      Jenkins

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. If we have 5 stages in a Jenkins pipeline and 5th stage having syntax error then what will happen if we run the pipeline.

2. How are you integrating the SonarQube with the Jenkins server?

3. How were you authenticating Jenkins to push docker image to registry?

4. Jenkins is failing to push a Docker image to the registry. How do you troubleshoot?

5. Your CI/CD pipeline has failed in jenkins. How do you investigate?

6. How do you store sensitive information like passwords in jenkins?

7. Explain how you would set up a multi-branch Jenkins pipeline for a GitHub repository.

8. How would you implement dynamic stages in a Jenkinsfile based on environment variables?

9. Terraform scripts for creating AWS services Jenkinsfile EKS and On Prem Kubernetes cluster upgrade steps.

10.Jenkins pipeline setup Kubernetes If a pod is getting restarted constantly, what steps are you going to follow?

11. A Jenkins pipeline is randomly failing at the deployment stage to EKS. Logs show timeouts during kubectl apply.

12. you are using jenkins server as open source s/w tool like in aws service which  service is available to implement CICD Pipeline

13. different plugins for ci/cd in jenkins using aws platform

14. Why do you want to use Argo CD over Jenkins?

15. Explain the CI/CD workflow you follow and the kind of pipeline you use. How do you define and invoke pipelines in Jenkins?

16. What are shared libraries in Jenkins, and how are they written and defined?

17. What kind of applications do you deploy using Jenkins pipelines, and what deployment tools do you use?

18. If the Jenkins pipeline runs but the build doesn’t happen, what possible issues could be causing it?

20. What is the purpose of a webhook, and how is it used in a CI/CD pipeline?

21. Describe your typical deployment flow and CI/CD workflow.

22. What stages do you define in your Jenkins pipeline, and how do you ensure full quality checks during deployment?

23. How do you use Jenkins shared libraries? Explain their typical structure and how they are integrated into your Jenkinsfiles.

24. you want to store the secrets in Jenkins pipeline , how will you do it

25. How does an AWS CodePipeline differ from a Jenkins pipeline? Can you give an example of when you would choose one over the other?

26. Can you explain how you would deploy a Kubernetes application using Jenkins? What plugins or tools would you use?

27.  Jenkins jobs are randomly failing at the artifact upload step. What layers would you check?

28. How does authentication happen in Jenkins pipeline to use aws with particular login, if you have 1 logout?

29. Please explain how did you setup sonarqube in your CI jenkins pipeline, what are the quality gates and how did you set the threshold checks for code coverage leveraging the test reports   created by developer.

30. Have you faced any memory issue in Jenkins pipeline? how would you troubleshoot?

31. Can you explain how you would deploy a Kubernetes application using Jenkins? What plugins or tools would you use?

32. Jenkins jobs are randomly failing at the artifact upload step. What layers would you check?

33. How does authentication happen in Jenkins pipeline to use aws with particular login, if you have 1 logout?

34. What is publisher in Jenkins?

35. How can you create a new copy of an existing Jenkins job ?

36. Have you faced any memory issue in Jenkins pipeline? how would you troubleshoot?

37. How about your experience developing CI/CD pipeline and utilizing tools such as Docker, Grafana and Prometheous. Share a particular project where these skills were critical.

38.  Suppose you have created a CI/CD process. After building the image, manual intervention is required. How would you configure it, and where ?

39. CI/CD pipeline needs rollback capability. How would you implement it?

40. How are you integrating the SonarQube with the Jenkins server?

41. How are you integrating the SonarQube with the Jenkins server?

42. How are you integrating the SonarQube with the Jenkins server?

43. If the code passes in a Jenkins pipeline and gets deployed but later encounters issues in production during testing, how can we resolve the issue?

44. Can you explain the steps involved in creating a CICD pipeline for deploying a Spring Boot application to ECS?

45. How azure key vault is integrated in cicd?

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

                                                        TERRAFORM

1. Write a Terraform code to create multiple S3 buckets?

2. How would you handle multiple environments (development, staging, production) using Terraform?

3. Difference between terraform and ansible?

4. What is the difference between resource and data in Terraform, and how are they used?

5. What are the most commonly used 10 Terraform commands and their use cases?

6. What are the data types in Terraform?

7. What are Terraform modules, and how do they function in an environment?

8. What is the Terraform lifecycle, and how does it work?

9. If the Terraform state file is deleted but existing resources need to be managed without recreation, how can we achieve this?

10. If the Terraform state file is deleted, can we recreate or modify existing resources? If so, how?

11. How can multiple EC2 instances be created in Terraform without using the count parameter?

12. What are local and variable definitions in Terraform?

13. If a child module is used for VPC and EC2 instances in Terraform, how can we deploy the EC2 instance in the VPC’s subnet?

14. What is the difference between a resource block and a data block in Terraform?

15. What command is used to initialize a Terraform working directory?

16. Blocks of Terraform?

17. Null resource in terraform?

18. Command to Stop and Delete the Docker container?

19. what is terraform taint

20. terragrunt terraform

21. Terraform workspace

22. Current Vs Desired state in terraform

23. If someone Deleted the terraform manage resources then how do we recover them?

24. Write a terraform main.tf file to create storage account?

25. What happens if your state file is accidentally deleted?

26. What happens if multiple team members run terraform apply simultaneously?

27. What happens if a resource fails halfway through a terraform apply?

28. What happens if a resource fails halfway through a terraform apply?

29. What happens if terraform plan shows no changes but infrastructure was modified outside Terraform?

30. What happens if a provider API changes between Terraform versions?

31. What happens if you have circular dependencies in your Terraform modules?

32. What happens if you exceed AWS service quotas during deployment?

33. What happens if you lose access to the remote backend storing your state?

34. AWS event bridge creation and setup via terraform?

35. What are the provisioners available in Terraform and can you explain the use cases?

36. I have created an EC2 instance through Terraform. I don't have a backup of the Terraform state file, it is not in the remote state and locally not available. Now when I do apply, what can I do?

37. What is the difference between a map of objects in Terraform and how can you write an example?

38. What is Terraform and how do you use terraform in your project and what all resources have you provisioned

Terraform questions

Intermediate to Advanced Terraform Questions

1. What is the purpose of the terraform state command?

2. How do you manage multiple environments (e.g., dev, staging, production) using Terraform?

3. Explain the difference between terraform plan and terraform apply.

4. What are Terraform modules, and how do you create and use them?

5. How can you handle sensitive data in Terraform?

6. What is the purpose of the terraform output command?

7. How do you implement version control for your Terraform configurations?

8. What is the difference between terraform import and terraform apply?

9. How do you manage dependencies between resources in Terraform?

10. What are the benefits of using remote state storage, and how do you configure it?

11. Explain the concept of "terraform workspaces." How do they differ from modules?

12. How can you use Terraform to manage resources across multiple cloud providers?

13. What is the purpose of the terraform fmt command?

14. How do you handle resource drift in Terraform?

15. What are the best practices for structuring Terraform code?

16. How do you use Terraform with CI/CD pipelines?

17. What is the purpose of the terraform validate command?

18. How can you use Terraform to create a VPC with multiple subnets?

19. What is the difference between count and for\_each in resource definitions?

20. How do you implement conditional resource creation in Terraform?

21. What are Terraform providers, and how do you create a custom provider?

22. How do you manage secrets in Terraform when using cloud providers?

23. What is the purpose of the terraform taint command?

24. How do you roll back changes in Terraform?

25. What is the significance of the terraform graph command?

26. How do you use Terraform to manage IAM roles and policies in AWS?

27. What are the differences between terraform destroy and terraform apply -destroy?

28. How do you handle resource dependencies when using modules?

29. What is the purpose of the terraform workspace select command?

30. How do you implement tagging for resources in Terraform?

31. What are the implications of using terraform apply without a plan?

32. How do you use Terraform to provision Kubernetes resources?

33. What is the purpose of the terraform state rm command?

34. How do you manage state file locking in Terraform?

35. What are the differences between terraform plan -out and terraform plan?

36. How do you use Terraform to create a load balancer with multiple target groups?

37. What is the purpose of the terraform console command?

38. How do you handle versioning of Terraform modules?

39. What are the benefits of using Terraform Cloud or Terraform Enterprise?

40. How do you implement lifecycle rules for resources in Terraform?

41. What is the purpose of the terraform apply -auto-approve flag?

42. How do you use Terraform to manage DNS records in Route 53?

43. What are the differences between terraform apply and terraform apply -refresh=true?

44. How do you implement a blue-green deployment strategy using Terraform?

45. What is the purpose of the terraform workspace new command?

46. How do you use Terraform to provision serverless functions in AWS?

47. What are the implications of using terraform apply in a production environment?

48. How do you handle multi-cloud deployments with Terraform?

49. What is the purpose of the terraform state push command?

50. How do you use Terraform to manage network security groups in Azure?

51. What is the purpose of the terraform state pull command?

52. How do you implement a canary deployment strategy using Terraform?

53. What are the implications of using terraform apply with the -target option?

54. How do you manage Terraform configurations for a microservices architecture?

55. What is the role of the terraform plan output in a collaborative environment?

56. How do you use Terraform to provision a serverless database?

57. What are the best practices for managing Terraform state files in a team?

58. How do you implement resource dependencies using the depends\_on argument in Terraform?

59. What is the purpose of the terraform workspace list command?

60. How do you use Terraform to create and manage Elastic Beanstalk applications?

61. What are the differences between terraform apply and terraform apply -refresh=false?

62. How do you implement a rollback strategy in Terraform?

63. What is the significance of the terraform state mv command?

64. How do you use Terraform to provision a Redis cluster?

65. What are the best practices for writing reusable Terraform modules?

66. How do you manage Terraform configurations for different teams within an organization?

67. What is the purpose of the terraform output -json command?

68. How do you use Terraform to provision a VPN connection?

69. What are the implications of using terraform apply with the -lock=false option?

70. How do you implement monitoring and alerting for Terraform-managed resources?

71. What is the purpose of the terraform state list command?

72. How do you use Terraform to manage Kubernetes namespaces and resources?

73. What are the differences between terraform plan and terraform apply in terms of execution?

74. How do you handle state file encryption in Terraform?

75. What is the significance of the terraform state show command?

76. How do you use Terraform to provision a CloudFront distribution?

77. What are the best practices for managing Terraform provider versions?

78. How do you implement a disaster recovery strategy using Terraform?

79. What is the purpose of the terraform validate command in CI/CD pipelines?

80. How do you use Terraform to manage application load balancers?

81. What are the implications of using terraform apply in a multi-tenant environment?

82. How do you implement resource quotas in Terraform?

83. What is the purpose of the terraform state replace-provider command?

84. How do you use Terraform to provision a managed Kubernetes cluster?

85. What are the differences between terraform apply and terraform apply -refresh=true in a CI/CD context?

86. How do you handle resource naming conventions in Terraform?

87. What is the significance of the terraform state pull command in a team environment?

88. How do you use Terraform to manage secrets in a Kubernetes cluster?

89. What are the best practices for managing Terraform state files in a cloud environment?

90. How do you implement a security policy for Terraform-managed resources?

91. What is the purpose of the terraform workspace delete command?

92. How do you use Terraform to provision a data lake?

93. What are the implications of using terraform apply with the -input=false option?

94. How do you implement a logging strategy for Terraform-managed resources?

95. What is the significance of the terraform state pull command in disaster recovery?

96. How do you use Terraform to manage serverless API gateways?

97. What are the best practices for managing Terraform configurations in a microservices architecture?

98. How do you implement a change management process for Terraform configurations?

99. What is the purpose of the terraform state rm command in a collaborative environment?

100. How do you use Terraform to provision a machine learning model deployment?

Scenario-Based Questions

1. You need to create a multi-tier application architecture using Terraform. How would you structure your Terraform code?

