|  |
| --- |
|  |
|  |  |
|  | **Devops Interview Question Bank**  **Topic- Github**   1. What is GIT ? 2. What is difference between GIT & Github ? 3. Why we use GIT ? 4. What is SCM & VCS ? 5. What are the process of pushing the code to Github Repository ? 6. Why do we commit ? 7. What are the commands of GIT to push the code ? 8. How you can merge a git repository with another ? 9. What is branching in git ? 10. Different types of branching in GIT ? 11. What is merge conflict in git ? 12. How you can resolve merge conflict if you are merging same project and in the same   branch ?  1. Have you worked with containers? What is a docker basically?  Docker container is an open source software development platform. Its main benefit is to package applications in “containers,” allowing them to be portable among any system running the Linux operating system (OS). |
|  |  |
|  | 2. What is orchestration in cloud computing? |
|  |  |
|  | In most situations, cloud automation describes a task or function accomplished without human intervention. Cloud orchestration describes the arranging and coordination of automated tasks, ultimately resulting in a consolidated process or workflow. It is simplest to see this in an example. |
|  |  |
|  | 3. What is Docker and what does it do? |
|  |  |
|  | Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. |
|  | Docker provides this same capability without the overhead of a virtual machine. It lets you put your environment and configuration into code and deploy it. The same Docker configuration can also be used in a variety of environments. This decouples infrastructure requirements from the application environment. |
|  |  |
|  | 4. Do you have any expreience with CI tools? Which ones? |
|  | Jenkins, Bamboo, TeamCity, etc. |
|  |  |
|  | 5. Describe your experience implementing continuous deployment in a production environment |
|  | Answer should describe personal experience on types of jobs run, systems integrated, etc. |
|  |  |
|  | 6. What function does DNS play on a network? |
|  | The DNS plays a critical role in supporting the Internet infrastructure by providing a distributed and fairly robust mechanism that resolves Internet host names into IP addresses and IP addresses back into host names. |
|  |  |
|  | 7. What is HTTP? |
|  | The Hypertext Transfer Protocol (HTTP) is an application protocol for distributed, collaborative, hypermedia information systems.[1] HTTP is the foundation of data communication for the World Wide Web. |
|  | Hypertext is structured text that uses logical links (hyperlinks) between nodes containing text. HTTP is the protocol to exchange or transfer hypertext. |
|  |  |
|  | 8. What commands do you know that can be used to check DNS records? |
|  | $ host example.com |
|  | $ nslookup example.com  9. How to get all environment variables and how can you use them?  All UNIX-like operating systems such as OpenBSD, Linux, Redhat, CentOS, Debian allows you to set environment variables. When you log in on UNIX, your current shell (login shell) sets a unique working environment for you which is maintained until you log out.  printenv\env - command to print all or part of environment  Full command:  $ printenv PATH HOME  $PATH - Display lists directories the shell searches, for the commands.  $HOME - User's home directory to store files.  10. What is version control and why should version control system (VCS) be used?  Define version control and talk about how this system records any changes made to one or more files and saves them in a centralized repository. VCS tools will help you recall previous versions and perform the following:  Go through the changes made over a period of time and check what works versus what doesn’t.  Revert specific files or specific projects back to an older version.  Examine issues or errors that have occurred due to a particular change.  Using VCS gives developers the flexibility to simultaneously work on a particular file and all modifications can be logically combined later.  11. What is Docker hub?  Docker hub is a cloud-based registry service which allows you to link to code repositories, build your images and test them, stores manually pushed images, and links to Docker cloud so you can deploy images to your hosts. It provides a centralized resource for container image discovery, distribution and change management, user and team collaboration, and workflow automation throughout the development pipeline.  12. What is Dockerfile used for?  Dockerfile is nothing but a set of instructions that have to be passed on to Docker itself, so that it can build images automatically reading these instructions from that specified Dockerfile. A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image. Using docker build users can create an automated build that executes several command-line instructions in succession.  13. Tell us how you have used Docker in your past position?  This is a question that you could bring upon your whole experience with Docker and if you have used any other Container technologies before Docker. You could also explain the ease that this technology has brought in the automation of the development to production lifecycle management. You can also discuss about any other integrations that you might have worked along with Docker such as Puppet, Chef or even the most popular of all technologies – Jenkins. If you do not have any experience with Docker itself but similar tools from this space, you could convey the same and also show in your interest towards learning this leading containerization technology.  14. How to create Docker container?  