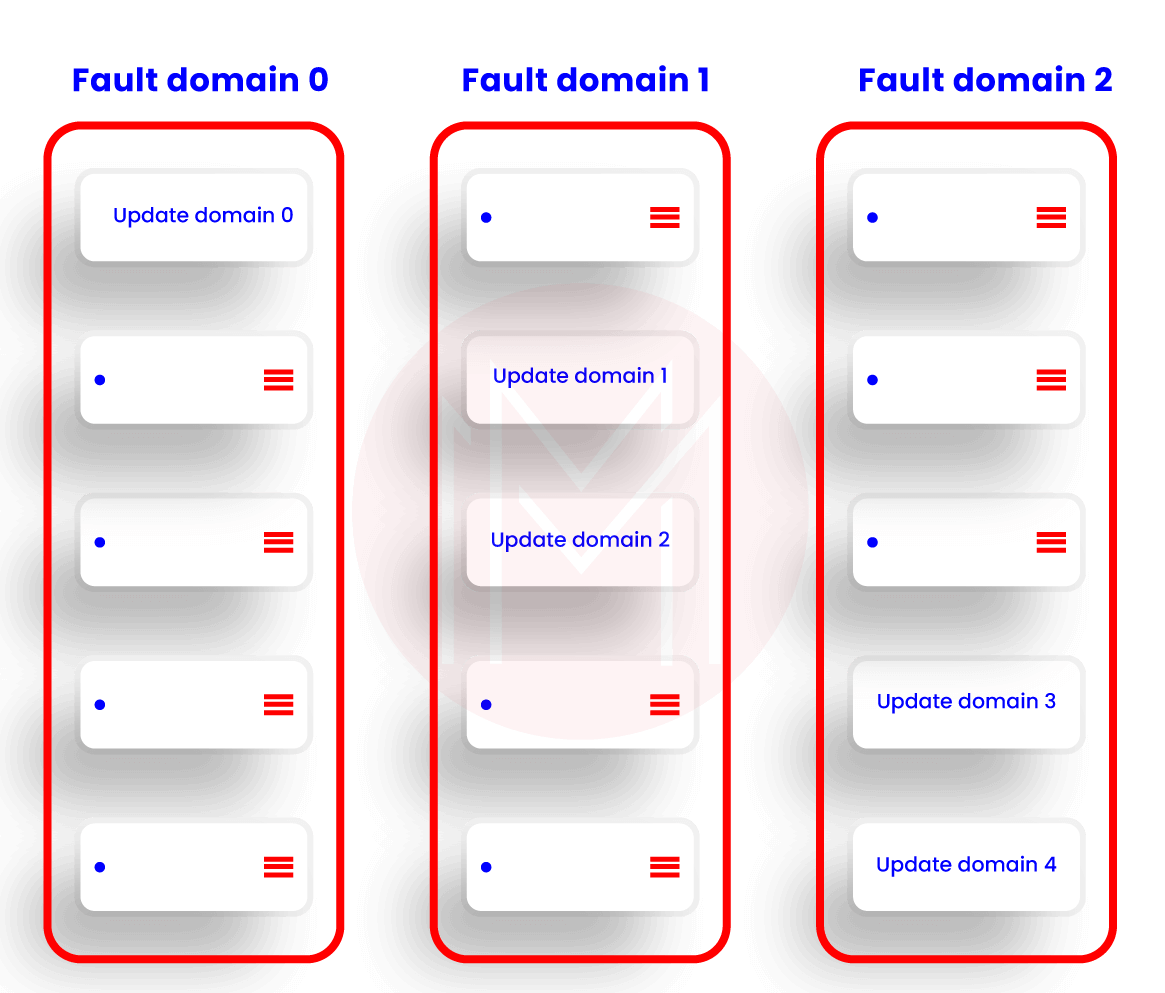
## **Azure DevOps Interview Questions And Answers For Experienced**

### **1. Tell us something about fault domains.**

They are the logical hardware groups that can share the same power source as well as network. They are used for racking the data centers on-premise. These fault domains get automatically distributed by the Azure platform when VMs are designed using available sets. This approach limits network outages, physical hardware failures, etc.

### **2. Tell us something about the Update domains feature.**

They are the logical hardware groups that can be rebooted as well as maintained at the same time. These updated domains get automatically distributed when the VMs are designed using available sets. This ensures our application instance, as well as the running of the Azure platform, is under periodic maintenance. Only a single order of updates can be rebooted at a time.



### **3. How can Azure Ad applications be used?**

A secured and simple way is provided to select and connect with web applications by this feature. The same as the SaaS apps access can also be accessed in the Azure. It doesn't require a VPN to change the infrastructure of the network.

### **4. What is VNet?**

The rendering of our cloud network refers to the VNet. Our launched cloud instances get isolated from the remaining resources with the help of VNet.

### **5. What is meant by system capabilities and user capabilities in the Azure pipeline?**

The self-hosted agent contains a set of capabilities that defines the scope of various capabilities. The name-value pairs are referred to as capabilities. They are called system capabilities when they get discovered automatically by the agent software. On the other hand, they are called user capabilities when they are defined by us. System capabilities such as operating system name, machine name, and other installed software are detected by the agent software.

### **6. What is the azure cli command used to run the pipeline?**

Az pipelines run [-branch]

[-commit-id]

[-folder-path]

[-id]

[-name]

[-open]

[-org]

[-project]

[-variables]

### **7. Elaborate self-hosted agent in Azure pipeline.**

Self-hosted agents allow us to set up as well as management jobs on our own. We opt for self-hosted agents when we want more access to installing dependent software as per our needs and deployments. Also, the speed can be boosted by the persistence of machine-level caches and configuration from run to run.

|  |
| --- |
| Related Articles:****"**[Azure SaaS](https://mindmajix.com/azure-saas" \o "Introduction To Azure SaaS" \t "https://mindmajix.com/_blank)**"**** |

### **8. Elaborate Microsoft-hosted agents in the Azure pipeline.**

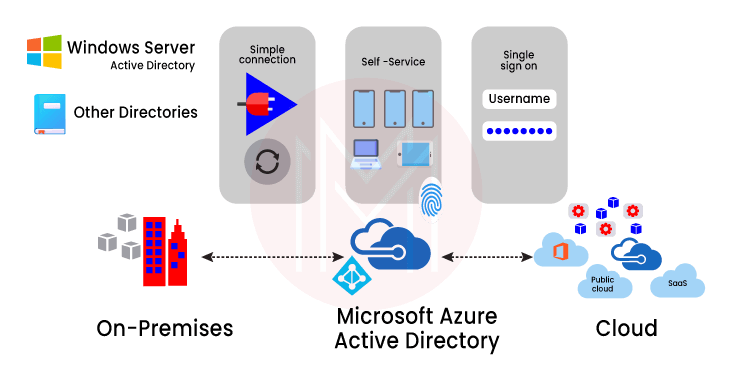
It is the by default Microsoft agents that get the jobs executed. From assigning the resources to installing and maintaining, everything gets done when we opt for this Microsoft-hosted agent.

### **9. What is a Dogpile effect? How can it be prevented?**

* The period after the cache expires and several requests have been made on the website from the client refer to the Dogpile effect.
* The simplest technique to avoid this effect is a semaphore lock. The semaphore lock generates a new value after the cache expires.

### **10. Explain the use of an active directory in Azure.**

It is used for access management. It helps in enabling access to the employees to particular products and services of our networks. Some major examples include Salesforce.com and Twitter. Application-built support in galleries is provided by this active directory.



### **11. Elaborate Azure service fabric.**

This distributed systems platform makes it simple for the package, scalable management, deployment, and microservices reliability. Development and management of the applications of the cloud can also be done by this. Hence the developers can pay more attention to the critical implementation jobs.

### **12. Tell us about the Azure Redis cache.**

It is an open-source data structure. It works as a database and cache. Azure Redis cache is made based on the famous Redis cache of open source. Secured access from azure applications is provided with the help of this feature. The structure of data which are like strings, sets, and hashes along with queries, is maintained by this.

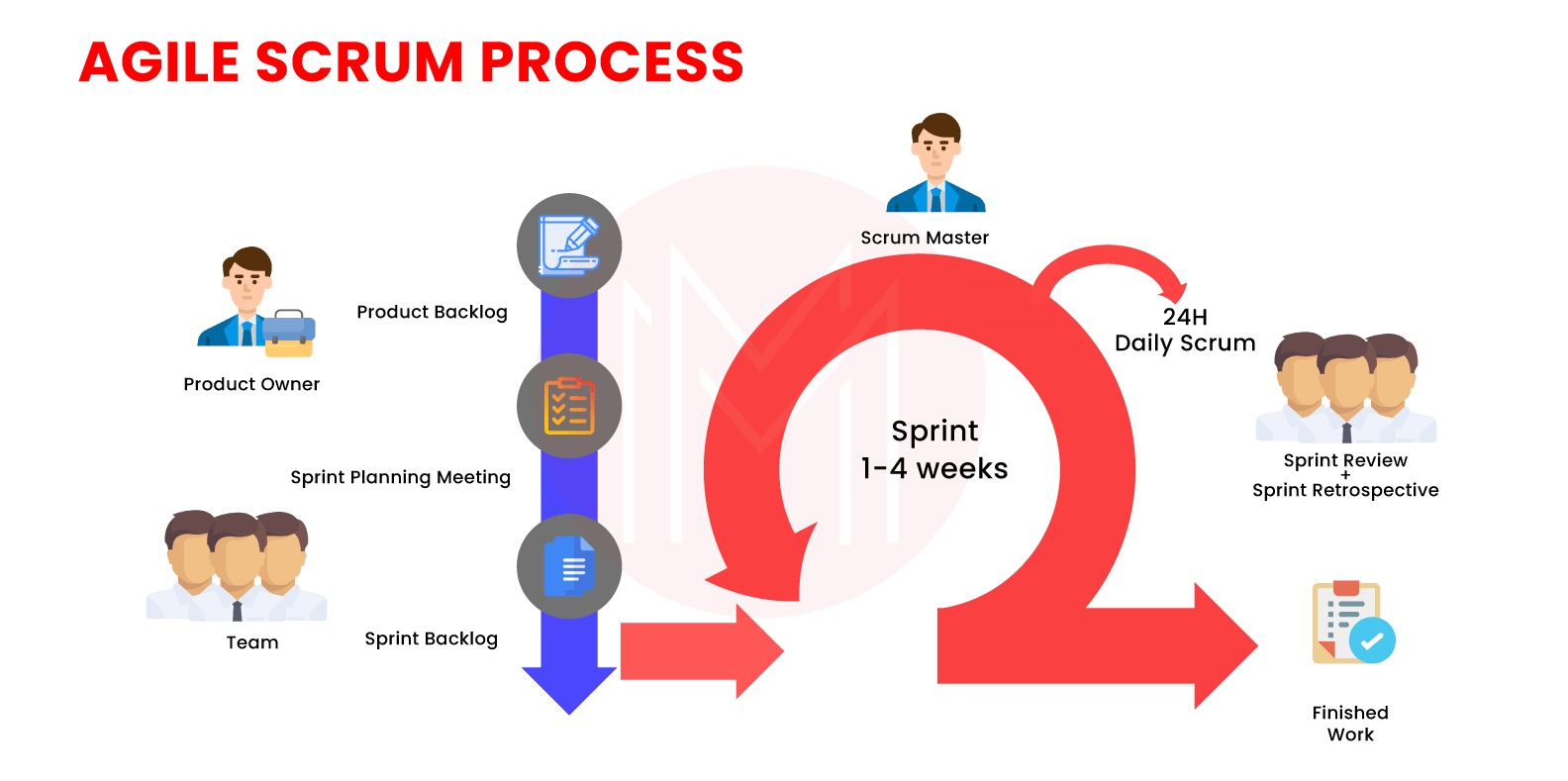
### **13. What is meant by containers? Which containers are supported by azure DevOps?**

To package the software code, its dependencies, and its configurations into a single object or unit, containers are used. Multiple containers can be there working and sharing the operating system on the same machine. This enables fast, consistent, and reliable deployments. The following containers are supported by azure DevOps:

* Asp.Net with containers
* Docker
* Azure service fabric app along with docker support
* Azure Kubernetes services

### **14. Explain the function of Scrum master in azure boards.**

Scrum master is always on the lookout to remove all the hindrances, resolve impediments, and other blocking issues arising during a sprint. They are continually ensuring the improvement of the team. They can be called part team members, part cheerleaders, and part coaches.

********

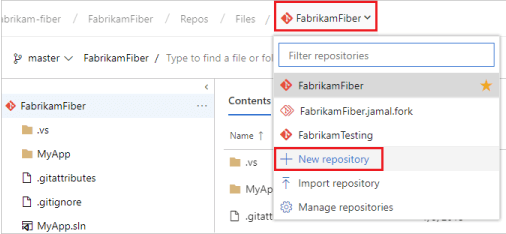
### **15. How can a Git repository be created in Azure DevOps?**

A Git repository can be created by following the given steps:

1. Open DevOps
2. Choose the respective project
3. Click on Repos
4. Click on Files from Repos
5. Click on repositories
6. Click on the "New repository" option from the dropdown menu
7. Choose repository type as Git in the panel appeared.
8. Give the repository a name
9. The "Add a README" checkbox can be unchecked. Otherwise, a readme.txt file will be added.
10. Now, click on create button

(The repository will get selected by default once it is created)

Now, given below are the Top 10 FAQs about Azure DevOps.