2. Imagine you have a Terraform configuration that is failing due to a dependency issue. How would you troubleshoot and resolve it?

3. You are tasked with migrating an existing infrastructure to Terraform. What steps would you take?

4. How would you handle a situation where a resource was accidentally deleted in your Terraform-managed infrastructure?

5. You need to implement a blue-green deployment strategy for a web application using Terraform. Describe your approach.

6. How would you manage secrets for a production application using Terraform?

7. You are working in a team where multiple people are modifying Terraform configurations. How would you ensure consistency and avoid conflicts?

8. You need to provision resources in multiple cloud providers using Terraform. What strategies would you employ?

9. How would you implement a canary deployment strategy for a microservices architecture using Terraform?

10. You have a requirement to create a highly available architecture using Terraform. What components would you include?

11. How would you handle resource drift in a production environment managed by Terraform?

12. You need to create a Terraform module for a common infrastructure pattern. What considerations would you take into account?

13. Imagine you need to roll back a recent change that caused issues in production. How would you approach this using Terraform?

14. You are tasked with creating a Terraform configuration for a serverless application. What resources would you include?

15. How would you structure your Terraform code to support multiple environments (dev, staging, production)?

16. You need to implement monitoring and alerting for resources created by Terraform. What tools or services would you use?

17. How would you manage Terraform state files in a team environment to ensure security and accessibility?

18. You are required to provision a Kubernetes cluster using Terraform. What steps would you follow?

19. How would you handle a situation where a Terraform plan shows changes that you did not expect?

20. You need to create a Terraform configuration that provisions a database with read replicas. How would you do this?

21. Imagine you are integrating Terraform with a CI/CD pipeline. What best practices would you follow?

22. You need to create a Terraform module that can be reused across different projects. What features would you include?

23. How would you implement tagging for resources in a Terraform configuration to meet organizational policies?

24. You are tasked with creating a disaster recovery plan using Terraform. What components would you include?

25. How would you manage IAM roles and policies for a cloud provider using Terraform?

26. You need to provision a load balancer with multiple target groups. Describe your Terraform configuration.

27. How would you handle versioning of Terraform modules in a collaborative environment?

28. You are required to create a Terraform configuration for a hybrid cloud architecture. What considerations would you take into account?

29. How would you implement a security policy for resources managed by Terraform?

30. You need to provision a Redis cluster using Terraform. What resources and configurations would you include?

31. Imagine you have a Terraform configuration that is failing due to a provider issue. How would you troubleshoot it?

32. You need to create a Terraform configuration for a data lake. What components would you include?

33. How would you implement resource quotas in a Terraform-managed environment?

34. You are tasked with creating a Terraform configuration for a serverless API gateway. What steps would you follow?

35. How would you handle state file encryption in a Terraform configuration?

36. You need to provision a managed Kubernetes cluster. Describe your Terraform approach.

37. How would you implement a logging strategy for resources created by Terraform?

38. You are required to create a Terraform configuration for a multi-region deployment. What considerations would you take into account?

39. How would you manage Terraform configurations for different teams within an organization?

40. You need to create a Terraform configuration that provisions a machine learning model deployment. What resources would you include?

41. Imagine you need to create a highly available web application using Terraform. What architecture would you design?

42. How would you handle a situation where a resource was accidentally modified in your Terraform-managed infrastructure?

43. You are tasked with creating a Terraform configuration for a VPN connection. What steps would you follow?

44. How would you implement a change management process for Terraform configurations?

45. You need to provision a CloudFront distribution using Terraform. Describe your configuration.

46. How would you manage Terraform provider versions in a collaborative environment?

47. You are required to create a Terraform configuration for a microservices architecture. What components would you include?

48. How would you implement a rollback strategy in Terraform for a production environment?

49. You need to create a Terraform configuration that provisions a serverless database. What resources would you include?

50. Imagine you are integrating Terraform with a monitoring tool. What best practices would you follow?

51. You are tasked with creating a Terraform configuration for a multi-cloud deployment. What strategies would you employ?

52. How would you handle resource dependencies when using modules in Terraform?

53. You need to provision a data warehouse using Terraform. What components would you include?

54. How would you implement a security audit for Terraform-managed resources?

55. You are required to create a Terraform configuration for a content delivery network. What steps would you follow?

56. How would you manage Terraform configurations for a large-scale infrastructure?

57. You need to create a Terraform configuration that provisions a messaging queue. What resources would you include?

58. Imagine you have a Terraform configuration that is failing due to a syntax error. How would you troubleshoot it?

59. You are tasked with creating a Terraform configuration for a serverless function. What steps would you follow?

60. How would you implement a tagging strategy for resources in a Terraform configuration? 61. You need to provision a monitoring solution for your infrastructure using Terraform. What tools would you integrate?

62. How would you handle a situation where a Terraform plan shows unexpected changes due to a provider update?

63. You are required to create a Terraform configuration for a distributed database. What considerations would you take into account?

64. How would you implement a CI/CD pipeline for Terraform configurations?

65. You need to create a Terraform configuration that provisions a network security group. What rules would you define?

66. Imagine you are tasked with migrating an existing Terraform configuration to a new cloud provider. What steps would you take?

67. You need to provision a serverless API using Terraform. What resources and configurations would you include?

68. How would you manage Terraform state files when working with multiple environments?

69. You are required to create a Terraform configuration for a virtual private cloud (VPC). What components would you include?

70. How would you implement a resource naming convention in your Terraform configurations?

71. You need to create a Terraform configuration for a web application firewall. What steps would you follow?

72. How would you handle a situation where a Terraform apply fails due to insufficient permissions?

73. You are tasked with creating a Terraform configuration for a container orchestration platform. What resources would you include?

74. How would you implement a backup strategy for Terraform-managed resources?

75. You need to provision a service mesh using Terraform. What components would you include?

76. Imagine you have a Terraform configuration that is not following best practices. How would you refactor it?

77. You are required to create a Terraform configuration for a data pipeline. What resources would you include?

78. How would you manage Terraform configurations for a multi-tenant application?

79. You need to create a Terraform configuration that provisions a caching layer. What resources would you include?

80. How would you implement a security policy for Terraform-managed cloud resources?

81. You are tasked with creating a Terraform configuration for a hybrid cloud solution. What considerations would you take into account?

82. How would you handle a situation where a Terraform module is not working as expected?

83. You need to provision a serverless event-driven architecture using Terraform. What resources would you include?

84. How would you implement a compliance check for Terraform-managed resources?

85. You are required to create a Terraform configuration for a big data solution. What components would you include?

86. How would you manage Terraform configurations for a rapidly changing infrastructure?

87. You need to create a Terraform configuration that provisions a machine learning training environment. What resources would you include?

88. How would you implement a logging and monitoring strategy for Terraform-managed applications?

89. You are tasked with creating a Terraform configuration for a real-time data processing application. What resources would you include?

90. How would you handle a situation where a Terraform apply results in unexpected downtime?

91. You need to provision a multi-region application using Terraform. What strategies would you employ?

92. How would you implement a versioning strategy for Terraform modules in a collaborative environment?

93. You are required to create a Terraform configuration for a serverless data processing pipeline. What resources would you include?

94. How would you manage Terraform configurations for a large team with multiple projects?

95. You need to create a Terraform configuration that provisions a network load balancer. What steps would you follow?

96. How would you implement a security audit process for Terraform-managed infrastructure?

97. You are tasked with creating a Terraform configuration for a cloud-native application. What components would you include?

98. How would you handle a situation where a Terraform plan shows changes that conflict with existing resources?

99. You need to provision a data lake using Terraform. What resources and configurations would you include?

100. How would you implement a change management process for Terraform configurations in a regulated environment?

--------------------------------------------------------------------------------------------------------------------------

                                       Shell scripting

Scripting Questions for DevOps Engineers

Intermediate Questions:

