1. What is AWS?

AWS (Amazon Web Services) is a comprehensive, evolving cloud computing platform provided by Amazon that includes a mixture of infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS) offering.

1. Difference between on premise and cloud?

On-premises software is typically locally installed on the computers and servers your business uses. Cloud software, on the other hand, is hosted on a vendor's server and can be easily accessed through a web browser.

1. What is a VPC?

A VPC is a virtual network that closes resembles a traditional network that you'd operate in your own data center. After you create a VPC, you can add subnets.

1. What are subnets?

A subnet is a range of IP addresses in your VPC. A subnet must reside in a single Availability Zone. After you add subnets, you can deploy AWS resources in your VPC.

1. What is internet gateway?

An internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between your VPC and the internet.

1. What is NAT gateway?

NAT Gateway is a highly available AWS managed service that makes it easy to connect to the Internet from instances within a private subnet in an Amazon Virtual Private Cloud (Amazon VPC).

1. What is difference between public and private subnet?

The instances in the public subnet can send outbound traffic directly to the internet, whereas the instances in the private subnet can't. Instead, the instances in the private subnet can access the internet by using a network address translation (NAT) gateway that resides in the public subnet.

1. What are route tables?

A route table contains a set of rules, called routes, that determine where network traffic from your subnet or gateway is directed.

1. What is an ec2?

An Amazon EC2 instance is a virtual server in Amazon's Elastic Compute Cloud (EC2) for running applications on the Amazon Web Services (AWS) infrastructure.

1. What is security group?

A security group controls the traffic that can reach and leave the resources that it is associated with. For example, after you associate a security group with an EC2 instance, it controls the inbound and outbound traffic for the instance.

1. What is terraform?

HashiCorp Terraform is an infrastructure as code tool that lets you define both cloud and on-prem resources in human-readable configuration files that you can version, reuse, and share.

1. How do you debug connectivity to EC2 instance?

* Verify the username for your instance.
* Verify that your security group rules allow traffic. (port 22 should be open)
* Verify that your instance is ready.

1. What is Devops?

DevOps is a set of practices that combines [software development](https://en.wikipedia.org/wiki/Software_development) (*Dev*) and [IT operations](https://en.wikipedia.org/wiki/IT_operations) (*Ops*). It aims to shorten the [systems development life cycle](https://en.wikipedia.org/wiki/Systems_development_life_cycle) and provide [continuous delivery](https://en.wikipedia.org/wiki/Continuous_delivery) with high [software quality](https://en.wikipedia.org/wiki/Software_quality).

1. What is Git?

Git is a DevOps tool used for source code management. Git is used to tracking changes in the source code, enabling multiple developers to work together on non-linear development

1. What is AWS IAM?

AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources. You use IAM to control who is authenticated (signed in) and authorized (has permissions) to use resources.

1. How do I connect two VPCs?

The simplest way to connect two VPCs is to use VPC Peering. In this setup, a connection enables full bidirectional connectivity between the VPCs. This peering connection is used to route traffic between the VPCs. VPCs across accounts and AWS Regions can also be peered together.

1. What is VPC peering?

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses.

1. Terraform commands

**terraform version** - prints the version of terraform

**terraform init** - The terraform init command is used to initialize a working directory containing Terraform configuration files. This is the first command that should be run after writing a new Terraform configuration or cloning an existing one from version control.

**terraform plan** - The terraform plan command creates an execution plan, which lets you preview the changes that Terraform plans to make to your infrastructure.

**terraform apply** - The terraform apply command executes the actions proposed in a Terraform plan.

**terraform destroy** - The terraform destroy command is a convenient way to destroy all remote objects managed by a particular Terraform configuration.

**terraform refresh** - The terraform refresh command reads the current settings from all managed remote objects and updates the Terraform state to match.

**terraform validate** - The terraform validate command validates the configuration files in a directory, referring only to the configuration and not accessing any remote services such as remote state, provider APIs, etc.

**terraform state** - The terraform state list command is used to list resources within a [terraform state](https://www.terraform.io/language/state).

**terraform output** - The terraform output command is used to extract the value of an output variable from the state file.

**terraform modules** - A Terraform module **allows you to create logical abstraction on the top of some resource set**. In other words, a module allows you to group resources together and reuse this group later, possibly many times.

**terraform state list** - The terraform state list command is used to list resources within a [terraform state](https://www.terraform.io/language/state).

**What is user\_data in terraform** - AWS user\_data is the set of commands/data you can provide to an instance at launch time.