## **Frequently Asked Azure DevOps Interview Questions:**

### **1.Mention some of the alternatives to azure DevOps?**

Some of the main alternatives to azure DevOps are GitHub, Jenkins, GitLab, Jira, TeamCity, Bitbucket, Jira, Octopus Deploy, and Amazon Web Services.

### **2. Mention the advantages of using Azure DevOps.**

The main advantages of using azure DevOps are:

1. Higher efficiency
2. Customer satisfaction
3. Speedier code deployment in the market via ceaseless integration and delivery
4. More collaboration between the operation and development teams.
5. Incessant improvements as well as reduced failures
6. Transparency between different teams
7. Speedier operational support
8. Better adaptation
9. Robust infrastructure as well as IT performance

### **3. Which service is used for managing Azure resources?**

Various services, such as deployment, management, etc., are handled by the Azure Resource Manager.

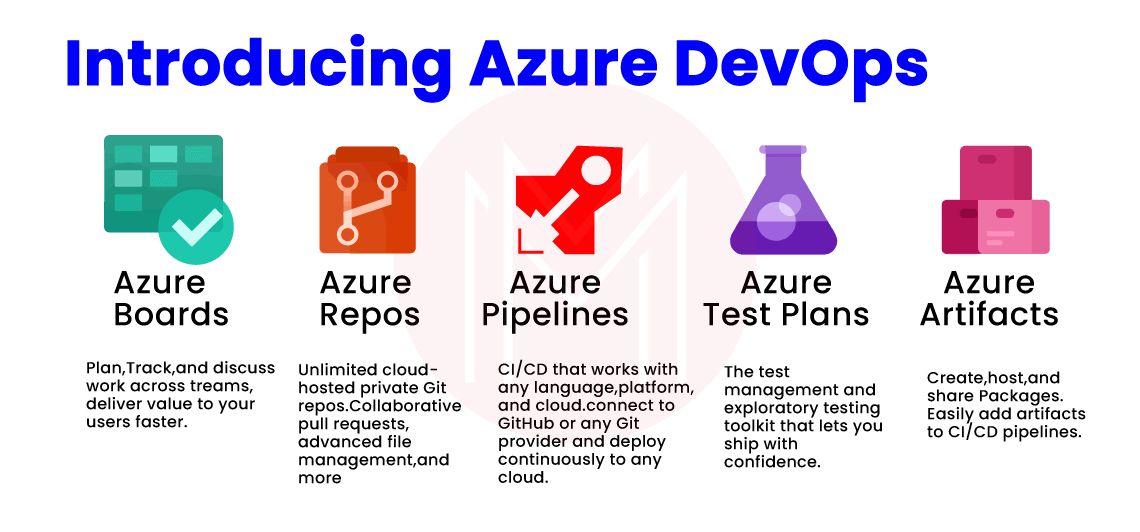
### **4. Differentiate between Azure DevOps server and azure DevOps services.**

Azure DevOps Server is an on-premise platform based on a SQL server. Generally, enterprises opt for an on-premise platform whenever there is a need to access SQL server reporting services integrated with azure DevOps tools and data.  
Azure DevOps services, on the other hand, are the cloud services of Microsoft Azure. It offers highly reliable as well as scalable globally available services.

### **5. What are the main components of Azure DevOps?**

The main components of azure DevOps are given below:

1. Azure Boards- work item tracking, agile planning, reporting, and visualization tool
2. Azure Repos- cloud-hosted private git Repos are provided
3. Azure pipelines- a language and cloud-agnostic CD/CI platform supporting containers or Kubernetes
4. Azure test plans- a planned and integrated testing solution is provided
5. Azure artifacts- integrated package management supporting npm, Maven, NuGet, and Python package feeds from private or public sources.



### **6. Elaborate on the job role of Azure DevOps engineer.**

An Azure DevOps engineer is responsible for designing and implementing a scalable, strong, realistic, and practical cloud solution that would work for the clients.

### **7. Which web applications can be deployed with Azure?**

The web applications given below can be deployed with Azure:

* PHP
* ASP.NET
* WCF

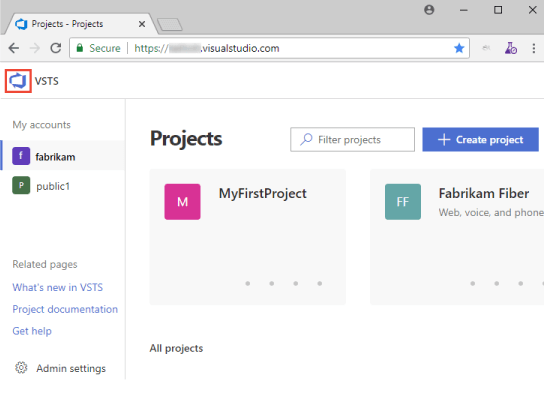
### **8. Mention different ways for connecting with a project in Azure DevOps.**

Following different ways can be followed to connect to a project:

* Integrating with Team Explorer or Visual studio.
* Visual studio code
* Through web portal
* Eclipse/Team explorer
* Intellij with the Azure DevOps services
* Android studio with the Azure DevOps services

### **9. Elaborate on the projects of Azure DevOps.**

To gain experience, the projects of azure DevOps are very important. This helps in the Git repository and code existence for designing the ceaseless delivery as well as integration pipelines of Azure DevOps.



### **10. Briefly explain continuous monitoring.**

DevOps and agile are used simultaneously to develop the process through expectations and adoption. We can constantly monitor the progress, infatuation, and process visualization. And we will be provided with quick alerts in real-time. We can select the process as per our requirement through data analysis. Hence continuous monitoring is of great help.

## 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.

## 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.

## 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.

## 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

## 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.

## 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

## 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.

## 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

## 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.

## 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

## 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.

## 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.

## 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.

## 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.

## What services Azure DevOps Provides?

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

## What is Azure Boards?

Azure board provides service to manage your works, using the Agile Scrum and Kanban templates, Dashboard that we can customize and reporting.

## What are Azure Repos?

Azure Repos is a code version control system that can manage your code and its version.

Using that we can track the changes, whenever team edits code it has all the version history so later, we can coordinate with the team and merge the changes.

The azure repo has both a centralized version control system as well as a distributed version control system.

Git: Distributed Version Control System

Team FoundationVersion Control (TFVC): Centralized Version Control System.

## What are Azure Pipelines?

Azure Pipelines has all the features that are required for supporting Continuous Integration (CI) and Continuous Deployment (CD).

Using that we can constantly test and build the code and release it to any target.

## What are Azure Test Plans?

Azure test plan provides browser-based test management using that we can manage all the testing like Exploratory & manual testing, Continuous testing, Unit & functional testing also we can ask or Request stakeholder to provide feedback.

## What is the significance of having configuration management in DevOps?

Configuration management (CM) assists the team to streamline various tasks and accomplish them in less time. In this way, it enhances the organization’s agility and performance. Also, it assists in providing consistency and enhancing the product development process by implementing means of design streamlining, control, extensive documentation, and change implementation during different phases of the project. You can be aware of configuration management when you go through the [devops tutorial](https://www.dotnettricks.com/learn/devops" \t "https://www.dotnettricks.com/learn/devops/_blank).

## Describe the flow of the DevOps pipeline?

Generally, a pipeline is a set of automated processes that are defined and being followed by the software engineering team. Essentially, DevOps pipeline permits the DevOps software developers and DevOps engineers to competently and consistently compile, create and deploy the software code into production environments.

The corresponding flow is as below:

Developer focus on accomplishing functionality.

The developer sets up his code for the test environment.

Testers work on authenticating the feature. The business team can arbitrate and give feedback.

Developers focus on the test as well as business feedback in a constant collaboration manner.

The code is finally handed over the production and authenticated again

## What are the key differences between DevOps and Agile?

When preparing for DevOps interviw Questions answer, this is one of the most typical questions a candidate needs to be aware of. There are several similarities between DevOps and Agile methodology. But when it comes to software development, both are fundamentally unique approaches. Here are the differences between the two:

Agile prioritizes timeliness but DevOps provides identical priority to quality and timeliness.

DevOps presents smaller release cycles with instant feedback whereas Agile presents only smaller release cycles without instant feedback.

Agile depends on feedback from customers whereas DevOps depends on feedback from its monitoring tools.

The scope of work is agility only for Agile whereas it is agility and need for automation for DevOps.

## How does AWS help DevOps?

AWS (Amazon Web Services) is a popular cloud provider. It assists DevOps by offering the following benefits:.

Scaling: Plenty of machines can be set up on AWS by using computation power and unlimited storage.

AWS presents ready-to-use flexible resources.

With various services presented by AWS, a myriad of tasks can be automated.

AWS is safe to implement and through its diverse security options offered under the Identity and Access Management (IAM), certainly, the application deployments and builds can be safeguarded.

## Briefly explain various phases in DevOps:

AWS (Amazon Web Services) is a popular cloud provider. It assists DevOps by offering the following benefits:.

When you undergo [devops training](https://www.dotnettricks.com/training/masters-program/devops-certification-training" \t "https://www.dotnettricks.com/learn/devops/_blank)**,**you can get familiar with different phases in DevOps. The different phases available in the DevOps lifecycle are as below:

**Plan:** Firstly, there must be a plan for the type of application required to be developed. It is always a decent idea to get a rough picture of the development process.

**Code:** The application is coded according to the end-user requirements.

**Build:** It involves building the application by assimilating different codes created in the previous steps.

**Test:** Recognized as the most vital step of the application development, it involves testing the application and rebuilding, if necessary.

**Integrate:** Several codes from different programmers are combined into one.

**Deploy:** The code is deployed in a cloud environment for future usage. Any new modifications do not influence the overall functioning of a high-traffic website.

**Operate:** Operations are carried out on the code if needed.

**Monitor:** The performance of the application is monitored. Modifications are done to fulfill the end-user requirements.

## What are the antipatterns of DevOps?

A pattern is the one that is most frequently followed by huge masses of entities. When a pattern is accepted by an organization by considering that it is being used by others without determining the requirements of the organization then it turns out to be an anti-pattern. Likewise, various myths around DevOps can lead to antipatterns, they are as below:

DevOps is not a culture but a process.

DevOps is nothing but an Agile approach.

There must be a distinct DevOps group.

DevOps resolves every problem.

DevOps associates to developers operating a production environment.

DevOps follows Development-driven management

DevOps does not excessively focus on development.

DevOps culture cannot be implemented because of the absence of the right set of people.

## What are a few of the finest practices that must be followed for DevOps success?

Though you [Learn DevOps Step by Step](https://www.dotnettricks.com/courses/devops" \t "https://www.dotnettricks.com/learn/devops/_blank), it is important that you know which practices to follow for success in DevOps implementation.

Listed below are the important best practices necessary to be followed for DevOps implementation:

The speed of delivery implies the time taken for a task to continue into the production environment.

It is crucial to calculate the definite or the average time that it needs to recover in case any failure occurs in the production environment.

The number of bugs being mentioned by the customer also influences the overall quality of the application.

## What is the DevOps toolchain?

DevOps toolchain is formed by a stack of tools. It automates some tasks like the development and deployment of your application. Tasks in DevOps can be carried out manually with easy steps, but the requirement for automation rapidly increases due to the rise in its complexity. Therefore, toolchain automation is important for continuous delivery. Basically, GitHub a Version Control Repository is recognized as the core component of a DevOps toolchain. In this toolchain, some other tools can be delivery pipelines, backlog tracking, etc. When you enroll in the [Devops course in Hyderabad](https://www.dotnettricks.com/training/hyderabad/devops-online-training-hyderabad" \t "https://www.dotnettricks.com/learn/devops/_blank), you gradually become aware of various tools and toolchains in DevOps.

## What does it mean by a branching strategy in DevOps?

Branching is a method implemented for code isolation. It creates a copy of the source code to make two versions that are separately developed. Various kinds of branching are available. Thus, the DevOps team should make a choice based on application requirements. Correspondingly, the choice is referred to as a branching strategy.

## What is a Puppet?

Puppet is famously known as an open-source configuration management tool being utilized for the deployment, configuration, and management of servers. Essentially, its working is based on client-server architecture. In this architecture, the client is an agent whereas the server is recognized as the master. It is found that a puppet agent and master converse via a safely encrypted channel with the assistance of SSL.

****1. List out some famous tools of DevOps ?****

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

****2. Mention various phases of DevOps ?****

The given below are the different phases of DevOps

1. Plan: plan is essential for any application which requires the development, preparing plan for the process of development is the best practice.
2. Code: The code of application is designed according to the needs of users.
3. Build: with the help of different codes that are generated in earlier steps, the build of the application created.
4. 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.
5. Integrate: It is used to collaborate the various codes, which are from multiple programers.
6. 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.
7. Operate: When it is necessary operations take place on code.
8. Monitor: For the customer needs, the performance of the application is monitored.