1. What are positional parameters in shell scripting?

2. Difference between $\* and $@.

3. How do you read a file line-by-line in bash?

4. Use of set -e in scripts.

5. How to define and call functions.

6. How to check if a file exists.

7. Meaning of $? in shell scripting.

8. How to check if a variable is empty.

9. What does trap do in bash?

10. How to redirect both stdout and stderr.

11. How to use case statements.

12. How do you loop over files in a directory?

13. Use of IFS variable.

14. What is shift used for?

15. What is the purpose of exec?

16. Difference between source script.sh and ./script.sh.

17. How to pass arguments to a script.

18. How to exit a script with a status code.

19. How to perform arithmetic in shell scripts.

20. How to compare strings.

21. Difference between [ ] and [[ ]].

22. How do you comment a block in bash?

23. How to append to a file.

24. Difference between && and ||.

25. How to make a script executable.

26. How to get the length of a string.

27. How to slice a string in bash.

28. How to declare arrays and loop through them.

29. How to capture command output into a variable.

30. What is read command used for?

31. How do you write a script to monitor a process?

32. Use of getopts in command-line parsing.

33. What are here-documents in bash?

34. How to replace text in a file using sed.

35. How to count lines in a file.

36. What is the use of basename and dirname?

37. How to remove whitespace from a variable.

38. How to split a string by delimiter.

39. How to write logs to a log file.

40. How to check if a command exists in bash.

Advanced Questions

1. How to use associative arrays in bash.

2. How to handle parallel execution using &.

3. How do you use xargs with custom delimiters?

4. How do you trap and handle multiple signals?

5. Write a bash script with a dynamic menu.

6. Explain process substitution in bash.

7. How to profile a slow shell script.

8. How to create reusable bash libraries.

9. How to perform JSON parsing in shell.

10. How to read a config file and use values.

11. How to retry a failed command with a delay.

12. How to implement exponential backoff.

13. How to create a custom command-line tool with bash.

14. Use of eval and its dangers.

15. Difference between exec and subshell.

16. How to manage background jobs and wait.

17. How to ensure script idempotency.

18. How to prevent race conditions in parallel scripts.

19. How to create a lock file mechanism.

20. How to create colorized output in terminal.

21. How to encrypt and decrypt secrets in bash.

22. How to securely prompt for a password.

23. How to log stdout and stderr separately.

24. How to perform multi-threaded processing.

25. How to handle file locking in a bash script.

26. How to monitor memory usage of a process.

27. How to kill long-running processes via script.

28. How to write unit tests for bash scripts.

29. How to generate a dynamic HTML report.

30. How to create a cron job within a script.

31. How to parse XML with bash.

32. How to interact with REST API using curl in shell.

33. How to integrate a script with systemd service.

34. How to read .env file and export variables.

35. How to create a bash script to validate user input.

36. How to restrict execution to only one instance of the script.

37. How to use select for interactive menu creation.

38. How to write a script to monitor SSH failures.

39. How to build a shell wrapper for Docker commands.

40. How to display progress bars in bash.

Scenario-Based Questions

1. Write a script to monitor disk usage and alert when over 80%.

2. Write a script to clean /tmp directory older than 7 days.

3. Script to monitor a web service and alert on failure.

4. Rotate logs and keep last 5 files.

5. Backup MySQL DB daily with timestamp.

6. Validate IP addresses in a text file.

7. Read server IPs from a file and ping each.

8. Fetch and email top CPU-consuming processes.

9. SSH into multiple servers and execute commands.

10. Script to archive application logs to S3.

11. Script to restart a service if down.

12. Script to tail multiple logs in parallel.

13. Monitor SSL cert expiration.

14. Parse log file and count failed logins.

15. Report users with UID > 1000.

16. Find and kill zombie processes.

17. Script to compare two directories for differences.

18. Monitor network bandwidth in real-time.

19. Script to create users from CSV file.

20. Alert if a Docker container is not running.

21. Script to auto-deploy a static site using rsync.

22. Create a script to automate git pull and restart service.

23. Generate a dynamic inventory file for Ansible.

24. Build a script to validate YAML/JSON files.

25. Parse Jenkins job status via API and alert if failed.

---------------------------------------------------------------------------------------------------------------------------------

                          GIT and GITHUB

Basic Git Questions

1. What is Git, and why is it used?

2. Explain the difference between Git and GitHub.

3. How do you initialize a new Git repository?

4. What is the purpose of the Git staging area?

5. Describe the process of committing changes in Git.

6. What does the command `git status` do?

7. How do you check the commit history in Git?

8. What is a branch in Git?

9. How do you create a new branch in Git?

10. Explain the difference between `git merge` and `git rebase`.

11. What is a merge conflict, and how do you resolve it?

12. How do you delete a branch in Git?

13. What is the purpose of the `.gitignore` file?

14. How do you revert a commit that has already been pushed to a remote repository?

15. What is the difference between `git pull` and `git fetch`?

16. How do you stash changes in Git?

17. What is the difference between `git reset` and `git revert`?

18. How do you set up a remote repository in Git?

19. What is the difference between `origin` and `upstream` in Git?

20. How do you rename a branch in Git?

Intermediate Git Questions

1. Explain the concept of Git tags.

2. How do you create an annotated tag in Git?

3. What is the difference between a lightweight tag and an annotated tag?

4. How do you list all tags in a Git repository?

5. How do you delete a tag in Git?

Git Interview Questions (Basic to Advanced & Scenario-Based)

6. What is the purpose of the `git cherry-pick` command?

7. How do you use `git bisect` to find a bug?

8. What is the difference between `git clone` and `git fork`?

9. How do you configure a Git repository to ignore file mode changes?

10. What is the significance of the `HEAD` in Git?

11. How do you squash commits in Git?

12. What is the difference between `git stash pop` and `git stash apply`?

13. How do you apply a patch in Git?

14. What is the purpose of the 'git reflag' command?

15. How do you change the author of a commit in Git?

16. What is the difference between `git archive` and `git bundle`?

17. How do you handle large binary files in Git?

18. What is Git LFS, and when would you use it?

19. How do you sign commits in Git?

20. What is the purpose of the `git submodule` command?

Advanced Git Questions

1. Explain the internal structure of a Git repository.

2. How does Git handle data integrity?

3. What is the difference between `git gc` and `git prune`?

4. How do you optimize a Git repository?

5. What are Git hooks, and how do you use them?

6. How do you implement a pre-commit hook in Git?

7. What is the purpose of the `git fsck` command?

8. How do you recover a lost commit in Git?

9. What is the difference between `git blame` and `git annotate`?

10. How do you track the origin of a line in a file using Git?

11. What is the purpose of the `git worktree` command?

Git Interview Questions (Basic to Advanced & Scenario-Based)

12. How do you manage multiple worktrees in Git?

13. What is the difference between `git diff` and `git difftool`?

14. How do you configure an external diff tool in Git?

15. What is the purpose of the `git notes` command?

16. How do you add and view notes in Git?

17. What is the difference between `git log` and `git shortlog`?

18. How do you customize the output of `git log`?

19. What are reflogs, and how are they useful?

20. How do you expire and clean up reflogs in Git?

Scenario-Based Git Questions

1. You're working on a feature branch and realize it was branched off the wrong base. How do you rebase it

onto the correct base without losing your work?

2. After a pull request is merged, you realize that it introduced a bug. What steps do you take to fix it without

disrupting other developers' work?

3. You and a teammate both edited the same file. Youve committed your changes but cant push due to

merge conflicts. What do you do next?

4. You accidentally deleted a local branch that hadnt been pushed yet. How would you recover it?

5. You need to rollback your `main` branch to a previous commit, but others have already pulled the latest

changes. How do you handle this safely?

6. Your team follows a strict commit message convention. How do you ensure all contributors follow it

automatically?

7. A developer made a force push that overwrote shared history. How do you mitigate and recover from the

situation?

8. You need to cherry-pick multiple non-sequential commits from one branch to another. How do you ensure

no conflicts or dependency issues occur?

9. Youre asked to find the commit that introduced a performance degradation. How would you approach this

using Git?

10. Youve stashed your work multiple times. How do you inspect, apply, or drop a specific stash entry?

Git Interview Questions (Basic to Advanced & Scenario-Based)

11. Your team uses pull requests for code reviews. A teammate has many small, messy commits. What do

you suggest before merging?

12. A teammate pushed a large file to the repo accidentally. How do you remove the file completely from

history and reduce the repo size?

13. Youre managing a monorepo with multiple teams working on different projects. How do you maintain

clean Git workflows?

14. You want to review changes made by a teammate before a feature branch was merged. How do you do

that post-merge?

15. You accidentally pushed to the `main` branch instead of your feature branch. What do you do to fix it?

16. A tag was mistakenly created on the wrong commit and pushed to the remote. How do you fix and retag

properly?

17. Your company has a policy that requires code review before merging. How can Git hooks or automation

enforce this?

18. You cloned a repo, but when running `git pull`, you get a 'not a git repository' error. How would you

troubleshoot this?

19. You want to clean up your Git history before pushing your feature branch to the remote. What tools or

commands would you use?

20. Youre maintaining a forked repo and want to regularly sync it with the original upstream. How do you do

that effectively?

21. Youre reviewing a pull request that has unrelated changes mixed in. How do you ask the contributor to

clean it up, and what Git steps might they take?

22. During a rebase, you run into multiple conflicts. Whats your strategy to resolve them efficiently and

continue the rebase?

23. You have multiple developers pushing directly to the main branch, causing instability. How would you

enforce a Git workflow that prevents this?

24. Your CI/CD pipeline fails due to a missing configuration file after a merge. How can Git help you identify

where the issue was introduced?

25. Youve rebased a feature branch and now want to push it, but Git is rejecting it due to non-fast-forward

updates. How do you proceed?

26. You discover a security secret was committed to Git. How do you remove it from all history and rotate the

Git Interview Questions (Basic to Advanced & Scenario-Based)

key safely?

27. A teammate wants to undo their last 3 commits but keep the changes. What Git command would you

recommend?

28. You need to test a feature as it existed 3 weeks ago, but without knowing the exact commit hash. How do

you retrieve that snapshot?

29. You suspect a feature branch has diverged too far from main and might be hard to merge. How do you

verify and prepare for this?

30. Youre mentoring a junior developer who accidentally committed sensitive files. How do you walk them

through cleaning it up using Git?

---------------------------------------------------------------------------------------------------------------------------------

                                           LINUX

Basic Linux Questions:

1. What is Linux? How is it different from Windows?

2. What are the different Linux distributions?

3. Explain the Linux file system hierarchy.

4. What are the basic Linux file permissions?

5. How do you check the current working directory?

6. What command is used to list files and directories?

7. How do you create, delete, move, and rename files in Linux?

8. How do you view the contents of a file?

9. What is the difference between absolute and relative paths?

10.How do you check disk space usage in Linux?

11.What is the difference between rm, rmdir, and rm -rf?

12.How do you create a new user and group in Linux?

13.What is the purpose of the passwd command?

14.How do you change file ownership in Linux?

15.What is the purpose of the chmod command?

16.Intermediate Linux Questions:

17.What are hard links and soft links?

18.What is the difference between cron and at?

19.How do you schedule a cron job?

20.What is the /etc/passwd file?

21.What is the /etc/shadow file?

22.How do you check system uptime?

23.What are Linux runlevels?

24.How do you check the system's hostname?

25.How do you find a file in Linux?

26.What is the grep command used for?

27.How do you search for a specific word in a file?

28.How do you count the number of lines in a file?

29.What is the difference between find and locate?

30.How do you check currently running processes?

31.What is the purpose of the kill command?

32.How do you change a process's priority in Linux?

33.What is the difference between nice and renice?

34.How do you check CPU and memory usage?

Follow on Linkedin @JatinBansalDevops

35.How do you check open ports in Linux?

36.What is the difference between df and du commands?

37.How do you monitor real-time logs in Linux?

38.How do you find which process is using a particular port?

39.How do you check system logs in Linux?

40.What is the dmesg command used for?

41.What is the difference between ps, top, and htop?

42.Advanced Linux Questions:

43.What are the different types of process states in Linux?

44.How do you check environment variables in Linux?

45.How do you permanently set an environment variable?

46.What is the purpose of the /etc/fstab file?

47.How do you mount and unmount a filesystem?

48.What is LVM (Logical Volume Manager) in Linux?

49.How do you check free memory and swap space?

50.How do you clear swap space?