For example, if you are launching an ec2 instance and want to have docker installed on the newly launched ec2, then you can provide set of bash commands in the user\_data field of aws ec2 config page.

**What is the extension of terraform** - “.tf”

**Terraform works on which language** - “HCL (HashiCorp Configuration Language)”

1. Steps to create an EC2 machine:
2. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
3. Choose Launch Instance.
4. Choose an Amazon Machine Image (AMI), find an Amazon Linux 2 AMI at the top of the list and choose Select.
5. Choose an Instance Type, choose Next: Configure Instance Details.
6. Configure Instance Details, provide the following information:

* Leave Number of instances at one.
* Leave Purchasing option at the default setting.
* For Network, choose the entry for the same VPC that you noted when you created your EFS file system in Step 1: Create your Amazon EFS file system.
* For Subnet, choose a default subnet in any Availability Zone.
* For File systems, make sure that the EFS file system that you created in Step 1: Create your Amazon EFS file system is selected. The path shown next to the file system ID is the mount point that the EC2 instance will use, which you can change.
* The User data automatically includes the commands for mounting your Amazon EFS file system.

1. Choose Next: Add Storage.
2. Choose Next: Add Tags.
3. Name your instance and choose Next: Configure Security Group.
4. Configure Security Group, set Assign a security group to Select an existing security group. Choose the default security group to make sure that it can access your EFS file system.
5. Choose Review and Launch.
6. Choose Launch.
7. Select the check box for the key pair that you created, and then choose Launch Instances.
8. Once the EC2 instance is created and becomes available, it will be mounted to your EFS file system. At this point, you will be able to transfer files to your EFS file system.
9. Steps to create VPC:
10. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
11. In the navigation pane, choose Your VPCs, Create VPC.
12. Under Resources to create, choose VPC only.
13. Specify the following VPC details as needed.

* **Name tag**: Optionally provide a name for your VPC. Doing so creates a tag with a key of Name and the value that you specify.
* **IPv4 CIDR block**: Specify an IPv4 CIDR block (or IP address range) for your VPC.
* **IPv6 CIDR block**: Optionally associate an IPv6 CIDR block with your VPC.
* **Tags**: Add optional tags on the VPC. A tag is a label that you assign to an AWS resource.

1. Choose **Create VPC**.
2. Steps to create subnets:

To add a new subnet to your VPC, you must specify an IPv4 CIDR block for the subnet from the range of your VPC. You can specify the Availability Zone in which you want the subnet to reside. You can have multiple subnets in the same Availability Zone.

To add a subnet to your VPC.

* Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
* In the navigation pane, choose Subnets.
* Choose Create subnet.
* For VPC ID: Choose the VPC for the subnet.
* (Optional) For Subnet name, enter a name for for your subnet. Doing so creates a tag with a key of Name and the value that you specify.
* For Availability Zone, you can choose a Zone for your subnet, or leave the default No Preference to let AWS choose one for you.
* If the subnet should be an IPv6-only subnet, choose IPv6-only. This option is only available if the VPC has an associated IPv6 CIDR block. If you choose this option, you can't associate an IPv4 CIDR block with the subnet.
* For IPv4 CIDR block, enter an IPv4 CIDR block for your subnet. For example, 10.0.1.0/24. For more information, see VPC sizing for IPv4. If you chose IPv6-only, this option is unavailable.
* For IPv6 CIDR block, choose Custom IPv6 CIDR and specify the hexadecimal pair value (for example, 00). This option is available only if the VPC has an associated IPv6 CIDR block.
* Choose Create subnet.

1. What is Elastic Load Balancing?

* Elastic Load Balancing automatically distributes your incoming traffic across multiple targets, such as EC2 instances, containers, and IP addresses, in one or more Availability Zones.
* It monitors the health of its registered targets, and routes traffic only to the healthy targets. Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

1. What is launch configuration in AWS?

* A launch configuration is **a template that an EC2 Auto Scaling group uses to launch EC2 instances**.
* When you create a launch configuration, you specify information for the instances such as the ID of the Amazon Machine Image (AMI), the instance type, a key pair, one or more security groups, and a block device mapping.

1. Git commands: <https://www.atlassian.com/git/glossary>  
     
   Reference with screenshots (git commands): <https://www.javatpoint.com/git-commands>
2. What is autoscaling group?

* An Auto Scaling group contains a collection of EC2 instances that are treated as a logical grouping for the purposes of automatic scaling and management.
* An Auto Scaling group also enables you to use Amazon EC2 Auto Scaling features such as health check replacements and scaling policies.

1. Create autoscaling group –

<https://docs.aws.amazon.com/codedeploy/latest/userguide/tutorials-auto-scaling-group-create-auto-scaling-group.html>

27) What is a resource in AWS?