****IMAGE****

****3. 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.

****4. 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.

****5. 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.

****6. 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.ld & Project Based Learning[Explore Curriculum](https://www.opstrainerz.com/azure-devops-training" \l "course-content" \t "https://www.opstrainerz.com/blog/_blank)

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.

****7. 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.

****8. 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.

****9. 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.

****10. 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.

****11. 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.

****12. 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.

****13. 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

****14. 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

****15. 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.

### Azure DevOps Interview Questions for Intermediate:

****1. Make a comparison between DevOps and agile methodology ?****

DevOps: It is a kind of culture which helps to collaborate operation teams and the development, and makes them work simultaneously. The output of this process is continuous integration, testing, deployment, development and it scans the entire lifecycle of the software.

****IMAGE****

Agile: It is a type of procedure of software which concentrates small, iterative, the quick delivery of the software and the feedback from users. This process displays the problems and the gaps between developers and the users.

****IMAGE****

****2. Make a comparison between Azure DevOps server and services ?****

Azure DevOps services: It is a microsoft service of cloud, which is reliable, scalable and hosted services which are available globally.

Azure DevOps server: It is a kind of on premise, which is used to build the back end server of SQL. People select this option for their requirement of their day with the network, the main reason for selecting it contains the requirement of SQL servers access by using azure data and tools.

They Both provide the same services along with particular advantages, the given below are some benefits of services

* It easily manages the server.
* Its remote sites are used for best connectivity.
* The latest features get quick access.[Subscribe](https://www.youtube.com/watch?v=8KyHhWgvVrE&t=2s" \t "https://www.opstrainerz.com/blog/_blank)

****3. Mention the reasons for why should we use azure boards ?****

Its characteristics and applications are the reason for selecting them, the given below are some of the reasons.

* Easy to begin by using scaling as levels increase.
* It provides the visual tools which are interactive.
* Simple to customize.
* It is used for social communication by using in-built tools.
* The collected data is flexible and has the capacity of storage.
* Simple to identify the needs and notify the changes taking place.
* It provides a chance to begin with no cost.
* It provide the advantage of extensibility
* It observes the progress and status in the dashboards.

****4. Made a comparison between VSTS and Azure DevOps ?****

The services of the microsoft visual studio namely Azure DevOps, which have better management tools for the application lifecycle. Our projects may plan with the templates and the tools of the Agile, it helps to run and manage the test plans, source code, version control and to manage the branches. By using CI and CD the solution deployment beyond all platforms with the help of azure pipeline.

****5. Explain the roles and their use ?****

They are kind of servers, which are managed and loaded as virtual machines that help to work for common aim achievement.

Azure have 3 kinds of roles

* Web Role
* Worker Role
* VM  Role

****6. Briefly explain the virtual machine scale sets ?****

a compute resource which is used for the deployment and the management of a pair of twin VMs, they are designed with the similar configuration for true autoscale. There is no special need for VMs pre-provisioning and simple for services of large scale, they concentrate on workload of containerised, big data.

****7. Make a note about availability sets ?****

This set is VMs logical group, which helps us to find how our application is built in Azure that offers availability, to get highly availability for application it is suggested that two or multiple VMs are designed in the sets of availability which leads to the Azure SLA at 99.95. If we use single VMs along with the storage of Azure premium, then for the maintenance of unplanned events Azure SLA is used.

****8. Introduce fault domains.****

They are nothing but the logical groups belonging to hardware, which can share the same network and power source. They are used to rack the data centers on premise, when we designed our VMs with help of availability sets then these fault domains are automatically distributed by the platform of Azure. The failures of physical hardware, outgages of network, etc are limited by this approach.

****9. Introduce Update domains.****

a hardware logical group which can be maintained or rebooted at the similar time, these updated domains automatically distribute our VMs when we design VM in the set of availability. This process is used to ensure our application instance and it keeps running as a platform of azure that goes under the maintenance of periodic. We can robot only a single order of updated domain at a time, it may not continuously proceed while the maintenance.

****10. Briefly explain about Network security groups ?****

NSG includes the access control list which may permit or neglect the traffic of a network for subnets, they may be associated with an NIC of individuals that connect with a subnet or with the subnets. If a subnet is associated with an NSG then a subnet ACL is applied for VM, it also limits the traffic for an individual NIC with the help of NSG.

****11. Are Scale sets works with availability sets of Azure ?****

Yes, it is an inferred set of availability, which contains 5 update domains and 5 fault domains. Its VMs span is more than 100, that is equal to various sets of availability. VMs set of availability is present in the same network, VMs control node is created by the common configuration in sets of availability.

****12. Explain about break fix issues.****

Break fix issues are nothing but the technical issues, that term belongs to industrial which includes work with technical support when it fails in normal functioning. This failure needs supportive organizations interruption for working order restoration.

### Azure DevOps Interview Questions for Experienced:

****1. Explain the Containers and Azure DevOps supports which containers ?****

Containers offer a simple software code approach which associates the dependencies, package and the configuration in a solitary project. We may increase the reply through the beginning, by which various containers run on identical mechain and share with other containers also. Containers used for the quick and well founded deployment, Azure DevOps gave support for Asp.Net, docker and the containers.

****2. Why should we use Azure pipelines, continuous integration and continuous development ?****

The execution of the azure pipelines, continuous integration and continuous development is the better way to secure the code quality. They provide a simple, secured and quick way for process automation, which helps to project build and also warrant the obtainability.

It offers free services for public projects and also 30 hours free service per month for private projects. The given points are some reasons for why we should use Azure DevOps pipelines, continuous integration and continuous delivery.

* It holds up any platform.
* The continuous deployment for different kinds of the targets.
* Azure development integration.
* It helps for windows, linux machines and the mac building.
* Github integration.
* Working abilities through the project of open source.

### **Azure Devops Training**

Weekday / Weekend Batches

[See Batch Details](https://www.opstrainerz.com/azure-devops-training" \l "events" \t "https://www.opstrainerz.com/blog/_blank)

****3. What is the use of an active directory in Azure ?****

It is the system of access management, which helped to access the grant for our employees to particular services and products of our networks. Twitter and salesforce.com are best examples for it, it provides application built support in galleries that may be directly added.

****4. If we do maximum failed attempts of authentication via Azure AD, then what happens ?****

We have more advanced strategies for accounts lock, that is based on requests of IP addresses and passwords which are entered. By using those advanced strategies we can increase the time of lockout, which seems like an attack.

****5. How to find applications list which are pre integrated with AD and their capabilities ?****

It has nearly 2,600 applications, which are pre integrated. Those applications keep up SSO, single sign on allow us to utilize the credentials of our organization for our apps access. They also gave automated support for de provisioning and provisioning.

****6. How to use Azure Ad applications ?****

It provides us a simple and secure way  for selecting and connecting with web applications, we also have the chance to access the same as our saas apps access in Azure. VPN is not required to change the infrastructure of our network.

****7. Write a note on Azure service fabric.****

It is a platform of distributed systems, which made it simple for package, deployment, scalable management and microservices reliability. It is also used to develop and manage the applications of cloud, developers may neglect the tough problems of infrastructure and pay attention to mission critical implementation, workloads demand which are reliable and scalable. They constitute for future use and platform of mildware for management of enterprise class and the applications of cloudscale.

****8. Explain about VNet ?****

It is nothing but the rendering of our cloud network, that isolates our launched cloud instances from our remaining resources.

****9. Bring Out the main variations between Subscription Administrator and Directory Administrator ?****

We need to assign a subscription administrator for Azure signup, it may use a Microsoft account or the work account of the directory. It is used by Azure portal management service, we can add co-admins when others are required to access with the same subscription.

It has various admin sets used to manage the features of the directory, they have different azure portal access. They can design and assign roles of administration for others, to reset the passwords and the domain management.

****10. What are the restrictions for customers using managed disks ?****

They abolish the restrictions that are associated with accounts storage, its number per subscription is restricted to 2000.

****11. Make a note about Azure Redis cache ?****

Azure redis cache is an open source data structure, which works as cache and database. It is based on the famous redis cache of open source that provides secured access from azure applications. It maintains structure for data they are like hashes, sets, strings along with queries.

## **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.

## **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?**

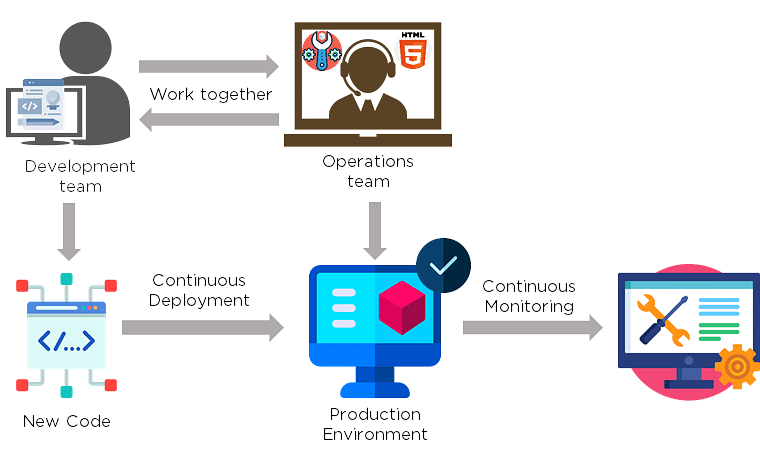
Azure Boards is an Azure DevOps service that helps in the management of work in software projects. Azure Boards provide a diverse set of capabilities such as customizable dashboards, integrated reporting, and native support for Kanban and Scrum. The core features of Azure Boards include work items, boards,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

### 1. What do you know about DevOps?

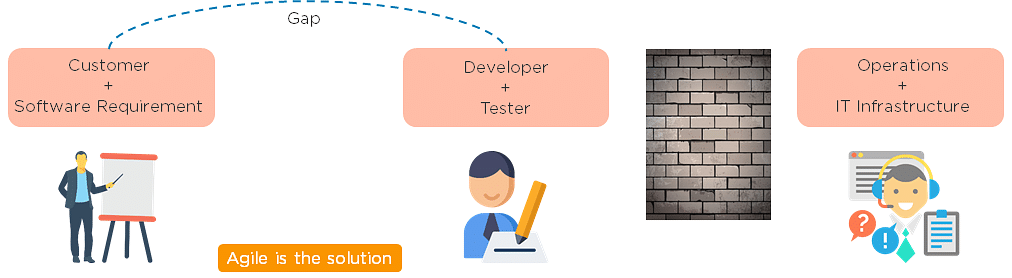
Your answer must be simple and straightforward. Begin by explaining the growing importance of DevOps in the [IT industry.](https://www.simplilearn.com/6-trends-shaking-up-the-it-industry-article" \o "IT industry." \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) Discuss how such an approach aims to synergize the efforts of the development and operations teams to accelerate the delivery of software products, with a minimal failure rate. Include how DevOps is a value-added practice, where development and operations engineers join hands throughout the product or service lifecycle, right from the design stage to the point of deployment.

### 2. How is DevOps different from agile methodology?

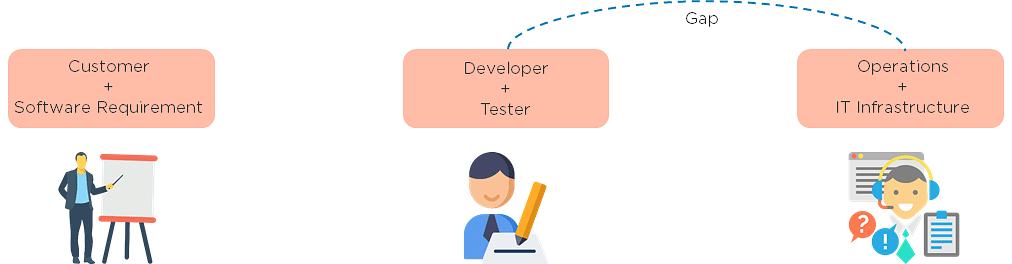
[DevOps is a culture](https://www.simplilearn.com/tutorials/devops-tutorial/what-is-devops" \o "DevOps is a culture" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) that allows the development and the operations team to work together. This results in [continuous development](https://www.simplilearn.com/tutorials/devops-tutorial/continuous-delivery-and-continuous-deployment" \o "continuous development" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank), testing, integration, deployment, and monitoring of the software throughout the lifecycle.



Agile is a [software development methodology](https://www.simplilearn.com/tutorials/agile-scrum-tutorial/what-is-agile" \o "software development methodology" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) that focuses on iterative, incremental, small, and rapid releases of software, along with customer feedback. It addresses gaps and conflicts between the customer and developers.



DevOps addresses gaps and conflicts between the Developers and IT Operations.