51.What is the difference between hard links and symbolic links?

52.What is the difference between sed and awk?

53.How do you extract specific columns from a file?

54.What is the difference between TCP and UDP?

55.How do you check network connectivity using Linux commands?

56.How do you view the routing table in Linux?

57.What is iptables, and how is it used in Linux?

58.What is the ufw firewall, and how do you configure it?

59.What is SSH, and how do you use it?

60.How do you set up passwordless SSH login?

61.How do you check disk partitions in Linux?

62.What is the purpose of chroot?

63.How do you create a swap file in Linux?

64.What is the difference between /dev/sda1 and /dev/sdb1?

65.What is the difference between systemctl and service commands?

66.How do you start, stop, and restart a service in Linux?

67.How do you check which services are enabled on startup?

68.What is the purpose of the journalctl command?

69.How do you extract specific lines from a file?

70.What is the difference between /proc, /sys, and /dev directories?

71.What are inodes in Linux?

72.What is the difference between nohup, screen, and tmux?

Follow on Linkedin @JatinBansalDevops

73.How do you copy an entire directory in Linux?

74.What is the purpose of the rsync command?

75.How do you set up a simple web server using python3 -m http.server?

76.What is the difference between /bin, /sbin, and /usr/bin?

77.How do you troubleshoot high CPU usage in Linux?

78.How do you kill multiple processes matching a pattern?

79.What is a zombie process in Linux?

80.How do you check if a user has sudo privileges?

81.What is the purpose of the /etc/hosts file?

82.How do you update all installed packages in Linux?

83.How do you list all installed packages on a Linux system?

84.What is the difference between rpm, yum, dnf, and apt?

85.How do you check the Linux kernel version?

86.How do you upgrade the Linux kernel?

87.How do you rollback a package update in Linux?

88.How do you create a new systemd service in Linux?

89.What is the purpose of the /var/log directory?

90.How do you check which Linux distribution you are using?

91.How do you secure an SSH server?

92.What is SELinux, and how does it work?

93.What is AppArmor, and how does it compare to SELinux?

94.What is the purpose of the /etc/resolv.conf file?

95.How do you configure a static IP in Linux?

96.How do you enable IP forwarding in Linux?

97.What is a kernel panic, and how do you troubleshoot it?

98.What is the difference between ext3, ext4, and XFS filesystems?

99.How do you perform a filesystem check in Linux?

Scenario-based Linux interview questions:

File & Directory Management Scenarios

1. You accidentally deleted an important file. How would you recover it in Linux?

2. A user reports that they cannot access a file even though they have read

permissions. What could be the issue?

3. You need to find and delete all log files older than 30 days. How would you do that?

4. A directory has thousands of files, and rm -rf is taking too long to delete them. How

can you speed up the deletion process?

Follow on Linkedin @JatinBansalDevops

5. How do you find the top 10 largest files on your Linux system?

6. A script is failing due to a "permission denied" error, even though you are running it

as root. What could be the reason?

7. You need to rename multiple files in a directory that have spaces in their names.

How would you do it?

8. How do you compare two directories and find the differences in files?

9. A user is unable to create files in a directory, but ls -l shows they have write

permission. What could be the problem?

10.A symbolic link you created is broken. How do you troubleshoot and fix it?

User & Permission Management Scenarios

1. A user complains they cannot log in via SSH but their credentials are correct. How

do you troubleshoot?

2. You need to give a user access to run only a specific command with sudo. How do

you configure that?

3. You have a large team, and you need to create 100 users with a script. How would

you do that?

4. A user left the company, and you need to disable their account but retain their files.

How do you do it?

5. You need to change ownership of all files belonging to a deleted user to another

user. How would you do this?

6. How do you reset the root password if you have lost it?

7. A user has been locked out due to too many failed login attempts. How do you

unlock them?

8. You need to add a user to multiple groups in one command. How would you do it?

9. How do you prevent a user from logging in but allow their cron jobs to continue

running?

10.A team needs shared access to a directory, but files they create should be

automatically accessible by others in the team. How do you configure this?

Process & Performance Management Scenarios

1. Your application is running slowly. How do you diagnose and fix the issue?

2. A process is consuming 100% CPU. How do you identify and fix it?

3. How do you identify which process is using the most memory?

Follow on Linkedin @JatinBansalDevops

4. A critical process keeps getting killed due to an "Out of Memory" error. How do you

debug and prevent this?

5. You need to keep a process running in the background even after logging out. How

would you do that?

6. A process is stuck and not responding to kill -9. What else can you try?

7. How do you set a process to run with low priority automatically?

8. You need to monitor system resource usage over time. Which tools would you use?

9. Your server is running out of swap memory frequently. What steps would you take to

fix this?

10.How do you identify and stop zombie processes?

Networking & Security Scenarios

1. A website running on your Linux server is inaccessible, but the server is up. How do

you troubleshoot?

2. You need to allow only a specific IP to access a server over SSH. How would you

configure it?

3. Your server is under a DDoS attack. How do you mitigate it using Linux commands?

4. You need to check which ports are open and which services are listening. How do

you do that?

5. A user is unable to reach an external website from a Linux server. How do you

troubleshoot network connectivity?

6. You need to set up a firewall to allow only HTTP, HTTPS, and SSH traffic. How do you

do it?

7. How do you check which process is using a specific port?

8. A connection to a database is failing from your Linux server. How do you

troubleshoot it?

9. You need to change the default SSH port for security reasons. How do you do it?

10.How do you enable IP forwarding on a Linux system?

Storage & Filesystem Scenarios

1. Your disk is 100% full. How do you find and remove unnecessary files?

2. How do you check which directories are using the most disk space?

3. A mount point is missing after a server reboot. How do you ensure it mounts

automatically?

4. How do you extend a partition without losing data?

Follow on Linkedin @JatinBansalDevops

5. Your /var partition is full. How do you move it to another disk without downtime?

6. How do you check the health of a hard disk in Linux?

7. How do you recover a deleted partition table?

8. A filesystem is mounted as read-only. How do you fix it?

9. How do you check and repair a corrupted filesystem?

10.What steps would you take to create a swap file and enable it?

System Logs & Troubleshooting Scenarios

1. Your server is rebooting randomly. How do you investigate the cause?

2. A user reports that they are getting a "Permission Denied" error when accessing a

log file. What could be the issue?

3. An application is crashing frequently. How do you collect logs and debug it?

4. How do you configure logging to rotate logs automatically?

5. Your system logs are filling up disk space. How do you manage them?

6. How do you check which user deleted a specific file?

7. How do you troubleshoot a slow boot process in Linux?

8. How do you check if a service crashed and restarted automatically?

9. How do you enable persistent logging on a Linux system?

10.A cron job is not running as expected. How do you debug it?

Backup & Disaster Recovery Scenarios

1. How do you back up a Linux system and restore it after failure?

2. You need to create a scheduled incremental backup of a directory. How do you do

it?

3. A developer accidentally overwrote a configuration file. How do you restore it from a

backup?

4. You need to clone a Linux system to another machine. What tools would you use?

5. How do you take a snapshot of a Linux filesystem for quick rollback?

6. You need to recover files from a failed RAID array. How do you do it?

7. How do you back up a MySQL/PostgreSQL database from the command line?

8. How do you ensure backups are automatically verified for integrity?

9. How do you restore a Linux system from a full disk backup?

10.How do you create a bootable recovery disk in Linux?

System & Kernel Scenarios

1. How do you upgrade the Linux kernel safely?

2. You need to enable a specific kernel module. How do you do it?

3. A server crashes with a kernel panic. How do you troubleshoot and fix it?

4. How do you find out which kernel version is running on your system?

5. You need to disable a specific kernel module permanently. How do you do it?

6. Your server has high load but low CPU and memory usage. What could be causing

it?

7. How do you check if the system supports virtualization?

8. A package update broke a system service. How do you roll back the update?

9. How do you force a system to reboot after a kernel panic?

10.How do you apply security patches to a running Linux system without rebooting?

Real-world situations (problem-solving and troubleshooting)

File Management & Permissions Scenarios

1. You created a file, but another user cannot access it even though they have read

permissions. What could be the issue?

2. A user is unable to delete a file, even though they have write permission on the file.

What might be causing this?

3. You need to change permissions on all files inside a directory but not on the

directory itself. How would you do that?

4. A directory has rwx------ permissions, but the owner cannot execute scripts inside it.

What could be wrong?

5. You need to give full permissions to a user on a file but prevent them from deleting it.

How can you achieve this?

6. You need to transfer a large file between two servers, but SCP is too slow. What

alternative methods can you use?

7. How do you find and list all files with 777 permissions and correct them securely?

8. A user has access to a file but gets a "Permission Denied" error when trying to

execute it. How do you fix it?

9. How do you ensure that newly created files in a shared directory automatically have

the correct permissions?

10.You need to remove execute permission from all .sh files inside a directory. How

would you do it?

Follow on Linkedin @JatinBansalDevops

11.A script runs fine manually but fails with a "Permission Denied" error when executed

via a cron job. What could be wrong?

12.You need to prevent a user from accessing a specific directory while keeping their

other access intact. How do you do it?

13.A user needs write access to a file but must not be able to read it. How can you

achieve this?

14.A file has rwxrwxrwx permissions, but a user still can't modify it. What could be the

reason?

15.You want to track changes made to a critical file. How do you do that?

File System & Storage Scenarios

1. Your /home partition is full, but there is free space in /var. How do you resolve this

without data loss?

2. A user reports that they cannot create new files, but df -h shows enough free space.

What could be the issue?

3. A mounted NFS share is not accessible, but the NFS service is running fine. How do

you troubleshoot?

4. After rebooting, a mounted filesystem disappears. How do you ensure it mounts

automatically?

5. You need to move a large directory from one disk to another without downtime. How

do you do it?

6. How do you check the available inodes on a Linux system? What do you do if inodes

are full?

7. A file appears to be taking up space, but du -sh and ls -lh show different sizes. What

could be happening?

8. How do you find and delete files taking up too much disk space in /var/log/?

9. You resized a partition, but df -h still shows the old size. What should you do next?

10.A user needs to mount an external drive, but they get a "Permission Denied" error.

How do you fix it?

11.How do you safely unmount a filesystem that is currently in use?

12.You need to create a new ext4 filesystem on a raw disk. How do you do it?

13.A disk shows as "read-only" after a power failure. How do you fix it?

14.You need to copy an entire disk to another disk. What command would you use?

15.How do you check which process is consuming the most disk I/O?

Advanced Storage & Backup Scenarios

1. How do you create and restore a tar backup of a directory, while preserving file

permissions?

Follow on Linkedin @JatinBansalDevops

2. You need to take an incremental backup of a directory every day. What tools would

you use?