In AWS, a resource is **an entity that you can work with**. Examples include an Amazon EC2 instance, an AWS CloudFormation stack, or an Amazon S3 bucket.

28) What is infrastructure as code?

Infrastructure as Code (IaC) is **the managing and provisioning of infrastructure through code instead of through manual processes**. With IaC, configuration files are created that contain your infrastructure specifications, which makes it easier to edit and distribute configurations.

29) What is a yaml file?

YAML is a digestible data serialization language often used to create configuration files with any programming language.

1. What is CICD?

A CI/CD (Continuous Integration and Continuous Delivery) pipeline **automates the process of software delivery**. It builds code, runs tests, and helps you to safely deploy a new version of the software. CI/CD pipeline reduces manual errors, provides feedback to developers, and allows fast product iterations.

1. What is elastic or static IP address?

An Elastic IP address is a static public IPv4 address associated with your AWS account in a specific Region. Unlike an auto-assigned public IP address, an Elastic IP address is preserved after you stop and start your instance in a virtual private cloud (VPC).

1. What is ssh-client?

An SSH client is **an application you install on the computer which you will use to connect to another computer or a server**. The client uses the provided remote host information to initiate the connection and if the credentials are verified, establishes the encrypted connection.

1. What is Git?

Git is **a DevOps tool used for source code management**. It is a free and open-source version control system used to handle small to very large projects efficiently. Git is used to tracking changes in the source code, enabling multiple developers to work together on non-linear development.

1. What is an API?

API stands for “Application Programming Interface.” An API is a software intermediary that allows two applications to talk to each other. In other words, an API is the messenger that delivers your request to the provider that you're requesting it from and then delivers the response back to you.

1. What is postman?

Postman is an application used for API testing. It is an HTTP client that tests HTTP requests, utilizing a graphical user interface, through which we obtain different types of responses that need to be subsequently validated.

1. What are different kinds of methods available in postman?

Postman offers many endpoint interaction methods. The following are some of the most used, including their functions:

* GET: Obtain information
* POST: Add information
* PUT: Replace information
* PATCH: Update certain information
* DELETE: Delete information.

1. How to install plugins in visual studio code?
   * + Open Visual Studio Code.
     + Select. (Extensions) from the left pane.
     + Select. from the top-right on the extensions pane.
     + Search for and select Power Platform Tools.
     + Select Install.
     + Verify the extension is installed successfully from the status messages.
2. What is docker?

Docker is **an open-source containerization platform**. It enables developers to package applications into containers standardized executable components combining application source code with the operating system (OS) libraries and dependencies required to run that code in any environment.

1. What is **Docker image?**

The Docker image help to create Docker containers. You can create the Docker image with the build command. Due to this, it creates a container that starts when it begins to run. Every docker images are stored in the Docker registry.

1. **What command should you run to see all running container in Docker?**

docker ps

1. **Write the command to stop the docker container?**

sudo docker stop “container name”

1. **What is the command to run the image as a container?**

sudo docker run -i -t alpine /bin/bash

1. What is Docker container?

Containers are **packages of an application that contain all the necessary elements to run in any environment**.

Also, a container is a runnable instance of an image. You can create, start, stop, move, or delete a container using the Docker API or CLI.

1. **What the states of Docker container?**

Important states of Docker container are:

* Running
* Paused
* Restarting
* Exited

1. What is docker registry?

A Docker registry is **a storage and distribution system for named Docker images**.

The same image might have multiple different versions, identified by their tags.

A Docker registry is organized into Docker repositories, where a repository holds all the versions of a specific image.

1. What is docker compose?

[(https://www.tutorialspoint.com/docker/docker\_compose.htm)]((https:/www.tutorialspoint.com/docker/docker_compose.htm))

**Docker Compose** is used to run multiple containers as a single service. For example, suppose you had an application which required NGNIX and MySQL, you could create one file which would start both the containers as a service without the need to start each one separately.

1. What is virtualization?

* Virtualization is the process of running a virtual instance of a computer system in a layer abstracted from the actual hardware.
* Most commonly, it refers to running multiple operating systems on a computer system simultaneously.
* It also allows for isolation, keeping programs running inside of a virtual machine safe from the processes taking place in another virtual machine on the same host.

1. What is Jenkins?

* Jenkins is an open-source automation tool written in Java with plugins built for Continuous Integration purposes.
* Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project and making it easier for users to obtain a fresh build.
* It also allows you to continuously deliver your software by integrating with many testing and deployment technologies.