### 3. Which are some of the most popular DevOps tools?

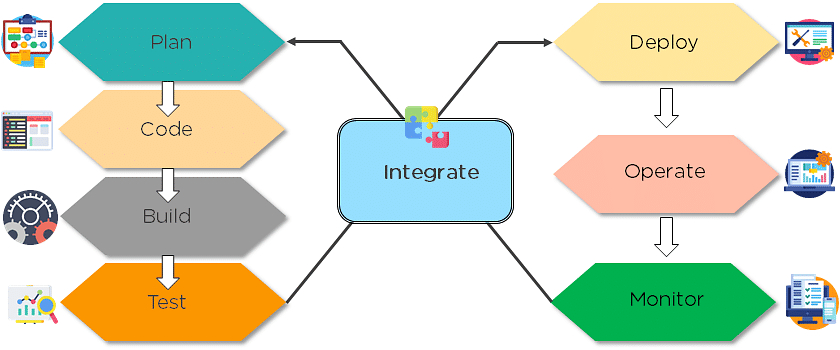
The most popular [DevOps tools](https://www.simplilearn.com/tutorials/devops-tutorial/devops-tools" \o "DevOps tools" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) include:

1. [Selenium](https://www.simplilearn.com/tutorials/selenium-tutorial/what-is-selenium" \o "Selenium" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)
2. [Puppet](https://www.simplilearn.com/puppet-tutorial-article" \o "Puppet" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)
3. [Chef](https://www.simplilearn.com/chef-tutorial-article" \o "Chef" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)
4. [Git](https://www.simplilearn.com/tutorials/git-tutorial" \o "Git" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)
5. [Jenkins](https://www.simplilearn.com/tutorials/jenkins-tutorial/what-is-jenkins" \o "Jenkins" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)
6. [Ansible](https://www.simplilearn.com/tutorials/ansible-tutorial" \o "Ansible" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)
7. [Docker](https://www.simplilearn.com/tutorials/docker-tutorial" \o "Docker" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)

### . What are the different phases in DevOps?

The various phases of the DevOps lifecycle are as follows:

* Plan - Initially, there should be a plan for the type of application that needs to be developed. Getting a rough picture of the development process is always a good idea.
* Code - The application is coded as per the end-user requirements.
* Build - Build the application by integrating various codes formed in the previous steps.
* Test - This is the most crucial step of the application development. Test the application and rebuild, if necessary.
* Integrate - Multiple codes from different programmers are integrated into one.
* Deploy - Code is deployed into a cloud environment for further usage. It is ensured that any new changes do not affect the functioning of a high traffic website.
* Operate - Operations are performed on the code if required.
* Monitor - Application performance is monitored. Changes are made to meet the end-user requirements.



The above figure indicates the DevOps lifecycle.

### 5. Mention some of the core benefits of DevOps.

The core benefits of DevOps are as follows:

#### Technical benefits

* Continuous software delivery
* Less complex problems to manage
* Early detection and faster correction of defects

#### Business benefits

* Faster delivery of features
* Stable operating environments
* Improved communication and collaboration between the teams

Also Read: [How to Become a DevOps Engineer?](https://www.simplilearn.com/tutorials/devops-tutorial/how-to-become-devops-engineer" \o "How to Become a DevOps Engineer?" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)

### 6. How will you approach a project that needs to implement DevOps?

The following standard approaches can be used to implement DevOps in a specific project:

Stage 1

An assessment of the existing process and implementation for about two to three weeks to identify areas of improvement so that the team can create a road map for the implementation.

Stage 2

Create a proof of concept (PoC). Once it is accepted and approved, the team can start on the actual implementation and roll-out of the project plan.

Stage 3

The project is now ready for implementing DevOps by using version control/integration/testing/deployment/delivery and monitoring followed step by step.

By following the proper steps for [version control](https://www.simplilearn.com/tutorials/devops-tutorial/version-control" \o "version control" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank), integration, testing, deployment, delivery, and monitoring, the project is now ready for DevOps implementation.

### 7. What is the difference between continuous delivery and continuous deployment?

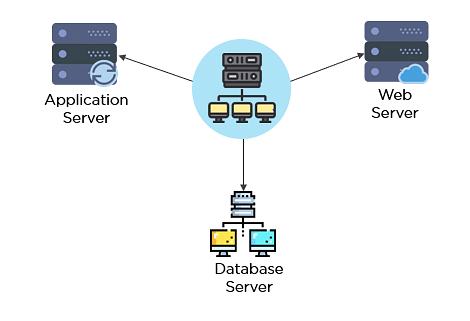
|  |  |
| --- | --- |
| Continuous Delivery | Continuous Deployment |
| Ensures code can be safely deployed on to production | Every change that passes the automated tests is deployed to production automatically |
| Ensures business applications and services function as expected | Makes software development and the release process faster and more robust |
| Delivers every change to a production-like environment through rigorous automated testing | There is no explicit approval from a developer and requires a developed culture of monitoring |

### IMG_261

### 8. What is the role of configuration management in DevOps?

* Enables management of and changes to multiple systems.
* Standardizes resource configurations, which in turn, manage IT infrastructure.
* It helps with the administration and management of multiple servers and maintains the integrity of the entire infrastructure.

### 9. How does continuous monitoring help you maintain the entire architecture of the system?



Continuous monitoring in DevOps is a process of detecting, identifying, and reporting any faults or threats in the entire infrastructure of the system.

* Ensures that all services, applications, and resources are running on the servers properly.
* Monitors the status of servers and determines if applications are working correctly or not.
* Enables continuous audit, transaction inspection, and controlled monitoring.

### 10. What is the role of AWS in DevOps?

AWS has the following role in DevOps:

* Flexible services - Provides ready-to-use, flexible services without the need to install or set up the software.
* Built for scale - You can manage a single instance or scale to thousands using AWS services.
* Automation - AWS lets you automate tasks and processes, giving you more time to innovate
* Secure - Using AWS Identity and Access Management (IAM), you can set user permissions and policies.
* Large partner ecosystem - AWS supports a large ecosystem of partners that integrate with and extend AWS services.

### 11. Name three important DevOps KPIs.

The three important KPIs are as follows:

* Meantime to failure recovery - This is the average time taken to recover from a failure.
* Deployment frequency - The frequency in which the deployment occurs.
* Percentage of failed deployments - The number of times the deployment fails.

### 12. Explain the term "Infrastructure as Code" (IaC) as it relates to configuration management.

* Writing code to manage configuration, deployment, and automatic provisioning.
* Managing data centers with machine-readable definition files, rather than physical hardware configuration.
* Ensuring all your servers and other infrastructure components are provisioned consistently and effortlessly.
* Administering cloud computing environments, also known as [infrastructure as a service (IaaS)](https://www.simplilearn.com/what-is-infrastructure-as-code-article" \o "infrastructure as a service (IaaS)" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank).

### 13. How is IaC implemented using AWS?

Start by talking about the age-old mechanisms of writing commands onto script files and testing them in a separate environment before deployment and how this approach is being replaced by IaC. Similar to the codes written for other services, with the help of AWS, IaC allows developers to write, test, and maintain infrastructure entities in a descriptive manner, using formats such as JSON or YAML. This enables easier development and faster deployment of infrastructure changes.

### 14. Why Has DevOps Gained Prominence over the Last Few Years?

Before talking about the growing popularity of DevOps, discuss the current industry scenario. Begin with some examples of how big players such as [Netflix and Facebook](https://techbeacon.com/10-companies-killing-it-devops" \o "Netflix and Facebook" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) are investing in DevOps to automate and accelerate application deployment and how this has helped them grow their business. Using Facebook as an example, you would point to Facebook’s continuous deployment and code ownership models and how these have helped it scale up but ensure the quality of experience at the same time. Hundreds of lines of code are implemented without affecting quality, stability, and security.

Your next use case should be Netflix. This streaming and on-demand video company follow similar practices with fully automated processes and systems. Mention the user base of these two organizations: Facebook has 2 billion users while Netflix streams online content to more than 100 million users worldwide.

These are great examples of how DevOps can help organizations to ensure higher success rates for releases, reduce the lead time between bug fixes, streamline and continuous delivery through automation, and an overall reduction in manpower costs.

We will now look into the next set of DevOps Interview Questions that includes - Git, Selenium, Jenkins.

### 15. What are the fundamental differences between DevOps & Agile?

The main differences between Agile and DevOps are summarized below:

|  |  |  |
| --- | --- | --- |
| Characteristics | Agile | DevOps |
| Work Scope | Only Agility | Automation needed along with Agility |
| Focus Area | Main priority is Time and deadlines | Quality and Time management are of equal priority |
| Feedback Source | The main source of feedback - customers | The main source of feedback - self (tools used for monitoring) |
| Practices or Processes followed | Practices like Agile Kanban, Scrum, etc., are followed. | Processes and practices like Continuous Development (CD), Continuous Integration (CI), etc., are followed. |
| Development Sprints or Release cycles | Release cycles are usually smaller. | Release cycles are smaller, along with immediate feedback. |
| Agility | Only development agility is present. | Both in operations and development, agility is followed. |

### 16. What are the anti-patterns of DevOps?

Patterns are common practices that are usually followed by organizations. An anti-pattern is formed when an organization continues to blindly follow a pattern adopted by others but does not work for them. Some of the myths about DevOps include:

* Cannot perform DevOps → Have the wrong people
* DevOps ⇒ Production Management is done by developers
* The solution to all the organization’s problems ⇒ DevOps
* DevOps == Process
* DevOps == Agile
* Cannot perform DevOps → Organization is unique
* A separate group needs to be made for DevOps

### 17. What are the benefits of using version control?

Here are the benefits of using Version Control:

* All team members are free to work on any file at any time with the Version Control System (VCS). Later on, VCS will allow the team to integrate all of the modifications into a single version.
* The VCS asks to provide a brief summary of what was changed every time we save a new version of the project. We also get to examine exactly what was modified in the content of the file. As a result, we will be able to see who made what changes to the project.
* Inside the VCS, all the previous variants and versions are properly stored. We will be able to request any version at any moment, and we will be able to retrieve a snapshot of the entire project at our fingertips.
* A VCS that is distributed, such as Git, lets all the team members retrieve a complete history of the project. This allows developers or other stakeholders to use the local Git repositories of any of the teammates even if the main server goes down at any point in time.

### 18. Describe the branching strategies you have used.

To test our knowledge of the purpose of branching and our experience of branching at a past job, this question is usually asked.

Below topics can help in answering this DevOps interview question -

* Release branching - We can clone the develop branch to create a Release branch once it has enough functionality for a release. This branch kicks off the next release cycle, thus no new features can be contributed beyond this point. The things that can be contributed are documentation generation, bug fixing, and other release-related tasks. The release is merged into master and given a version number once it is ready to ship. It should also be merged back into the development branch, which may have evolved since the initial release.
* Feature branching - This branching model maintains all modifications for a specific feature contained within a branch. The branch gets merged into master once the feature has been completely tested and approved by using tests that are automated.

Task branching - In this branching model, every task is implemented in its respective branch. The task key is mentioned in the branch name. We need to simply look at the task key in the branch name to discover which code implements which task.

### 19. Can you explain the “Shift left to reduce failure” concept in DevOps?

Shift left is a DevOps idea for improving security, performance, and other factors. Let us take an example: if we look at all of the processes in DevOps, we can state that security is tested prior to the deployment step. We can add security in the development phase, which is on the left, by employing the left shift method. [will be depicted in a diagram] We can integrate with all phases, including before development and during testing, not just development. This most likely raises the security level by detecting faults at an early stage.

### 20. What is the Blue/Green Deployment Pattern?

This is a method of continuous deployment that is commonly used to reduce downtime. This is where traffic is transferred from one instance to another. In order to include a fresh version of code, we must replace the old code with a new code version.

The new version exists in a green environment and the old version exists in a blue environment. After making changes to the previous version, we need a new instance from the old one to execute a newer version of the instance.

### 21. What is Continuous Testing?

Continuous Testing constitutes the running of automated tests as part of the software delivery pipeline to provide instant feedback on the business risks present in the most recent release. In order to prevent problems in step-switching in the Software delivery life-cycle and to allow Development teams to receive immediate feedback, every build is continually tested in this manner. This results in significant increase in speed in a developer's productivity as it eliminates the requirement for re-running all the tests after each update and project re-building.

### 22. What is Automation Testing?

Test automation or manual testing Automation is the process of automating a manual procedure in order to test an application or system. [Automation testing](https://www.simplilearn.com/what-is-automation-testing-article" \o "Automation testing" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) entails the use of independent testing tools that allow you to develop test scripts that can be run repeatedly without the need for human interaction.

### 23. What are the benefits of Automation Testing?

Some of the advantages of Automation Testing are -

* Helps to save money and time.
* Unattended execution can be easily done.
* Huge test matrices can be easily tested.
* Parallel execution is enabled.
* Reduced human-generated errors, which results in improved accuracy.
* Repeated test tasks execution is supported.

### 24. How to automate Testing in the DevOps lifecycle?

Developers are obliged to commit all source code changes to a shared DevOps repository.