3. How do you create a snapshot of a filesystem before making changes?

4. You need to set up RAID 1 mirroring for a production server. How would you

configure it?

5. A RAID 5 array has one failed disk. How do you replace it and rebuild the array?

6. You need to move all data from one disk to another without downtime. What’s the

best approach?

7. A user accidentally deleted a file. How do you recover it if it's not in a backup?

8. How do you check disk errors and repair a corrupted filesystem?

9. You need to encrypt a disk partition before storing sensitive data. How do you do it?

10.How do you schedule an automated disk usage report to be sent via email every

week?

NFS & Network Storage Scenarios

1. An NFS mount is extremely slow. How do you diagnose and fix the issue?

2. How do you restrict NFS access to a specific IP range?

3. A user gets "Stale NFS file handle" errors. What could be the cause?

4. You need to allow a client to mount an NFS share with read-only access. How do

you configure it?

5. After restarting the NFS server, clients cannot mount shared directories. How do you

troubleshoot?

6. How do you monitor disk usage on a remote system using SSH?

7. A Samba share is accessible but files copied to it are showing wrong permissions.

How do you fix it?

8. How do you mount a remote Linux filesystem securely over SSH?

9. A CIFS share is not mounting at boot. How do you ensure it mounts automatically?

10.How do you list all mounted remote filesystems and their options?

Logical Volume Management (LVM) Scenarios

1. You need to create a new logical volume from available free space. How do you do

it?

2. A logical volume is full, and you need to extend it. How do you do that?

3. How do you reduce the size of an LVM partition safely?

4. A volume group is out of space. How do you add a new physical disk to extend it?

5. How do you migrate data from one LVM disk to another without downtime?

6. A logical volume got corrupted. How do you restore it from a snapshot?

7. You need to encrypt an existing LVM partition. How do you proceed?

Follow on Linkedin @JatinBansalDevops

8. An LVM partition is not mounting after a reboot. How do you troubleshoot it?

9. How do you move a logical volume from one system to another?

10.How do you check which physical disks are used in an LVM setup?

File & Storage Performance Optimization Scenarios

1. A large number of small files are slowing down disk performance. How do you

optimize it?

2. How do you enable write caching on a disk to improve performance?

3. An application frequently reads a large file, causing high disk I/O. How do you

optimize it?

4. How do you enable and configure filesystem journaling for better data integrity?

5. How do you measure disk I/O speed and latency?

6. You need to optimize an NFS-mounted directory for better performance. What

changes would you make?

7. How do you configure Linux to store temporary files in RAM instead of disk?

8. A server frequently runs out of available disk cache. How do you troubleshoot it?

9. How do you prevent an application from consuming too much disk I/O?

10.What steps would you take to convert an ext4 filesystem to XFS without losing data?

------------------------------------------------------------------------------------------------------------------------------------

                                        Docker

Docker Questions Intermediate to Advance:

Intermediate Docker Interview Questions

1. What are the key differences between Docker containers and virtual machines?

2. How do Docker images differ from containers?

3. Explain the use of docker-compose and its advantages.

4. What is the purpose of the Dockerfile? Walk through a basic example.

5. How does Docker networking work by default?

6. How do you create a custom Docker network and why?

7. What’s the role of a Docker volume?

8. What are the types of Docker volumes and when do you use them?

9. How can you persist data in Docker containers?

10. What is the purpose of the EXPOSE instruction in a Dockerfile?

11. How do you inspect the logs of a Docker container?

12. How do you access a running container’s shell?

13. What’s the difference between CMD and ENTRYPOINT in a Dockerfile?

14. How do you restart a container automatically on failure?

15. How do you limit CPU and memory usage for a container?

16. What is a bind mount and how does it differ from a volume?

17. What happens when a container is deleted? Is the data lost?

18. What are Docker tags and how are they used in image management?

19. What is the difference between docker stop and docker kill?

20. How do you clean up unused images and containers?

21. How can you check the container’s resource usage?

22. What are the lifecycle states of a Docker container?

23. What’s the difference between COPY and ADD in Dockerfiles?

24. What are Docker labels used for?

25. What is the .dockerignore file and why is it useful?

26. How do you handle environment variables in Docker?

27. What is a multi-stage build in Docker and why is it useful?

28. How can you see which container is using which port?

29. How do you connect multiple containers using Docker Compose?

30. What is the purpose of a Docker registry?

31. How do you pull a specific version of a Docker image?

32. How do you share images within an organization securely?

33. What’s the purpose of health checks in a Dockerfile?

34. How do you build a Docker image from a Dockerfile?

35. How do you push Docker images to a remote registry?

36. What’s the difference between docker ps and docker ps -a?

37. What is an orphaned volume? How do you handle it?

38. How do you debug a Docker build that’s failing?

39. How do you list all Docker networks?

40. What are the differences between bridge, host, and none networks?

41. How do you make a container communicate with a host service?

42. How do you copy files from a container to the host?

43. How do you remove all stopped containers?

44. What are container image layers?

45. What is a dangling image? How do you remove it?

46. How do you inspect the details of an image?

47. What happens when two containers try to use the same port?

48. What is the use of --link in Docker networking?

49. What’s the difference between docker exec and docker run?

50. How do you update a Docker container with a new image version?

Advanced Docker Interview Questions

1. How does Docker ensure container isolation on a Linux system?

2. What are cgroups and namespaces, and how do they work in Docker?

3. How do you reduce the size of a Docker image significantly?

4. What security concerns should you consider when building Docker images?

5. How do you scan Docker images for vulnerabilities?

6. What is Docker Content Trust (DCT), and how does it enhance security?

7. How does Docker handle image layering under the hood?

8. What is a Union File System in Docker, and why is it important?

9. How do you implement image immutability in CI/CD pipelines?

10. How do you handle Docker secrets in production?

11. What is the difference between secrets and environment variables?

12. How can you pass secrets securely to a container without hardcoding?

13. Explain how to use docker save and docker load.

14. How do you ensure containers are ephemeral and stateless?

15. What are the best practices to follow while writing Dockerfiles?

16. What are the common anti-patterns in Dockerfile usage?

17. How do you layer caching in a Dockerfile for optimal performance?

18. Explain the build cache mechanism in Docker and how it works.

19. What is the difference between Docker BuildKit and traditional build?

20. How do you enable Docker BuildKit?

21. What is the impact of using latest tag in production deployments?

22. How do you version Docker images properly in production environments?

23. What tools can you use for Docker image analysis and auditing?

24. How do you monitor Docker containers in a production environment?

25. How do you set up logging for Docker containers in a centralized way?

26. What’s the difference between logging drivers: json-file, syslog, fluentd?

27. How do you troubleshoot a container that runs fine locally but not in prod?

28. How can you mount a configuration file dynamically inside a container?

29. What are init containers and how are they used?

30. What is the difference between ENTRYPOINT as exec vs shell form?

31. Why is it not recommended to run containers as root?

32. How can you configure non-root users inside containers?

33. How do you handle large file transfers between containers and host?

34. How do you analyze Docker container performance bottlenecks?

35. How can you isolate container network traffic for compliance?

36. How can you limit container IO bandwidth?

37. What is user namespace remapping in Docker?

38. How do you run Docker containers in a fully isolated environment?

39. What is the best way to back up persistent data in Docker?

40. How do you enforce resource quotas across multiple containers?

41. How do you update secrets/configs without restarting containers?

42. How do you prevent container escape vulnerabilities?

43. How do you avoid race conditions during container startup?

44. What are zombie containers and how do you clean them up?

45. How do you run Docker containers on Kubernetes vs standalone Docker?

46. How do you simulate network failures in containers for testing?

47. What are the common issues with Docker and IPv6?

48. How do you enable IPv6 support for Docker containers?

49. How do you configure custom DNS for Docker containers?

50. How do you restrict container access to certain domains or IPs?

51. How do you automate container health and readiness checks?

52. How do you roll back a faulty image deployed to production?

53. What is Docker’s image signing mechanism?

54. How do you restrict container image sources to trusted registries?

55. What are container runtime interfaces (CRI)? How does Docker interact with them?

56. How does Docker manage shared layers across images?

57. What’s the purpose of squash option while building images?

58. What are tmpfs mounts and their benefits in Docker?

59. How can you isolate containers using AppArmor/SELinux profiles?

60. How do you perform fine-grained container access controls?

61. How do you implement rate limiting for containers?

62. How do you manage image lifecycles in an enterprise setup?

63. What’s the role of container image scanning in DevSecOps?

64. How do you detect and mitigate CVEs in base images?

65. How do you rotate container credentials without downtime?

66. What’s the best way to share a Docker volume across multiple services?

67. How do you secure Docker’s remote API?

68. What are the implications of Docker socket exposure?

69. How do you manage Docker Swarm secrets and configs?

70. How do you handle rollbacks in Docker Swarm vs Kubernetes?

71. What are the challenges with multi-arch builds and how do you solve them?

72. How can you build images for ARM and x86 architectures?

73. What is a layered filesystem in Docker and its pros/cons?

74. How does Docker handle file system changes during container runtime?

75. How can you trace file changes inside a running container?

76. How can you export container logs to external systems?

77. What is Docker rootless mode and how do you enable it?

78. How do you implement backup and restore for containerized databases?

79. What are best practices for ephemeral vs persistent volumes?

80. How do you avoid storage bloat in containerized environments?

81. What’s the impact of Docker garbage collection?

82. How do you set up a local Docker registry with authentication?

83. How do you mirror Docker Hub images for offline access?

84. What’s the role of overlay networks in Docker Swarm?

85. How do you perform zero-downtime deployments using Docker?

86. What is blue-green deployment using Docker?

87. How do you load test containers under real-world conditions?

88. What’s the difference between Docker pause and stop?

89. What are health checks and how do they help in orchestration?

90. How do you design a CI/CD pipeline with Docker at the core?

91. How do you promote Docker images across environments (dev → prod)?

92. What is the benefit of using .dockerignore in builds?

93. How do you run Docker-in-Docker (dind)? What are the risks?

94. How do you isolate container workloads on shared infrastructure?

95. How do you containerize an app that uses system-level dependencies?

96. What are the limitations of Docker for production-scale apps?

97. How does Docker interact with kernel-level features?

98. How can you optimize image rebuilds in large monorepos?

99. What’s the difference between daemonless and traditional Docker runtimes?

100. What tools complement Docker for production-grade deployments?

-----------------------------------------------------------------------------------------------------------------------------

                                               AWS

AWS cloud infrastructure

Compute (EC2, Auto Scaling, Elastic Beanstalk):

1. Explain the difference between On-Demand, Reserved, and Spot Instances.

2. What are the different EC2 instance types and when would you choose each?

3. How do you ensure high availability for an EC2 instance?

4. Describe the purpose and benefits of using Auto Scaling groups.

5. What are the different scaling policies available in Auto Scaling?

6. How do you configure a launch template or launch configuration for an Auto Scaling group?

7. Explain the lifecycle hooks in Auto Scaling.

8. What are the benefits of using Elastic Load Balancing (ELB)?

9. Describe the different types of ELBs (Application Load Balancer, Network Load Balancer, Classic Load Balancer) and their use cases.