You can create a Docker container out of any specific Docker image of your choice and the same can be achieved using the command given below:  docker run -t -i command name  The command above will create the container and also starts it for you. In order to check whether the Docker container is created and whether it is running or not, you could make use of the following command. This command will list out all the Docker containers along with its statuses on the host that the Docker container runs.  docker ps -a  15. How to stop and restart the Docker container?  The following command can be used to stop a certain Docker container with the container id as  CONTAINER\_ID:  docker stop CONTAINER\_ID  The following command can be used to restart a certain Docker container with the container id as  CONTAINER\_ID:  docker restart CONTAINER\_ID  16. How far do Docker containers scale?  Best examples in the Web deployments like Google, Twitter and best examples in the Platform Providers like Heroku, dotCloud run on Docker which can scale from the ranges of hundreds of thousands to millions of containers running in parallel, given the condition that the OS and the memory doesn’t run out from the hosts which runs all these innumerable containers hosting your applications.  17. What are the various states that a Docker container can be in at any given point in time?  There are four states that a Docker container can be in, at any given point in time. Those states are as given as follows:  • Running  • Paused  • Restarting  • Exited  19. Can you remove a paused container from Docker?  To answer this question blatantly, No, it is not possible to remove a container from Docker that is just paused. It is a must that a container should be in the stopped state, before it can be removed from the Docker container.  20. What is Mesos?  Mesos is a cluster manager that provides efficient resource isolation and sharing across distributed applications or frameworks. Mesos is a open source software originally developed at the University of California at Berkeley.  21. What is Marathon?  Marathon is a production-grade container orchestration platform for Mesosphere's Datacenter Operating System (DCOS) and Apache Mesos.  22. What are the basic parameters you have to pass in a json file to orchestrate a service on Mesos via Marathon?  Bare minimum parameters would be service name, cpu, memory and application to deploy and the containerizer to be used.  23. What is Git?  Git is a Distributed Version Control system (DVCS). It can track changes to a file and allows you to revert back to any particular change.  24. In Git how do you revert a commit that has already been pushed and made public?  It can be done in two ways:  By removing or fixing the bad file in a new commit and pushing it to the remote repository. Once the necessary changes to the file has been made, commit it to the remote repository. Use: git commit -m “commit message”  By creating a new commit that undoes all changes that were made in the bad commit. To do this, use command: git revert <name of bad commit>  25. What Is Ldap?  A GLDAP (Light-Weight Directory Access Protocol) determines how an object in an Active Directory should be named. LDAP is the industry standard directory access protocol, making Active Directory widely accessible to management and query applications. Active Directory supports LDAPv2 and LDAPv3.  26. According to an HTTP monitor, a website is down. You're able to telnet to the port, so how do you resolve the issue?  "Assuming the web page is up, I would investigate what could be wrong with the monitor. It could be a system overload or flapping, among other issues. Identifying the problem helps me prevent it in the future."  27. What are three attributes that make you a great systems engineer?  This question lends insight into candidates' self-awareness skills as well as whether their values match those of your business. Look for:  Answers that match areas of emphasis in your job description  At least one soft skill  Attributes that fit your company culture  "I'm analytical and curious. I always dig to find out why a problem occurred. Otherwise, it is liable to happen again and hurt the business. I am also a great communicator, able to share my insight with anyone in jargon-free language."  28. A Linux administrator wants to review the messages that scrolled up the screen during a system boot. How can this be accomplished?  The boot kernel messages are saved to the log file /var/log/dmesg. He can check this file.  29. Which utility could you use to repair the corrupted file system?  You can use fsck to repair the corrupted file system.  30. A user can not access a remote server. Which command he can use to verify that remote server is up and which command should he use to check if port 22 is open or closed?  He can use ping command to check whatever remote server is up or not and use the telnet command to check port accessibility.  1. Explain the difference between rebasing and merge in Git?  • Git rebase is a command that allows developers to integrate changes from one branch to another.  • Git merge is a command that allows you to merge branches from Git.  Git rebase and merge both integrate changes from one branch into another. Where they differ is how they used. Git rebase moves a feature branch into a master. Git merge adds a new commit, preserving the history.  (If you’re working alone or on a small team, use rebase. If you’re working with a big team, use merge.)  2. Have you faced the situation where you resolve conflicts in Git? How?  A merge conflict is an event that takes place when Git is unable to automatically resolve differences in code between two commits. Git can merge the changes automatically only if the commits are on different lines or branches. Here are the steps that will help you resolve conflicts in Git:  1. The easiest way to resolve a conflicted file is to open it and make any necessary changes  2. After editing the file, we can use the git add a command to stage the new merged content  3. The final step is to create a new commit with the help of the git commit command  4. Git will create a new merge commit to finalize the merge  3. How to revert a commit that has already been pushed and made public?  