Every time a change is made in the code, Jenkins-like Continuous Integration tools will grab it from this common repository and deploy it for Continuous Testing, which is done by tools like Selenium.

### 25. Why is Continuous Testing important for DevOps?

Any modification to the code may be tested immediately with Continuous Testing. This prevents concerns like quality issues and release delays that might occur whenever big-bang testing is delayed until the end of the cycle. In this way, Continuous Testing allows for high-quality and more frequent releases.

### 26. What are the key elements of Continuous Testing tools?

Continuous Testing key elements are:

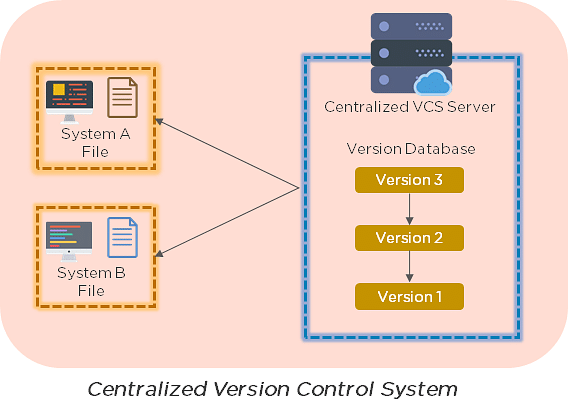
* Test Optimization - It guarantees that tests produce reliable results and actionable information. Test Data Management, Test Optimization Management, and Test Maintenance are examples of aspects.
* Advanced Analysis - In order to avoid problems from occurring in the first place and to achieve more within each iteration, it employs automation in areas like scope assessment/prioritization, changes effect analysis, and static code analysis.
* Policy Analysis - It guarantees that all processes are in line with the organization's changing business needs and that all compliance requirements are met.
* Risk Assessment - Test coverage optimization, technical debt, risk mitigation duties, and quality evaluation are all covered to guarantee the build is ready to move on to the next stage.
* Service Virtualization - Ensures that real-world testing scenarios are available. Service visualisation provides access to a virtual representation of the needed testing phases, ensuring its availability and reducing the time spent setting up the test environment.
* Requirements Traceability - It guarantees that no rework is necessary and real criteria are met. To determine which needs require additional validation, are in jeopardy and performing as expected, an object evaluation is used.

## DevOps Interview Questions for Source Code Management — Git

### 27. Explain the difference between a centralized and distributed version control system (VCS).

#### Centralized Version Control System

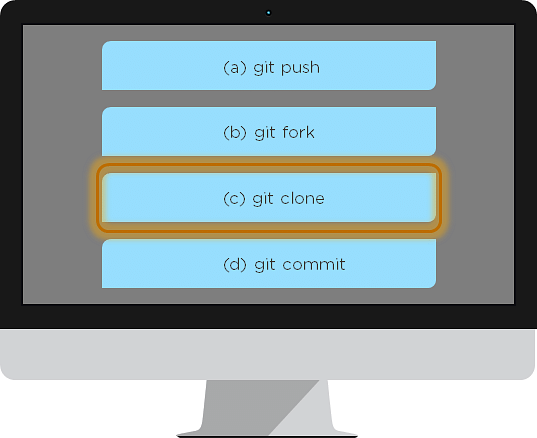
* All file versions are stored on a central server
* No developer has a copy of all files on a local system
* If the central server crashes, all data from the project will be lost



#### Distributed Control System

* Every developer has a copy of all versions of the code on their systems
* Enables team members to work offline and does not rely on a single location for backups
* There is no threat, even if the server crashes

### 28. What is the git command that downloads any repository from GitHub to your computer?



The [git command](https://www.simplilearn.com/tutorials/git-tutorial/git-commands" \o "git command" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) that downloads any repository from GitHub to your computer is git clone.

### 29. How do you push a file from your local system to the GitHub repository using Git?

First, connect the local repository to your remote repository:

git remote add origin [copied web address]

// Ex: git remote add origin [https://github.com/Simplilearn-github/test.git](https://github.com/Simplilearn-github/test.git" \o "https://github.com/Simplilearn-github/test.git" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)

Second, push your file to the remote repository:

git push origin master

### 30. How is a bare repository different from the standard way of initializing a Git repository?

Using the standard method:

#### git init

* You create a working directory with git init
* A .git subfolder is created with all the git-related revision history

Using the bare way

#### git init --bare

* It does not contain any working or checked out a copy of source files
* Bare repositories store git revision history in the root folder of your repository, instead of the .git subfolder

### 31. Which of the following CLI commands can be used to rename files?

1. git rm
2. git mv
3. git rm -r
4. None of the above

The correct answer is B) git mv

### 32. What is the process for reverting a commit that has already been pushed and made public?

There are two ways that you can revert a commit:

1. Remove or fix the bad file in a new commit and push it to the remote repository. Then commit it to the remote repository using:  
     
   git commit –m "commit message"
2. Create a new commit that undoes all the changes that were made in the bad commit. Use the following command:  
     
   git revert <commit id>

Example: git revert 56de0938f

#### FREE GIT Training

Learn the basics of GIT[ENROLL NOW](https://www.simplilearn.com/learn-git-basics-skillup?utm_source=frs&utm_medium=skillup-course-banner&utm_campaign=frs-skillup-course-promotion" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)

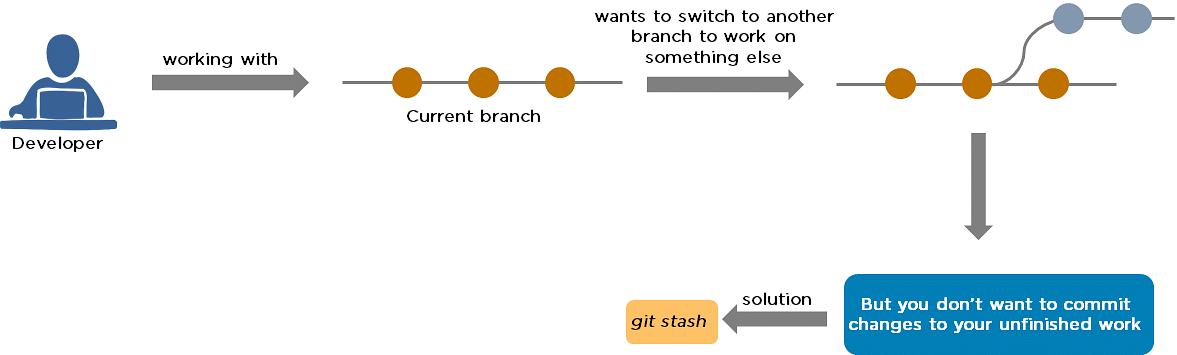


### 33. Explain the difference between git fetch and git pull.

|  |  |
| --- | --- |
| Git fetch | Git pull |
| Git fetch only downloads new data from a remote repository | Git pull updates the current HEAD branch with the latest changes from the remote server |
| Does not integrate any new data into your working files | Downloads new data and integrate it with the current working files |
| Users can run a Git fetch at any time to update the remote-tracking branches | Tries to merge remote changes with your local ones |
| Command - git fetch origin                    git fetch –-all | Command - git pull origin master |

### 34. What is Git stash?

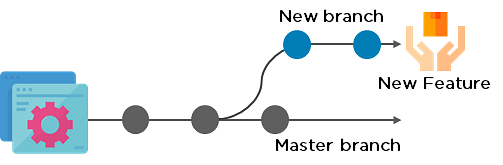
A developer working with a current branch wants to switch to another branch to work on something else, but the developer doesn't want to commit changes to your unfinished work. The solution to this issue is Git stash. Git stash takes your modified tracked files and saves them on a stack of unfinished changes that you can reapply at any time.



### 35. Explain the concept of branching in Git.

Suppose you are working on an application, and you want to add a new feature to the app. You can create a new branch and build the new feature on that branch.

* By default, you always work on the master branch
* The circles on the branch represent various commits made on the branch
* After you are done with all the changes, you can merge it with the master branch



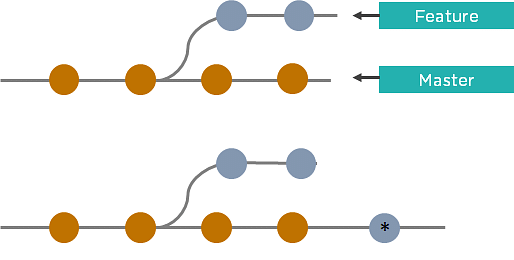
### 36. What is the difference between Git Merge and Git Rebase?

Suppose you are working on a new feature in a dedicated branch, and another team member updates the master branch with new commits. You can use these two functions:

Git Merge

To incorporate the new commits into your feature branch, use Git merge.

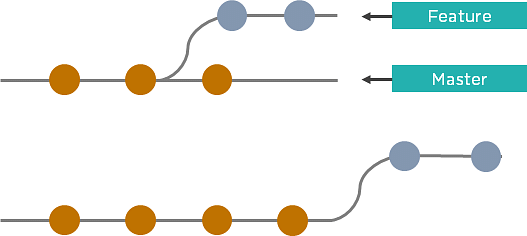
* Creates an extra merge commit every time you need to incorporate changes
* But, it pollutes your feature branch history



Git Rebase

As an alternative to merging, you can rebase the feature branch on to master.

* Incorporates all the new commits in the master branch
* It creates new commits for every commit in the original branch and rewrites project history



### 37. How do you find a list of files that have been changed in a particular commit?

The command to get a list of files that have been changed in a particular commit is:

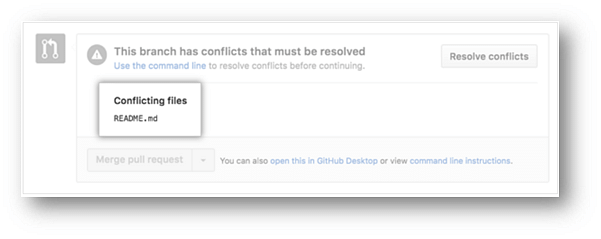
git diff-tree –r {commit hash}

Example: git diff-tree –r 87e673f21b

* -r flag instructs the command to list individual files
* commit hash will list all the files that were changed or added in that commit

### 38. What is a merge conflict in Git, and how can it be resolved?

A [Git merge conflict](https://www.simplilearn.com/tutorials/git-tutorial/merge-conflicts-in-git" \o "Git merge conflict" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) happens when you have merge branches with competing for commits, and Git needs your help to decide which changes to incorporate in the final merge.

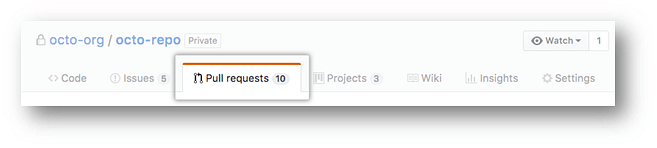


Manually edit the conflicted file to select the changes that you want to keep in the final merge.

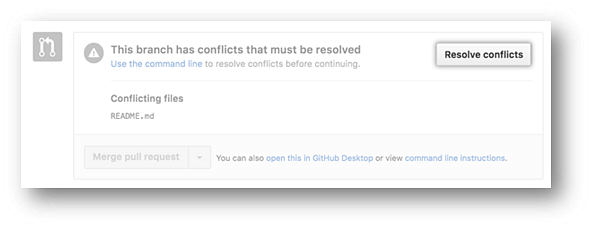
Resolve using GitHub conflict editor

This is done when a merge conflict is caused after competing for line changes. For example, this may occur when people make different changes to the same line of the same file on different branches in your Git repository.

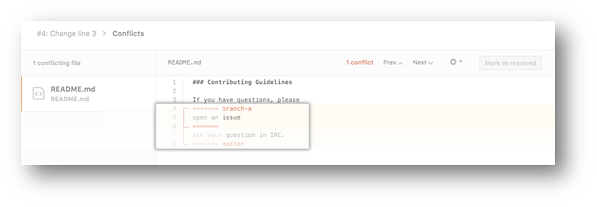
* Resolving a merge conflict using conflict editor:
* Under your repository name, click "Pull requests."



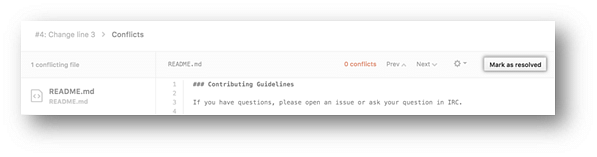
* In the "Pull requests" drop-down, click the pull request with a merge conflict that you'd like to resolve
* Near the bottom of your pull request, click "Resolve conflicts."



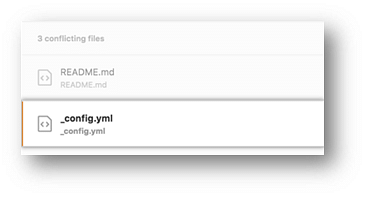
* Decide if you only want to keep your branch's changes, the other branch's changes, or make a brand new change, which may incorporate changes from both branches.
* Delete the conflict markers <<<<<<<, =======, >>>>>>> and make changes you want in the final merge.