10. How do you configure health checks for an ELB?

11. What is the purpose of Elastic Beanstalk?

12. What are the different deployment environments in Elastic Beanstalk?

13. How do you deploy and manage applications using Elastic Beanstalk?

14. Explain the concept of instance metadata and user data in EC2.

15. How can you securely access EC2 instances?

16. What are Placement Groups in EC2 and when would you use them?

17. Explain the difference between horizontal and vertical scaling.

18. How can you monitor the performance of your EC2 instances?

19. What are the limitations of Spot Instances?

20. How can you automate the process of starting and stopping EC2 instances?

Storage (S3, EBS, EFS):

21. Explain the different storage classes in Amazon S3 and their use cases.

22. How do you control access to objects in S3?

23. What are S3 Bucket Policies and Access Control Lists (ACLs)?

24. How do you enable versioning on an S3 bucket? What are the benefits?

25. Explain the concept of S3 lifecycle rules.

26. How can you improve the performance of S3?

27. What is Amazon EBS? Explain the different EBS volume types.

28. How do you create and manage EBS snapshots?

29. How can you increase the size of an EBS volume?

30. What is Amazon EFS? What are its use cases?

31. How do you mount an EFS file system to EC2 instances?

32. What are the performance considerations for EFS?

33. Explain the difference between block storage (EBS) and object storage (S3).

34. How can you encrypt data at rest in S3 and EBS?

35. What is S3 Intelligent-Tiering?

Networking (VPC, Subnets, Security Groups, NACLs, Route 53):

36. What is an Amazon Virtual Private Cloud (VPC)?

37. Explain the concept of subnets (public and private).

38. What is an Internet Gateway (IGW) and how is it used?

39. What is a NAT Gateway and why is it needed in a private subnet?

40. Explain the purpose of Security Groups. How do they work?

41. What are Network Access Control Lists (NACLs)? How do they differ from Security Groups?

42. What is Amazon Route 53? What are its key functionalities?

43. Explain the different routing policies in Route 53.

44. How do you register a domain name using Route 53?

45. How do you configure DNS records in Route 53?

46. What is a VPC peering connection? What are its limitations?

47. What are VPC endpoints? Explain the different types.

48. How do you create a VPN connection to your VPC?

49. What is AWS Direct Connect? When would you use it?

50. Explain the concept of Elastic IP addresses?

Databases (RDS, DynamoDB):

51. What is Amazon Relational Database Service (RDS)?

52. Explain the different database engines supported by RDS.

53. How do you create and manage RDS instances?

54. How do you ensure high availability for an RDS database?

55. What are RDS read replicas? How are they used?

56. How do you perform backups and restores in RDS?

57. What is Amazon DynamoDB? What are its key characteristics?

58. Explain the concepts of primary key, partition key, and sort key in DynamoDB.

59. What are DynamoDB read and write capacity units (RCUs and WCUs)?

60. How do you query and scan data in DynamoDB?

61. What are DynamoDB Global Secondary Indexes (GSIs) and Local Secondary Indexes (LSIs)?

62. How do you manage backups and restores in DynamoDB?

63. What is DynamoDB Streams?

64. Explain the difference between SQL and NoSQL databases.

65. When would you choose RDS over DynamoDB, and vice versa?

Security (IAM, KMS, Secrets Manager):

66. What is AWS Identity and Access Management (IAM)?

67. Explain the concepts of IAM users, groups, and roles.

68. What are IAM policies? How do you create and attach them?

69. What is the principle of least privilege? How do you apply it in IAM?

70. What is Multi-Factor Authentication (MFA) and why should you enable it for IAM users?

71. What is the IAM Security Token Service (STS)?

72. What is AWS Key Management Service (KMS)? How does it work?

73. What are Customer Master Keys (CMKs)? Explain the different types.

74. How do you encrypt and decrypt data using KMS?

75. What is AWS Secrets Manager? When would you use it?

76. How does Secrets Manager help in managing database credentials?

77. Explain the concept of IAM roles for EC2 instances.

78. How do you audit IAM activities?

79. What is AWS Organizations? How can it help manage multiple AWS accounts?

80. What are Service Control Policies (SCPs)?

Monitoring and Management (CloudWatch, CloudTrail):

81. What is Amazon CloudWatch? What are its key features?

82. How do you create CloudWatch alarms?

83. What are CloudWatch Logs? How do you collect and analyze them?

84. What are CloudWatch Metrics? What types of metrics are available?

85. What is AWS CloudTrail? What information does it record?

86. How can you use CloudTrail for security and compliance?

87. How do you analyze CloudTrail logs?

88. What is AWS Config? How does it help with compliance?

89. What are AWS Trusted Advisor? What types of checks does it perform?

90. What is AWS Systems Manager? What are some of its key capabilities?

Advanced Level Questions

      Compute:

• Explain the advanced networking options for EC2, such as Enhanced Networking (ENA) and Elastic Fabric Adapter (EFA).

• How can you optimize the performance of compute-intensive workloads on EC2?

• Describe the architecture and benefits of using AWS Lambda for serverless computing.

• How do you troubleshoot performance issues in Auto Scaling groups?

• Explain advanced load balancing concepts like sticky sessions and cross-zone load balancing.

• How can you integrate Elastic Beanstalk with other AWS services?

• Describe the different container orchestration services offered by AWS (ECS, EKS, Fargate).

• When would you choose ECS over EKS, and vice versa?

• Explain the architecture of Amazon EKS and its components (control plane, worker nodes).

• What are the different networking options for containers in AWS?

 Storage:

• How can you optimize the cost of your S3 storage using different storage classes and lifecycle policies?

• Explain advanced S3 features like S3 Select and S3 Glacier Deep Archive.

• How do you design a highly available and durable storage solution using S3 and other AWS services?

• Describe advanced EBS features like Multi-Attach and Fast Snapshot Restore.

• How can you optimize the performance and cost of your EBS volumes?

• Explain advanced EFS features like EFS One Zone and EFS Infrequent Access.

• How do you implement data replication and disaster recovery for your storage solutions?

• What are the considerations for choosing between different block storage options (EBS, Instance Store)?

• How can you integrate AWS Storage Gateway with your on-premises environment?

• Explain the purpose and benefits of using Amazon FSx?

Networking:

• Design a highly available and scalable network architecture on AWS.

• Explain advanced VPC networking concepts like Transit Gateway and PrivateLink.

• How do you troubleshoot network connectivity issues within your VPC?

• Describe advanced Route 53 features like weighted routing, latency-based routing, and failover routing.

• How can you secure your VPC using advanced security measures?

• Explain the architecture and benefits of using AWS CloudFront as a CDN.

• How do you configure and manage CloudFront distributions?

• What are Web Application Firewalls (WAF) and how can you use them with CloudFront and ALB?

• How do you implement hybrid networking between your on-premises environment and AWS?

• Explain the different options for connecting your on-premises network to AWS (VPN, Direct Connect).

Databases:

• Design a highly available and scalable database architecture on AWS.

• Explain advanced RDS features like Multi-AZ deployments, cross-region read replicas, and Aurora Global Database.

• How do you optimize the performance of your RDS databases?

• Describe advanced DynamoDB features like DAX (DynamoDB Accelerator) and Global Tables.

• How do you design efficient data models for DynamoDB?

• Explain the use cases for other AWS database services like ElastiCache, Redshift, and Neptune.

• How do you migrate your existing databases to AWS?

• What are the considerations for choosing the right database service for your application?

• How do you implement database security on AWS?

• Explain the concept of databaseless applications using services like DynamoDB and API Gateway?

Security:

• Design a comprehensive security strategy for your AWS environment.

• Explain advanced IAM features like attribute-based access control (ABAC) and federated identity.

• How do you implement and manage encryption at rest and in transit across different AWS services?

• Describe advanced KMS features like custom key stores and cross-account key sharing.

• How do you use AWS Security Hub to centralize security findings?

• What is Amazon GuardDuty and how does it help in threat detection?

• Explain the role of AWS Inspector in identifying security vulnerabilities.

• How do you implement a robust incident response plan for your AWS environment?

• What are AWS WAF rules and how do you configure them to protect your applications?

• How do you manage and audit access to sensitive data in your AWS environment?

Monitoring and Management:

• Design a comprehensive monitoring and logging solution for your AWS environment.

• Explain advanced CloudWatch features like metric math and cross-account dashboards.

• How do you use CloudWatch Events (EventBridge) to build event-driven architectures?

• Describe advanced CloudTrail features like log file integrity validation and multi-region trails.

• How do you use AWS Config rules to automate compliance checks?

• Explain advanced AWS Systems Manager capabilities like Patch Manager, Automation, and Parameter Store.

• How do you implement infrastructure as code (IaC) using AWS CloudFormation or AWS CDK?

• What are the best practices for managing and updating your AWS infrastructure using IaC?

• How do you implement cost optimization strategies for your AWS environment?

• Explain the use of AWS Cost Explorer and AWS Budgets.

Compute Scenarios:

• Your web application experiences sudden spikes in traffic. How would you design your compute infrastructure to handle this efficiently and cost-effectively?

• You need to run a batch processing job that requires a large number of compute resources for a short period. What AWS compute service would be most suitable?

• You have a microservices architecture. How would you deploy and manage your containers on AWS?

• Your application requires low latency access to a large dataset in memory. What EC2 instance type and storage options would you consider?

• You need to automatically replace unhealthy EC2 instances in your production environment. How would you achieve this?

• You want to deploy a simple web application quickly without managing the underlying infrastructure. What AWS service could you use?

• You need to run a custom application on EC2 instances, but you want to ensure they are launched with a consistent configuration. How would you achieve this?

• Your application needs to scale based on custom metrics. How would you configure Auto Scaling to achieve this?

• You want to ensure that your critical EC2 instances are always running, even in the event of an Availability Zone failure. How would you design for this?

• You need to provide temporary access to an EC2 instance for a developer without sharing permanent SSH keys. How can you do this securely?

Storage Scenarios:

• You need to store a large number of static files for your website and serve them with low latency to users globally. What AWS storage and content delivery services would you use?

• You need a durable and cost-effective solution for archiving infrequently accessed data for compliance purposes. What S3 storage class would you choose?

• Your application requires a shared file system that can be accessed concurrently by multiple EC2 instances. What AWS storage service would be suitable?

• You need to back up your on-premises database to AWS. What AWS storage options could you use?

• You want to implement a data lake on AWS to store and analyze large volumes of structured and unstructured data. What AWS storage services would be involved?

• You need to encrypt sensitive data at rest in S3 and ensure that only authorized users can access it. How would you configure this?

• Your application generates a large number of small files that need to be processed. How can you optimize the storage and retrieval of these files in S3?

• You need to replicate data between two different S3 buckets in different AWS regions for disaster recovery. How would you set this up?

• Your application requires block storage with high IOPS and low latency. What EBS volume type would you choose?

• You need to take consistent snapshots of your EC2 instance's root volume before performing maintenance. How would you automate this process?

Networking Scenarios:

• You need to create a secure and isolated network environment for your application on AWS. How would you design your VPC?

• You have a public-facing web application and a backend database that should not be directly accessible from the internet. How would you configure your VPC subnets and SG?

• You need to allow traffic from a specific IP address range to your EC2 instances. How would you configure your security groups?

• You want to inspect all network traffic going in and out of your VPC for security purposes. What AWS service could you use?

• You need to connect your on-premises data center to your AWS VPC. What are the different options and when would you choose each?

• You want to improve the performance and availability of your web application for users around the world. How would you use Route 53 and CloudFront?

• You need to route traffic to different versions of your application based on user-defined criteria. What Route 53 routing policy could you use?

• You want to restrict access to your S3 buckets to only requests originating from your VPC. How can you achieve this?

• You need to create a private connection between your VPC and an AWS service without exposing your traffic to the public internet. What AWS service would you use?

• You have multiple VPCs in different AWS accounts that need to communicate with each other. How would you establish connectivity?

Database Scenarios:

• You need a highly available and scalable relational database for your transactional application. What AWS database service would you recommend?

• Your application requires very low latency reads and writes and can tolerate eventual consistency. What AWS database service would be a good fit?

• You need to perform complex analytical queries on a large dataset. What AWS database service would be most suitable?

• You want to offload read traffic from your primary RDS database. How would you achieve this?

• You need to automatically scale the storage and compute capacity of your relational database based on demand. What AWS service features can help?

• You need to migrate a large on-premises Oracle database to AWS with minimal downtime. What AWS services and strategies could you use?

• Your application needs a graph database to model and query relationships between data. What AWS database service would you consider?

• You want to implement in-memory caching to improve the performance of your database reads. What AWS service could you use?

• You need to ensure that your RDS database is backed up regularly and that you can restore it quickly in case of failure. How would you configure this?

• You need to control access to your DynamoDB tables and items based on fine-grained permissions. How would you achieve this?

Security Scenarios:

• You need to enforce strong password policies for all IAM users in your AWS account. How would you configure this?

• You want to grant a third-party vendor temporary access to specific S3 buckets in your account. How would you do this securely using IAM roles and policies?

• You need to monitor all API calls made in your AWS account for security and compliance purposes. What AWS service would you use?

• You suspect that an unauthorized user has gained access to your AWS environment. What steps would you take to investigate and remediate this?

• You need to protect your web application from common web exploits like SQL injection and cross-site scripting. What AWS service could you use?

• You want to centrally manage and rotate database credentials for your applications running on EC2 instances. How would you achieve this?

• You need to ensure that all data stored in your S3 buckets is encrypted at rest. How would you enforce this?

• You want to receive alerts when potentially malicious activity is detected in your AWS environment. What AWS service can help you with this?

• You need to implement a system to audit access to sensitive resources and track changes made to your AWS infrastructure. What AWS services would you use?

• You want to prevent users from creating EC2 instances of a certain type in your AWS account. How can you enforce this restriction?

Monitoring and Management Scenarios:

• Your web application is experiencing performance issues, and you need to identify the bottleneck. What CloudWatch metrics and logs would you examine?

• You want to be notified automatically when the CPU utilization of your EC2 instances exceeds a certain threshold. How would you configure this?

• You need to track the configuration changes made to your AWS resources over time for compliance purposes. What AWS service would you use?

• You want to automate the process of patching the operating systems of your EC2 instances. How can you achieve this?

• You need to collect and analyze logs from multiple AWS services in a centralized location. What AWS service would you use?

• You want to visualize the performance and health of your application components in a single dashboard. What AWS service can help you create this?

• You need to automatically scale your Auto Scaling group based on the number of messages in an SQS queue. How would you configure this?

• You want to manage your AWS infrastructure as code. What AWS service or tool would you use?

• You need to understand the cost breakdown of your AWS usage. What AWS service can provide this information?

• You want to set up alerts when your AWS spending exceeds a certain budget. How would you configure this?

Integration and Hybrid Cloud Scenarios:

• You need to integrate your on-premises application with a serverless function running on AWS Lambda. How would you design this integration?

• You want to extend your on-premises network to AWS and allow seamless communication between resources in both environments. What AWS services would you use?

• You need to migrate a large amount of data from your on-premises data center to AWS S3. What are some strategies and tools you could use?

• You have an application that needs to access resources in multiple AWS accounts. How would you manage cross-account access securely?

• You want to build an event-driven architecture where different AWS services communicate with each other through events. What AWS service would you use as a central event bus?

• You need to provide secure access to your internal web applications running on EC2 instances to users outside your corporate network without exposing them directly to the internet.

 How would you achieve this?

• You want to use your existing on-premises Active Directory for authentication and authorization of users accessing AWS resources. How would you integrate these systems?

• You need to build a CI/CD pipeline for deploying applications to your AWS environment. What AWS services could you use in this pipeline?

• You have a regulatory requirement to store data in a specific geographic region. How would you ensure that your AWS resources and data are deployed in that region?

• You want to monitor the health and performance of your hybrid cloud environment (both on-premises and AWS resources) from a single pane of glass. What AWS services could you leverage?

Disaster Recovery and Business Continuity Scenarios:

• You need to design a disaster recovery plan for your critical web application running on EC2 and RDS. What are some common DR strategies and how would you implement them on AWS?

• You want to minimize downtime for your application in the event of an Availability Zone failure. How would you architect your application across multiple AZs?

• You need to back up your DynamoDB tables and be able to restore them in a different AWS region in case of a regional disaster. How would you configure this?

• You want to test your disaster recovery plan to ensure its effectiveness. What AWS services and techniques can help you with this?

• You need to ensure that your critical data stored in S3 is protected against accidental deletion or data loss. What S3 features can help?

• You want to implement a warm standby disaster recovery strategy for your database. How would you configure this using RDS?

• You need to recover your entire AWS environment in a new region in the event of a catastrophic failure in your primary region. What are some key considerations for this scenario?

• You want to automate the failover process for your application in case of a disaster. How can you achieve this using AWS services?

• You need to meet a Recovery Time Objective (RTO) of less than 15 minutes for your critical application. How would you design your infrastructure to meet this requirement?

• You need to meet a Recovery Point Objective (RPO) of near zero for your transactional database. What AWS services and configurations would you use?

Cost Optimization Scenarios:

• You notice that your EC2 costs are higher than expected. What are some common reasons for this, and how would you identify and address them?

• You have a workload that runs predictably for a specific period each day. How can you optimize the cost of your EC2 instances for this workload?

• You are storing a large amount of data in S3 that is accessed infrequently. How can you reduce the storage costs?

• You want to analyze your AWS spending and identify areas where you can save money. What AWS tools can help you with this?

• You need to right-size your RDS instances to ensure you are not overpaying for resources you are not using. How would you determine the optimal instance size?

• You want to reduce the cost of your data transfer out of AWS. What strategies could you employ?

• You have non-critical workloads that can tolerate interruptions. How can you leverage Spot Instances to reduce compute costs?

• You want to automatically stop and start your development and testing EC2 instances outside of business hours. How can you automate this?

• You are using Elastic Load Balancers. How can you optimize their costs?

• You want to get recommendations on how to optimize your AWS spending. What AWS service can provide these recommendations?

Serverless Scenarios:

• You need to build a serverless API that can handle a large number of requests. What AWS services would you use?

• You want to automatically process files uploaded to an S3 bucket. How can you achieve this using serverless services?

• You need to build a real-time data processing pipeline using serverless technologies. What AWS services would be suitable?

• You want to schedule a task to run periodically without managing any servers. What AWS service could you use?

• You need to integrate your serverless application with a relational database. What are some considerations and best practices?

• You want to monitor the performance and troubleshoot issues in your serverless application. What AWS services can help?

• You need to secure your serverless API and control access to it. How would you implement authentication and authorization?

• You want to deploy and manage your serverless application infrastructure as code. What AWS service or tool would you use?

• You need to build a mobile backend that can handle user authentication, data storage, and push notifications using serverless services. What AWS services would you leverage?

• You want to build a chatbot that uses AI services and integrates with other applications using serverless functions. What AWS services could you use?

Containerization Scenarios:

• You need to deploy and manage a containerized web application that requires high availability and scalability. What AWS container orchestration service would you choose?

• You want to run your Docker containers without managing the underlying EC2 instances. What AWS service could you use?

• You need to orchestrate a large number of microservices running in Docker containers. What AWS service would be most suitable?

• You want to build a CI/CD pipeline for your containerized applications. How would you integrate Amazon ECR and your chosen container orchestration service?

• You need to securely store and manage your Docker images. What AWS service would you use?

• You want to monitor the health and performance of your containerized applications. What AWS services can help?

• You need to scale your containerized application based on custom metrics. How would you configure this?

• You want to control network traffic between your containers. What networking options are available in AWS container services?

• You need to manage secrets and configuration for your containerized applications. How would you do this securely on AWS?

• You want to run batch processing jobs using Docker containers. What AWS service would be a good fit?

Big Data and Analytics Scenarios:

• You need to process a large volume of streaming data in real-time. What AWS services would you use?

• You want to build a data lake to store and analyze large amounts of structured and unstructured data. What AWS services would be involved?

• You need to perform interactive SQL queries on petabytes of data stored in S3. What AWS service would you use?

• You want to build a machine learning model using data stored in your data lake. What AWS services could you leverage?

• You need to visualize your data and create dashboards. What AWS service would you use?

• You want to process and analyze log data from various AWS services at scale. What AWS services would be suitable?

• You need to build an ETL (Extract, Transform, Load) pipeline to move and transform data between different data stores on AWS. What AWS services would you use?

• You want to run large-scale data processing jobs using the Hadoop framework. What AWS service would you use?

• You need to build a recommendation engine based on user behavior data. What AWS services could you leverage?

• You want to perform serverless data transformation and analysis. What AWS services would be appropriate?