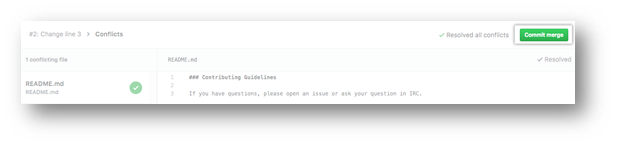
* If you have more than one merge conflict in your file, scroll down to the next set of conflict markers and repeat steps four and five to resolve your merge conflict.
* Once you have resolved all the conflicts in the file, click Mark as resolved.



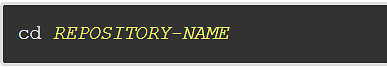
* If you have more than one file with a conflict, select the next file you want to edit on the left side of the page under "conflicting files" and repeat steps four to seven until you've resolved all of your pull request's merge conflicts.



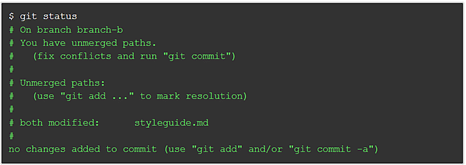
* Once you've resolved your merge conflicts, click Commit merge. This merges the entire base branch into your head branch.



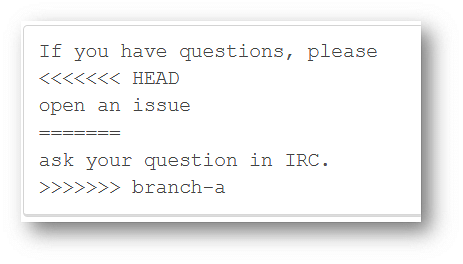
* To merge your pull request, click Merge pull request.
* A merge conflict is resolved using the command line.
* Open Git Bash.
* Navigate into the local Git repository that contains the merge conflict.



* Generate a list of the files that the merge conflict affects. In this example, the file styleguide.md has a merge conflict.

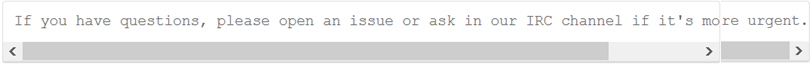


* Open any text editor, such as Sublime Text or Atom, and navigate to the file that has merge conflicts.
* To see the beginning of the merge conflict in your file, search the file for the conflict marker "<<<<<<<. " Open it, and you'll see the changes from the base branch after the line "<<<<<<< HEAD."
* Next, you'll see "=======", which divides your changes from the changes in the other branch, followed by ">>>>>>> BRANCH-NAME".



* Decide if you only want to keep your branch's changes, the other branch's changes, or make a brand new change, which may incorporate changes from both branches.
* Delete the conflict markers "<<<<<<<", "=======", ">>>>>>>" and make the changes you want in the final merge.

        In this example, both the changes are incorporated into the final merge:



* Add or stage your changes.



* Commit your changes with a comment.

IMG_284

Now you can merge the branches on the command line, or push your changes to your remote repository on GitHub and merge your changes in a pull request.

### 39. What is Git bisect? How can you use it to determine the source of a (regression) bug?

Git bisect is a tool that uses binary search to locate the commit that triggered a bug.

Git bisect command -

git bisect <subcommand> <options>

The git bisect command is used in finding the bug performing commit in the project by using a binary search algorithm.

The bug occurring commit is called the “bad” commit, and the commit before the bug occurring one is called the “good” commit. We convey the same to the git bisect tool, and it picks a random commit between the two endpoints and prompts whether that one is the “good” or “bad” one. The process continues uptil the range is narrowed down and the exact commit that introduced the exact change is discovered.

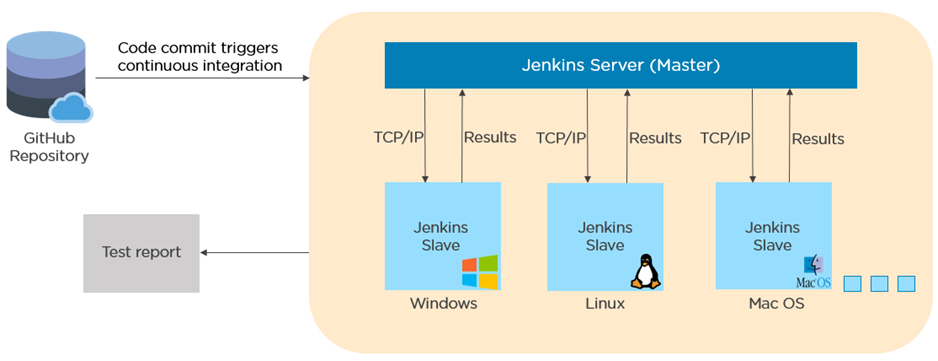
### 40. Explain some basic Git commands.

Some of the Basic Git Commands are summarized in the below table -

|  |  |
| --- | --- |
| Command | Purpose |
| git init | Used to start a new repository. |
| git config -   * git config –global user.name “[name]” * git config –global user.email “[email address]” | This helps to set the username and email to whom the commits belong to. |
| git clone <repository path> | Used to create a local copy of an existing repository. |
| git add -   * git add <file names separated by commas> * git add . | Used to add one or more files to the staging area. |
| git commit -   * git commit -a * git commit -m “<add commit message>” | Creates a snapshot or records of the file(s) that are in the staging area. |
| git diff -   * git diff [first branch] [second branch] * git diff -staged | Used to show differences between the two mentioned branches/differences made in the files in the staging area vs current version. |
| git status | Lists out all the files that are to be committed. |
| git rm <file name(s)> | Used to delete a file(s) from the current working directory and also stages it. |
| git show <commit> | Shows the content changes and metadata of the mentioned commit. |
| git branch -   * git branch [branch name] * git branch -d [branch name] * git branch | The first one creates a brand new branch.  The second is used to delete the mentioned branch.  The last one lists out all the branches available and also highlights the branch we are in currently. |

## DevOps Interview Questions for Continuous Integration - Jenkins

### 41. Explain the master-slave architecture of Jenkins.



* Jenkins master pulls the code from the remote GitHub repository every time there is a code commit.
* It distributes the workload to all the Jenkins slaves.
* On request from the Jenkins master, the slaves carry out, builds, test, and produce test reports.

### 42. What is Jenkinsfile?

Jenkinsfile contains the definition of a Jenkins pipeline and is checked into the source control repository. It is a text file.

* It allows code review and iteration on the pipeline.
* It permits an audit trail for the pipeline.
* There is a single source of truth for the pipeline, which can be viewed and edited.

### 43. Which of the following commands runs Jenkins from the command line?

1. java –jar Jenkins.war
2. java –war Jenkins.jar
3. java –jar Jenkins.jar
4. java –war Jenkins.war

The correct answer is A) java –jar Jenkins.war

### 44. What concepts are key aspects of the Jenkins pipeline?

* Pipeline: User-defined model of a CD pipeline. The pipeline's code defines the entire build process, which includes building, testing and delivering an application
* Node: A machine that is part of the Jenkins environment and capable of executing a pipeline
* Step: A single task that tells Jenkins what to do at a particular point in time
* Stage: Defines a conceptually distinct subset of tasks performed through the entire pipeline (build, test, deploy stages)

### 45. Which file is used to define dependency in Maven?

1. build.xml
2. pom.xml
3. dependency.xml
4. Version.xml

The correct answer is B) pom.xml

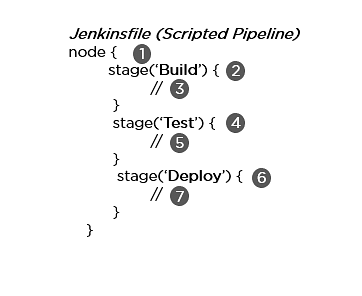
### 46. Explain the two types of pipeline in Jenkins, along with their syntax.

Jenkins provides two ways of developing a pipeline code: Scripted and Declarative.

A. Scripted Pipeline: It is based on Groovy script as their Domain Specific Language. One or more node blocks do the core work throughout the entire pipeline.

Syntax:

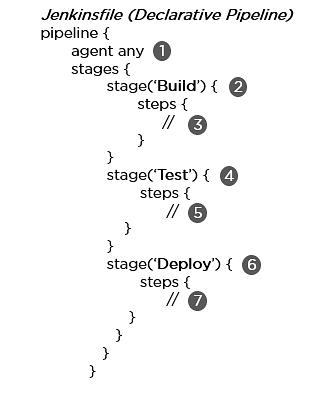
1. Executes the pipeline or any of its stages on any available agent
2. Defines the build stage
3. Performs steps related to building stage
4. Defines the test stage
5. Performs steps related to the test stage
6. Defines the deploy stage
7. Performs steps related to the deploy stage



B. Declarative Pipeline: It provides a simple and friendly syntax to define a pipeline. Here, the pipeline block defines the work done throughout the pipeline.

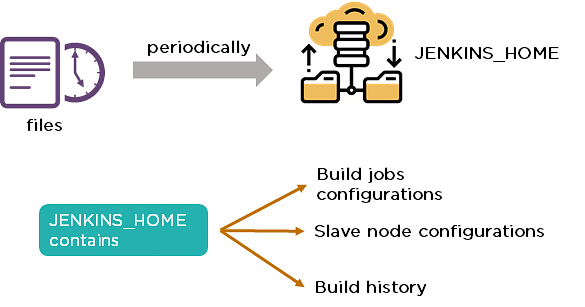
Syntax:

1. Executes the pipeline or any of its stages on any available agent
2. Defines the build stage
3. Performs steps related to building stage
4. Defines the test stage
5. Performs steps related to the test stage
6. Defines the deploy stage
7. Performs steps related to the deploy stage



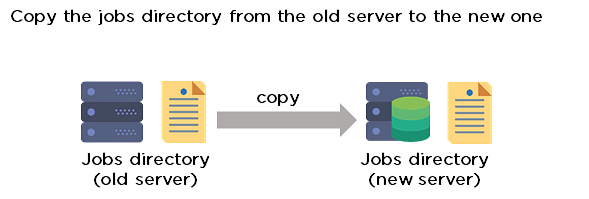
### 47. How do you create a backup and copy files in Jenkins?

In order to create a backup file, periodically back up your JENKINS\_HOME directory.



In order to create a backup of Jenkins setup, copy the JENKINS\_HOME directory. You can also copy a job directory to clone or replicate a job or rename the directory.

### 48. How can you copy Jenkins from one server to another?



* Move the job from one Jenkins installation to another by copying the corresponding job directory.
* Create a copy of an existing job by making a clone of a job directory with a different name.
* Rename an existing job by renaming a directory.

### 49. Name three security mechanisms Jenkins uses to authenticate users.

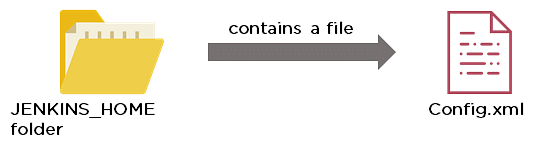
* Jenkins uses an internal database to store user data and credentials.
* Jenkins can use the Lightweight Directory Access Protocol (LDAP) server to authenticate users.
* Jenkins can be configured to employ the authentication mechanism that the deployed application server uses.

### 50. How is a custom build of a core plugin deployed?

Steps to deploy a custom build of a core plugin:

* Copy the .hpi file to $JENKINS\_HOME/plugins
* Remove the plugin's development directory
* Create an empty file called <plugin>.hpi.pinned
* Restart Jenkins and use your custom build of a core plugin

### 51. How can you temporarily turn off Jenkins security if the administrative users have locked themselves out of the admin console?



* When security is enabled, the Config file contains an XML element named useSecurity that will be set to true.
* By changing this setting to false, security will be disabled the next time Jenkins is restarted.

### 52. What are the ways in which a build can be scheduled/run in Jenkins?

* By source code management commits.
* After completion of other builds.
* Scheduled to run at a specified time.
* Manual build requests.

### 53. What are the commands that you can use to restart Jenkins manually?

Two ways to manually restart Jenkins:

1. (Jenkins\_url)/restart            // Forces a restart without waiting for builds to complete
2. (Jenkins\_url)/safeRestart    // Allows all running builds to complete before it restarts

### 54. Explain how you can set up a Jenkins job?

To create a Jenkins Job, we go to the top page of Jenkins, choose the New Job option and then select Build a free-style software project.

The elements of this freestyle job are -

* Optional triggers for controlling when Jenkins builds.
* Optional steps for gathering data from the build, like collecting javadoc, testing results and/or archiving artifacts.
* A build script (ant, maven, shell script, batch file, etc.) that actually does the work.
* Optional source code management system (SCM), like Subversion or CVS.

## DevOps Interview Questions for Continuous Testing - Selenium

### 55. What are the different Selenium components?

Selenium has the following components:

[Selenium Integrated Development Environment (IDE)](https://www.simplilearn.com/tutorials/selenium-tutorial/selenium-ide" \o "Selenium Integrated Development Environment (IDE)" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)

* It has a simple framework and should be used for prototyping.
* It has an easy-to-install Firefox plug-in.