Application Integration Scenarios:

• You need to decouple different components of your application and enable asynchronous communication between them. What AWS service would you use for message queuing?

• You want to integrate different applications and services using a fully managed integration service. What AWS service would you use?

• You need to build a workflow that involves multiple steps and integrates with different AWS services. What AWS service would you use to orchestrate this workflow?

• You want to build a real-time communication channel between your web application and backend services. What AWS service could you use?

• You need to send email notifications from your application. What AWS service would you use?

• You want to build a scalable and reliable API for your application. What AWS services would you use?

• You need to integrate your application with third-party SaaS applications. What AWS services could facilitate this?

• You want to build an event-driven architecture where different services react to events published by other services. What AWS service would you use as a central event bus?

• You need to implement a caching layer to improve the performance of your application. What AWS service would you use for in-memory caching?

• You want to build a search functionality for your application data stored in DynamoDB. What AWS service could you integrate with?

Edge Computing and IoT Scenarios:

• You need to collect and process data from a large number of IoT devices. What AWS services would you use?

• You want to run compute and analytics closer to your IoT devices to reduce latency. What AWS service could you use for edge computing?

• You need to securely connect and manage your IoT devices. What AWS service would you use for device management?

• You want to build an IoT application that can react to events from your devices in real-time. What AWS services would you leverage?

• You need to analyze time-series data from your IoT devices. What AWS service would be suitable for this?

• You want to build a voice interface for your IoT devices. What AWS service could you integrate with?

• You need to securely provision and authenticate your IoT devices. What AWS services would you use for device security?

• You want to build a mobile application that can interact with your IoT devices. What AWS services would you use for the mobile backend?

• You need to deploy and manage machine learning models on your edge devices. What AWS service could you use?

• You want to build a smart home application that integrates with various IoT devices. What AWS services would you leverage?

Migration Scenarios:

• You are planning to migrate a large number of virtual machines from your on-premises data center to AWS. What AWS service would you use to facilitate this?

• You need to migrate a large database to AWS with minimal downtime. What AWS services and strategies could you use?

• You want to re-architect your monolithic application into microservices on AWS. What are some key considerations and AWS services you would use?

• You need to migrate your existing web application to AWS without making significant code changes. What AWS services would be suitable?

• You want to migrate your data warehouse to AWS. What AWS service would you consider?

• You need to discover and assess your on-premises infrastructure before migrating to AWS. What AWS tools can help with this?

• You want to migrate your containerized applications to AWS. What are the different options available?

• You need to migrate a large amount of static data to AWS S3. What are some efficient methods for doing this?

• You want to migrate your DNS management to AWS Route 53. What are the steps involved?

• You are migrating a mission-critical application to AWS and need to ensure high availability and minimal disruption. What are some key architectural considerations?

Multi-Account Scenarios:

• You need to manage multiple AWS accounts for different departments in your organization. What AWS service can help you with this?

• You want to centrally manage security policies across all your AWS accounts. How can you achieve this?

• You need to share resources (e.g., VPCs, subnets) between different AWS accounts securely. What AWS services can facilitate this?

• You want to consolidate billing for all your AWS accounts into a single bill. How can you set this up?

• You need to grant cross-account access to specific resources in your AWS environment. How would you configure IAM roles for this?

• You want to implement a standardized set of security controls across all your AWS accounts. What AWS services and strategies would you use?

• You need to monitor the health and compliance of resources across all your AWS accounts from a central location. What AWS services can help?

• You want to automate the creation and management of new AWS accounts within your organization. What AWS service could you use?

• You need to enforce tagging policies across all your AWS accounts for cost tracking and resource management. How can you achieve this?

• You want to delegate administrative responsibilities to different teams for their respective AWS accounts while maintaining overall governance. How can you structure this using AWS Organizations and IAM?

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

                                                 SHELL SCRIPTING

Scripting Questions for DevOps Engineers

Intermediate Questions:

1. What are positional parameters in shell scripting?

2. Difference between $\* and $@.

3. How do you read a file line-by-line in bash?

4. Use of set -e in scripts.

5. How to define and call functions.

6. How to check if a file exists.

7. Meaning of $? in shell scripting.

8. How to check if a variable is empty.

9. What does trap do in bash?

10. How to redirect both stdout and stderr.

11. How to use case statements.

12. How do you loop over files in a directory?

13. Use of IFS variable.

14. What is shift used for?

15. What is the purpose of exec?

16. Difference between source script.sh and ./script.sh.

17. How to pass arguments to a script.

18. How to exit a script with a status code.

19. How to perform arithmetic in shell scripts.

20. How to compare strings.

21. Difference between [ ] and [[ ]].

22. How do you comment a block in bash?

23. How to append to a file.

24. Difference between && and ||.

25. How to make a script executable.

26. How to get the length of a string.

27. How to slice a string in bash.

28. How to declare arrays and loop through them.

29. How to capture command output into a variable.

30. What is read command used for?

31. How do you write a script to monitor a process?

32. Use of getopts in command-line parsing.

33. What are here-documents in bash?

34. How to replace text in a file using sed.

35. How to count lines in a file.

36. What is the use of basename and dirname?

37. How to remove whitespace from a variable.

38. How to split a string by delimiter.

39. How to write logs to a log file.

40. How to check if a command exists in bash.

Advanced Questions

1. How to use associative arrays in bash.

2. How to handle parallel execution using &.

3. How do you use xargs with custom delimiters?

4. How do you trap and handle multiple signals?

5. Write a bash script with a dynamic menu.

6. Explain process substitution in bash.

7. How to profile a slow shell script.

8. How to create reusable bash libraries.

9. How to perform JSON parsing in shell.

10. How to read a config file and use values.

11. How to retry a failed command with a delay.

12. How to implement exponential backoff.

13. How to create a custom command-line tool with bash.

14. Use of eval and its dangers.

15. Difference between exec and subshell.

16. How to manage background jobs and wait.

17. How to ensure script idempotency.

18. How to prevent race conditions in parallel scripts.

19. How to create a lock file mechanism.

20. How to create colorized output in terminal.

21. How to encrypt and decrypt secrets in bash.

22. How to securely prompt for a password.

23. How to log stdout and stderr separately.

24. How to perform multi-threaded processing.

25. How to handle file locking in a bash script.

26. How to monitor memory usage of a process.

27. How to kill long-running processes via script.

28. How to write unit tests for bash scripts.

29. How to generate a dynamic HTML report.

30. How to create a cron job within a script.

31. How to parse XML with bash.

32. How to interact with REST API using curl in shell.

33. How to integrate a script with systemd service.

34. How to read .env file and export variables.

35. How to create a bash script to validate user input.

36. How to restrict execution to only one instance of the script.

37. How to use select for interactive menu creation.

38. How to write a script to monitor SSH failures.

39. How to build a shell wrapper for Docker commands.

40. How to display progress bars in bash.

Scenario-Based Questions

1. Write a script to monitor disk usage and alert when over 80%.

2. Write a script to clean /tmp directory older than 7 days.

3. Script to monitor a web service and alert on failure.

4. Rotate logs and keep last 5 files.

5. Backup MySQL DB daily with timestamp.

6. Validate IP addresses in a text file.

7. Read server IPs from a file and ping each.

8. Fetch and email top CPU-consuming processes.

9. SSH into multiple servers and execute commands.

10. Script to archive application logs to S3.

11. Script to restart a service if down.

12. Script to tail multiple logs in parallel.

13. Monitor SSL cert expiration.

14. Parse log file and count failed logins.

15. Report users with UID > 1000.

16. Find and kill zombie processes.

17. Script to compare two directories for differences.

18. Monitor network bandwidth in real-time.

19. Script to create users from CSV file.

20. Alert if a Docker container is not running.

21. Script to auto-deploy a static site using rsync.

22. Create a script to automate git pull and restart service.

23. Generate a dynamic inventory file for Ansible.

24. Build a script to validate YAML/JSON files.

25. Parse Jenkins job status via API and alert if failed.

-----------------------------------------------------------------------------------------------------------------------

                                      SONARQUBE

1. Difference between code quality and code coverage?

2. What is default qualitygate in sonar?

3. What is the output of sonarqube, how to fix if any smell code/vurnabilities  found?

4. How do you integrate SonarQube & Snyk in Azure pipeline?

5. How are you integrating the SonarQube with the Jenkins server?

6. what is sonarqube and which purpose it is used for?

7. Docker cmd and entrypoint difference, how to configure sonarqube with azure?

8. Please explain how did you setup sonarqube in your CI jenkins pipeline, what are the quality gates and how did you set the threshold checks for code coverage leveraging the test reports created by developer?

9. Difference between code quality and code coverage?

10. What is the output of sonarqube, how to fix if any smell code/vurnabilities  found?

11. How are you managing code coverage in your project?

------------------------------------------------------------------------------------------------------------------------

                              ANSIBLE

What are ansible roles, and how do they improve playbooks organization?

Can you explain the difference between vars, defaults and set\_facts?

How does ansible handle variable precedence?

How do you use include\_role and import\_role? What is the difference?

What are some use cases for block and rescue in playbooks?

What are dynamic inventory, and how difference from static inventory?

Can you describe a scenario where would you use a dynamic inventory plugin?

How do you use templates in ansible?

What are handles in ansible, and how do they work?

10. What is ansible vault? How did you manage sensible data?

11. Can you de-crypt an Ansible Vault-encrypted file on the fly during execution?

12. How do you write a custom Ansible module?

13. Can you provide a example scenario where a custom module is required?

14. How do you optimize Ansible playbook performance?

15. How do you debug a Ansible playbook?

16. What are some best practices for writing Ansible Playbooks?

17. How do you test Ansible Playbooks?

18. How would you handle rolling updates using playbooks?

19. Describe how you would integrate ansible with Jenkins?

20. What is the purpose of the meta directory in roles?

21. How does Ansible handle dependencies between roles?

22. How Ansible is different from Puppet?

23. What is the minimum requirement for Ansible to work?

24. Which protocol does Ansible use to communicate with Linux & Windows?

25. What are Registered Variables in Ansible?

26. How Ansible manages multiple communication protocols?

27.How to handle different machines needing different user accounts or ports to log in with using Ansible?

28. How to see a list of all of the ansible variables?

29. How to create an AWS EC2 key using Ansible?

30. How to upgrade the Ansible version to the latest version?

31. How to install a WordPress application inside a Docker container using Ansible?

32. How to generate encrypted passwords for the user module in Ansible?

33. How to keep secret data in my playbook in Ansible?

34. What is the minimum requirement for using Docker modules in Ansible?

35. How does the Ansible module connect to Docker API?

36. How do you write an Ansible playbook, and what client requirements do you consider?

--------------------------------------------------------------------------------------------------------------------------