Selenium Remote Control (RC)

* Testing framework for a developer to write code in any programming language (Java, PHP, Perl, C#, etc.).

[Selenium WebDriver](https://www.simplilearn.com/tutorials/selenium-tutorial/what-is-selenium-webdriver" \o "Selenium WebDriver" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)

* Applies a better approach to automate browser activities.
* It does not rely on JavaScript.

Selenium Grid

* Works with Selenium RC and runs tests on different nodes using browsers.

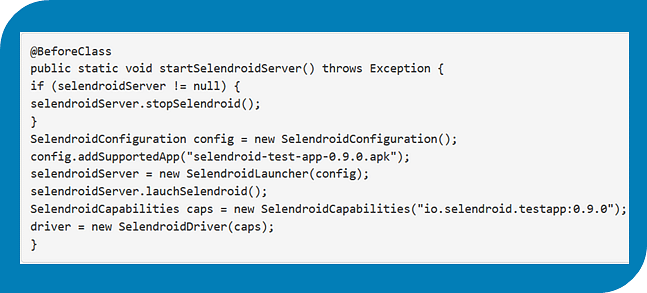
### 56. What are the different exceptions in Selenium WebDriver?

Exceptions are events that occur during the execution of a program and disrupt the normal flow of a program's instructions. Selenium has the following exceptions:

* TimeoutException - It is thrown when a command performing an operation does not complete in the stipulated time.
* NoSuchElementException - It is thrown when an element with specific attributes is not found on the web page.
* ElementNotVisibleException - It is thrown when an element is present in Document Object Model (DOM) but is not visible. Ex: Hidden Elements defined in HTML using type=“hidden”.
* SessionNotFoundException - The WebDriver is performing the action immediately after quitting the browser.

### 57. Can Selenium test an application on an Android browser?

Selenium is capable of testing an application on an Android browser using an Android driver. You can use the Selendroid or Appium framework to test native apps or web apps in the Android browser. The following is a sample code:



### 58. What are the different test types that Selenium supports?

Functional - This is a type of black-box testing in which the test cases are based on the software specification.

Regression - This testing helps to find new errors, regressions, etc. in different functional and non-functional areas of code after the alteration.

Load Testing - This testing seeks to monitor the response of a device after putting a load on it. It is carried out to study the behavior of the system under certain conditions.

### 59. How can you access the text of a web element?

Get command is used to retrieve the text of a specified web element. The command does not return any parameter but returns a string value.

Used for:

* Verification of messages
* Labels
* Errors displayed on the web page

Syntax:

String Text=driver.findElement(By.id(“text”)).getText();

### 60. How can you handle keyboard and mouse actions using Selenium?

You can handle keyboard and mouse events with the advanced user interaction API. The advanced user interactions API contains actions and action classes.

|  |  |
| --- | --- |
| Method | Description |
| clickAndHold() | Clicks without releasing the current mouse location |
| dragAndDrop() | Performs click-and-hold at the location of the source element |
| keyDown(modifier\_key) | Performs a modifier key press (ctrl, shift, Fn, etc.) |
| keyUp(modifier\_key) | Performs a key release |

### 60. Which of these options is not a WebElement method?

1. getText()
2. size()
3. getTagName()
4. sendKeys()

The correct answer is B) size()

### 61. When do we use findElement() and findElements()?

A. findElement()

It finds the first element in the current web page that matches the specified locator value.

Syntax:

WebElement element=driver.findElements(By.xpath(“//div[@id=‘example’]//ul//li”));

B. findElements()

It finds all the elements in the current web page that matches the specified locator value.

Syntax:

List elementList=driver.findElements(By.xpath(“//div[@id=‘example’]//ul//li”));

### 62. What are driver.close() and driver.quit() in WebDriver?

These are two different methods used to close the browser session in Selenium WebDriver:

* driver.close() - This is used to close the current browser window on which the focus is set. In this case, there is only one browser open.
* driver.quit() - It closes all the browser windows and ends the WebDriver session using the driver.dispose method.

### 63. How can you submit a form using Selenium?

The following lines of code will let you submit a form using Selenium:

WebElement el = driver.findElement(By.id(“ElementID”));

el.submit();

### 64. What are the Testing types supported by Selenium?

There are two types of testing that are primarily supported by Selenium:

Functional Testing - Individual testing of software functional points or features.

Regression Testing - Wherever a bug is fixed, a product is retested and this is called Regression Testing.

### 65. What is Selenium IDE?

Selenium integrated development environment (IDE)  is an all-in-one Selenium script development environment. It may be used to debug tests, alter and record and is also available as a Firefox extension. Selenium IDE comes with the whole Selenium Core that  allows us to rapidly and easily replay and record  tests in the exact environment where they will be conducted.

Selenium IDE is the best environment for building Selenium tests, regardless of the style of testing we prefer, thanks to the ability to move instructions around rapidly and the autocomplete support.

### 66. What is the difference between Assert and Verify commands in Selenium?

The difference between Verify and Assert commands in Selenium are:

* The verify commands determine whether or not the provided condition is true. The program execution does not halt regardless of whether the condition is true or not, i.e., all test steps will be completed, and verification failure will not stop the execution.
* The assert command determines whether a condition is false or true. To know whether the supplied element is on the page or not, we do the following. The next test step will be performed by the program control, if the condition is true. However, no further tests will be run, and the execution will halt, if the condition is false.

### 67. How to launch Browser using WebDriver?

To launch Browser using WebDriver, following syntax is followed -

WebDriver driver = new InternetExplorerDriver();

WebDriver driver = new ChromeDriver();

WebDriver driver = new FirefoxDriver();

### 68. What is the difference between Asset Management and Configuration Management?

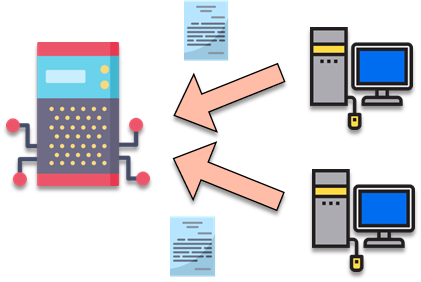
Differences between Configuration Management and Asset Management are:

|  |  |
| --- | --- |
| Configuration Management | Asset Management |
| Operational Relationships. | Incidental relationships only. |
| Maintains troubleshooting data. | Maintains taxes data. |
| Everything we deploy is scope. | Everything we own is scope. |
| Deployment to retirement - lifecycle. | Purchase to disposal - lifecycle. |
| Operations - main concern. | Finances - main concern. |
| ITIL processes from interfacing. | Leasing and purchasing from interfacing. |

## DevOps Interview Questions for Configuration Management — Chef, Puppet, Ansible

### 69. Why are SSL certificates used in Chef?

* SSL certificates are used between the Chef server and the client to ensure that each node has access to the right data.
* Every node has a private and public key pair. The public key is stored at the Chef server.
* When an SSL certificate is sent to the server, it will contain the private key of the node.
* The server compares this against the public key in order to identify the node and give the node access to the required data.



### 70. Which of the following commands would you use to stop or disable the 'httpd' service when the system boots?

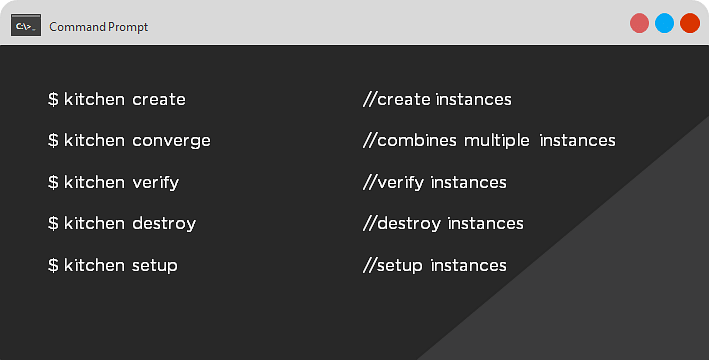
1. # systemctl disable httpd.service
2. # system disable httpd.service
3. # system disable httpd
4. # systemctl disable httpd.service

The correct answer is A) # systemctl disable httpd.service

### 71. What is Test Kitchen in Chef?

Test Kitchen is a command-line tool in Chef that spins up an instance and tests the cookbook on it before deploying it on the actual nodes.

Here are the most commonly used kitchen commands:



### 72. How does chef-apply differ from chef-client?

* chef-apply is run on the client system.  
    
  chef-apply applies the recipe mentioned in the command on the client system.  
    
  $ chef-apply recipe\_name.rb
* chef-client is also run on the client system.  
    
  chef-client applies all the cookbooks in your server's run list to the client system.  
    
  $ knife chef-client

### 73. What is the command to sign the requested certificates?

* For Puppet version 2.7:  
    
  # puppetca –sign hostname-of-agent  
    
  Example:  
    
  # puppetca –sign ChefAgent  
    
  # puppetca sign hostname-of-agent  
    
  Example:  
    
  # puppetca sign ChefAgent
* For Puppet version 2.7:  
    
  # puppetca –sign hostname-of-agent  
    
  Example:  
    
  # puppetca –sign ChefAgent  
    
  # puppetca sign hostname-of-agent  
    
  Example:  
    
  # puppetca sign ChefAgent

### 74. Which open source or community tools do you use to make Puppet more powerful?

* Changes in the configuration are tracked using Jira, and further maintenance is done through internal procedures.
* Version control takes the support of Git and Puppet's code manager app.
* The changes are also passed through Jenkin's continuous integration pipeline.

### 75. What are the resources in Puppet?

* Resources are the basic units of any configuration management tool.
* These are the features of a node, like their software packages or services.
* A resource declaration, written in a catalog, describes the action to be performed on or with the resource.
* When the catalog is executed, it sets the node to the desired state.

### 76. What is a class in Puppet?

Classes are named blocks in your manifest that configure various functionalities of the node, such as services, files, and packages.

The classes are added to a node's catalog and are executed only when explicitly invoked.

Class apache (String $version = ‘latest’) {

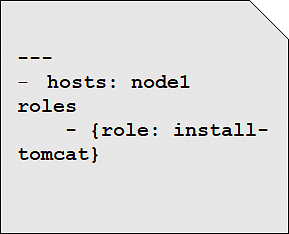
package{

‘httpd’: ensure => $version,

before => File[‘/etc/httpd.conf’],}

### 77. What is an Ansible role?

An Ansible role is an independent block of tasks, variables, files, and templates embedded inside a playbook.



This playbook installs tomcat on node1.

### 78. When should I use '{{ }}'?

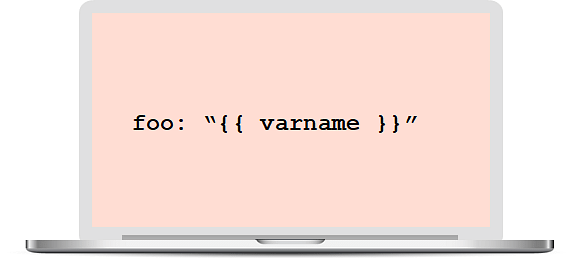
Always use {{}} for variables, unless you have a conditional statement, such as "when: …". This is because conditional statements are run through Jinja, which resolves the expressions.

 For example:

      echo “This prints the value of {{foo}}”

      when : foo is defined

Using brackets makes it simpler to distinguish between strings and undefined variables.



This also ensures that Ansible doesn't recognize the line as a dictionary declaration.

### 79. What is the best way to make content reusable/redistributable?

There are three ways to make content reusable or redistributable in Ansible:

* Roles are used to managing tasks in a playbook. They can be easily shared via Ansible Galaxy.
* "include" is used to add a submodule or another file to a playbook. This means a code written once can be added to multiple playbooks.
* "import" is an improvement of "include," which ensures that a file is added only once. This is helpful when a line is run recursively.

### 80. How is Ansible different from Puppet?

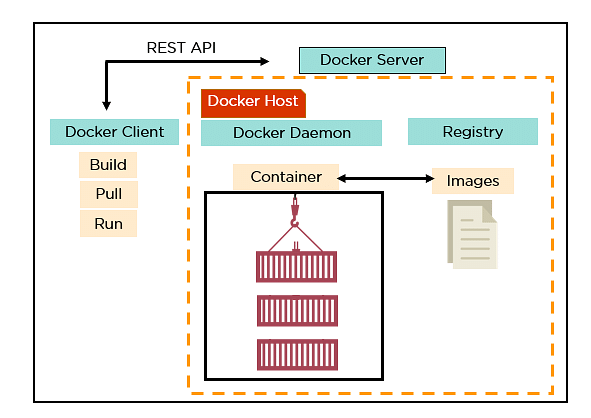
|  |  |
| --- | --- |
| Ansible | Puppet |
| Easy agentless installation | Agent-based installation |
| Based on Python | Based on Ruby |
| Configuration files are written in YAML | Configuration files are written in DSL |
| No support for Windows | Support for all popular OS's |

We will now look at some DevOps interview questions on contanerization.

## DevOps Interview Questions on Containerization

### 81. Explain the architecture of Docker.

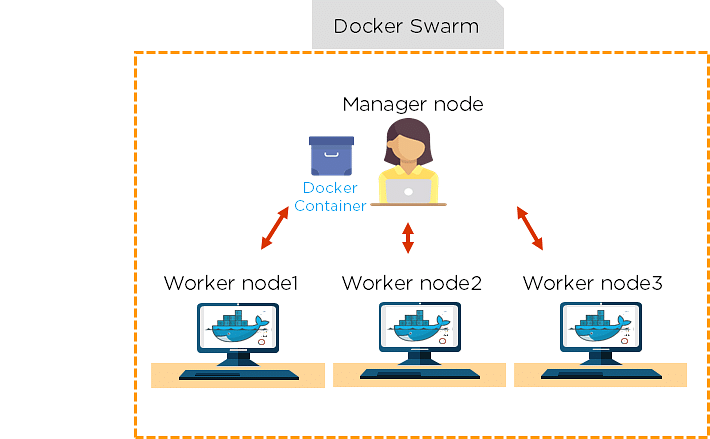
* [Docker](https://www.simplilearn.com/tutorials/docker-tutorial/what-is-docker" \o "Docker" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) uses a client-server architecture.
* Docker Client is a service that runs a command. The command is translated using the REST API and is sent to the Docker Daemon (server).
* Docker Daemon accepts the request and interacts with the operating system to build Docker images and run Docker containers.
* A Docker image is a template of instructions, which is used to create containers.
* [Docker container](https://www.simplilearn.com/tutorials/docker-tutorial/what-is-docker-container" \o "Docker container" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) is an executable package of an application and its dependencies together.
* Docker registry is a service to host and distribute Docker images among users.



### 82. What are the advantages of Docker over virtual machines?

|  |  |  |
| --- | --- | --- |
| Criteria | Virtual Machine | Docker |
| Memory space | Occupies a lot of memory space | Docker containers occupy less space |
| Boot-up time | Long boot-up time | Short boot-up time |
| Performance | Running multiple virtual machines leads to unstable performance | Containers have a better performance, as they are hosted in a single Docker engine |
| Scaling | Difficult to scale up | Easy to scale up |
| Efficiency | Low efficiency | High efficiency |
| Portability | Compatibility issues while porting across different platforms | Easily portable across different platforms |
| Space allocation | Data volumes cannot be shared | Data volumes are shared and used again across multiple containers |

### 83. How do we share Docker containers with different nodes?



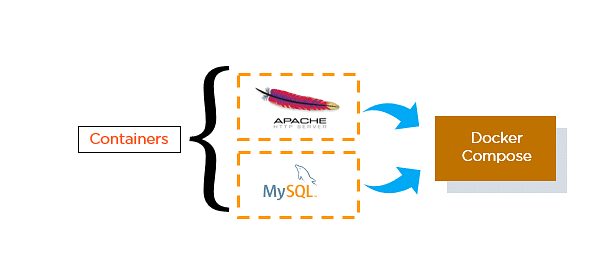
* It is possible to share Docker containers on different nodes with [Docker Swarm](https://www.simplilearn.com/tutorials/docker-tutorial/docker-swarm" \o "Docker Swarm" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank).
* Docker Swarm is a tool that allows IT administrators and developers to create and manage a cluster of swarm nodes within the Docker platform.
* A swarm consists of two types of nodes: a manager node and a worker node.

### 84. What are the commands used to create a Docker swarm?

* Create a swarm where you want to run your manager node.  
    
  Docker swarm init --advertise-addr <MANAGER-IP>
* Once you've created a swarm on your manager node, you can add worker nodes to your swarm.
* When a node is initialized as a manager, it immediately creates a token. In order to create a worker node, the following command (token) should be executed on the host machine of a worker node.  
    
  docker swarm join \ --token SWMTKN-1-49nj1cmql0jkz5s954yi3oex3nedyz0fb0xx14ie39trti4wxv-8vxv8rssmk743ojnwacrr2e7c \ 192.168.99.100:2377

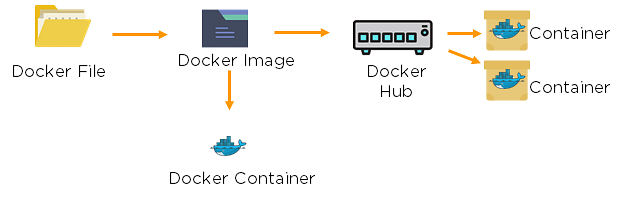
### 85. How do you run multiple containers using a single service?

* It is possible to run multiple containers as a single service with Docker Compose.
* Here, each container runs in isolation but can interact with each other.
* All Docker Compose files are YAML files.



### 86. What is a Dockerfile used for?

* A Dockerfile is used for creating Docker images using the build command.
* With a Docker image, any user can run the code to create Docker containers.
* Once a Docker image is built, it's uploaded in a Docker registry.
* From the Docker registry, users can get the Docker image and build new containers whenever they want.



### 87. Explain the differences between Docker images and Docker containers.

|  |  |
| --- | --- |
| Docker Images | Docker Container |
| Docker images are templates of Docker containers | Containers are runtime instances of a Docker image |
| An image is built using a Dockerfile | Containers are created using Docker images |
| It is stored in a Docker repository or a Docker hub | They are stored in the Docker daemon |
| The image layer is a read-only filesystem | Every container layer is a read-write filesystem |

### 88. Instead of YAML, what can you use as an alternate file for building Docker compose?

To build a Docker compose, a user can use a JSON file instead of YAML. In case a user wants to use a JSON file, he/she should specify the filename as given:

Docker-compose -f Docker-compose.json up

### 89. How do you create a Docker container?

Task: Create a MySQL Docker container

A user can either build a Docker image or pull an existing Docker image (like MySQL) from Docker Hub.

Now, Docker creates a new container MySQL from the existing Docker image. Simultaneously, the container layer of the read-write filesystem is also created on top of the image layer.

* Command to create a Docker container: Docker run -t –i MySQL
* Command to list down the running containers: Docker ps

### 90. What is the difference between a registry and a repository?

|  |  |
| --- | --- |
| Registry | Repository |
| A Docker registry is an open-source server-side service used for hosting and distributing Docker images | The repository is a collection of multiple versions of Docker images |
| In a registry, a user can distinguish between Docker images with their tag names | It is stored in a Docker registry |
| Docker also has its own default registry called Docker Hub | It has two types: public and private repositories |

### 91. What are the cloud platforms that support Docker?

The following are the cloud platforms that Docker runs on:

* [Amazon Web Services](https://www.simplilearn.com/tutorials/aws-tutorial/what-is-aws" \o "Amazon Web Services" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)
* [Microsoft Azure](https://www.simplilearn.com/tutorials/azure-tutorial/what-is-azure" \o "Microsoft Azure" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank)
* Google Cloud Platform
* Rackspace

### 92. What is the purpose of the expose and publish commands in Docker?

#### Expose

* Expose is an instruction used in Dockerfile.
* It is used to expose ports within a Docker network.
* It is a documenting instruction used at the time of building an image and running a container.
* Expose is the command used in Docker.
* Example: Expose 8080

#### Publish

* Publish is used in a Docker run command.
* It can be used outside a Docker environment.
* It is used to map a host port to a running container port.
* --publish or –p is the command used in Docker.
* Example: docker run –d –p 0.0.0.80:80

Now, let's have a look at the DevOps interview questions for continuous monitoring.

## DevOps Interview Questions for Continuous Monitoring

### 93. How does Nagios help in the continuous monitoring of systems, applications, and services?

Nagios enables server monitoring and the ability to check if they are sufficiently utilized or if any task failures need to be addressed.

* Verifies the status of the servers and services
* Inspects the health of your infrastructure
* Checks if applications are working correctly and web servers are reachable

### 94. How does Nagios help in the continuous monitoring of systems, applications, and services?

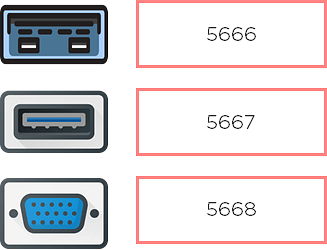
### 95. What do you mean by Nagios Remote Plugin Executor (NPRE) of Nagios?

Nagios Remote Plugin Executor (NPRE) enables you to execute Nagios plugins on Linux/Unix machines. You can monitor remote machine metrics (disk usage, CPU load, etc.)

* The check\_npre plugin that resides on the local monitoring machine
* The NPRE daemon that runs on the remote Linux/Unix machine

### 96. What are the port numbers that Nagios uses for monitoring purposes?

Usually, Nagios uses the following port numbers for monitoring:



### 97. What are active and passive checks in Nagios?

Nagios is capable of monitoring hosts and services in two ways:

#### Actively

* Active checks are initiated as a result of the Nagios process
* Active checks are regularly scheduled

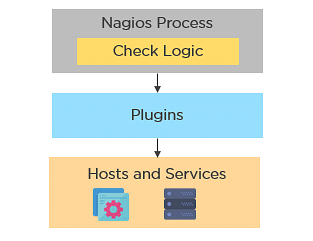
#### Passively

* Passive checks are initiated and performed through external applications/processes
* Passive checks results are submitted to Nagios for processing

### 98. What are active and passive checks in Nagios?

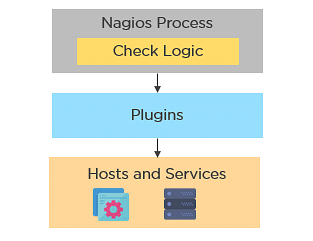
Active Checks:

* The check logic in the Nagios daemon initiates active checks.
* Nagios will execute a plugin and pass the information on what needs to be checked.
* The plugin will then check the operational state of the host or service, and report results back to the Nagios daemon.
* It will process the results of the host or service check and send notifications.



Passive Checks:

* In passive checks, an external application checks the status of a host or service.
* It writes the results of the check to the external command file.
* Nagios reads the external command file and places the results of all passive checks into a queue for later processing.
* Nagios may send out notifications, log alerts, etc. depending on the check result information.



Are you skilled enough for your next role as a DevOps Engineer? Well try answering these [DevOps Practice Test Questions](https://www.simplilearn.com/devops-exam-questions-free-practice-test" \o "DevOps Practice Test Questions" \t "https://www.simplilearn.com/tutorials/devops-tutorial/_blank) and find out yourself.

### 99. Explain the main configuration file and its location in Nagios.

The main configuration file consists of several directives that affect how Nagios operates. The Nagios process and the CGIs read the config file.

A sample main configuration file will be placed into your settings directory:

/usr/local/Nagios/etc/resource.cfg

### 100. What is the Nagios Network Analyzer?

* It provides an in-depth look at all network traffic sources and security threats.
* It provides a central view of your network traffic and bandwidth data.
* It allows system admins to gather high-level information on the health of the network.
* It enables you to be proactive in resolving outages, abnormal behavior, and threats before they affect critical business processes.

### 101. What are the benefits of HTTP and SSL certificate monitoring with Nagios?

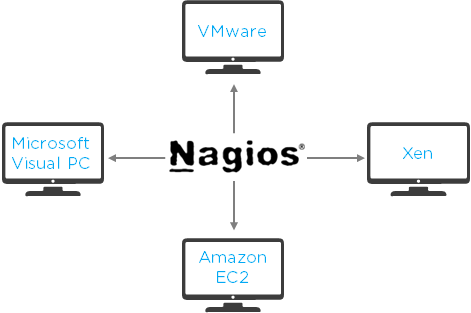
HTTP certificate monitoring

* Increased server, services, and application availability.
* Fast detection of network outages and protocol failures.
* Enables web transaction and web server performance monitoring.

SSL certificate monitoring

* Increased website availability.
* Frequent application availability.
* It provides increased security.

### 102. Explain virtualization with Nagios.



Nagios can run on different virtualization platforms, like VMware, Microsoft Visual PC, Xen, Amazon EC2, etc.

* Provides the capabilities to monitor an assortment of metrics on different platforms
* Ensures quick detection of service and application failures
* Has the ability to monitor the following metrics:
* CPU Usage
* Memory
* Networking
* VM status
* Reduced administrative overhead

### 103. Name the three variables that affect recursion and inheritance in Nagios.

name - Template name that can be referenced in other object definitions so it can inherit the object's properties/variables.

use - Here, you specify the name of the template object that you

want to inherit properties/variables from.

register - This variable indicates whether or not the object definition

should be registered with Nagios.

define someobjecttype{

              object-specific variables ….

              name template\_name

              use name\_of\_template

              register [0/1]

              }

### 104. Why is Nagios said to be object-oriented?



Using the object configuration format, you can create object definitions that inherit properties from other object definitions. Hence, Nagios is known as object-oriented.

Types of Objects:

* Services
* Hosts
* Commands
* Time Periods

### 105. Explain what state stalking is in Nagios.

* State stalking is used for logging purposes in Nagios.
* When stalking is enabled for a particular host or service, Nagios will watch that host or service very carefully.
* It will log any changes it sees in the output of check results.
* This helps in the analysis of log files.