There are two processes through which you can revert a commit:  1. Remove or fix the bad file in a new commit and push it to the remote repository. Then commit it to the remote repository using:  git commit –m “commit message”  2. Create a new commit to undo all the changes that were made in the bad commit. Use the following command:  git revert <commit id>  4. Tell about the commands git reset — mixed and git merge — abort?.  git reset — mixed is used to undo changes made in the working directory and staging area.  git merge — abort helps stop the merge process and return back to the state before the merging began.  5. How will you find a list of files that has been modified in a particular commit?  The command to get a list of files that has been changed in a particular commit is:  git diff-tree –r {commit hash}  • -r flag allows the command to list individual files  • commit hash lists all the files that were changed or added in the commit.  6. How will you fix a broken commit? What command you will use?  To fix a broken commit in Git, We use the “git commit — amend” command, which helps us combine the staged changes with the previous commits instead of creating a fresh new commit.  7. Explain git stash drop?  Git ‘stash drop’ command is used to remove the stashed item. This command will remove the last added stash item by default, and it can also remove a selected item as well.  Ex: If you want to delete item named stash@{manoj}; you can use the command:  git stash drop stash@{manoj}.  8. Explain about “git cherry-pick”?  This command enables you to pick up commits from a branch within a repository and apply it to another branch. This command is useful to undo changes when any commit is accidentally made to the wrong branch. Then, you can switch to the correct branch and use this command to git cherry-pick the commit.  9. Can you tell the difference between git pull and git fetch?  Git pull command pulls new changes or commits from a particular branch from your central repository and updates your target branch in your local repository. (Git pull = git fetch + git merge)  Git fetch is also used for the same purpose but it works in a slightly different way. When you perform a git fetch, it pulls all new commits from the desired branch and stores it in a new branch in your local repository. If you want to reflect these changes in your target branch, git fetch must be followed with a git merge.  10. What is origin in Git?  Origin refers to the remote repository that a project was originally cloned from and is used instead of the original repository’s URL.  11. What is the difference between resetting and reverting?  git reset changes the state of the branch to a previous one by removing all of the states after the desired commit,  git revert does it through the creation of new reverting commits and keeping the original one intact.  12. What is ‘staging area’ or ‘index’ in Git?  That before completing the commits, it can be formatted and reviewed in an intermediate area known as ‘Staging Area’ or ‘Index’. Every change is first verified in the staging area and then that change is committed to the repository.  13. What work is restored when the deleted branch is recovered?  The files which were stashed and saved in the stash index list will be recovered back. Any untracked files will be lost. Also, it is a good idea to always stage and commit your work or stash them.  14. What is Head in Git?  Git maintains a variable for referencing, called HEAD to the latest commit in the recent checkout branch. So if we make a new commit in the repo then the pointer or HEAD is going to move or change its position to point to a new commit.  15. What is the purpose of branching and its types?  It allows the user to switch between the branches to keep the current work in sync without disturbing master branches and other developer’s work as per their requirements.  · Feature branching — A feature branch model keeps all of the changes for a particular feature inside of a branch. When the feature is fully tested and validated by automated tests, the branch is then merged into master.  · Task branching — In this branch, each task is implemented on its own branch with the task key included in the branch name. It is easy to see which code implements which task, just look for the task key in the branch name.  · Release branching — Once the develop branch has acquired enough features for a release, you can clone that branch to form a Release branch. Creating this branch starts the next release cycle, so no new features can be added after this point, only bug fixes, documentation generation, and other release-oriented tasks should go in this branch. Once it is ready to ship, the release gets merged into master and tagged with a version number.  1. What is Version control system or Source code manager?  2. Benefits of VCS or SCM?  3. What is Git and What is a repository in GIT?  4. Diffrence between Git and SVN?  5. Two types of git authentication?  6. Branches in Git and its purpose?  7. What is Pull requests?  8. How to install git in Linux & Windows?  9. How to setup repository though command line?  10. How to setup repository in github and clone it?  11. What is git clone command used for?  12. What is git config command user for?  13. Git config data is stored in what location?  14. Git config global and local files?  15. Content of git config file?  16. Git add command’s purpose?  17. How to remove/rename files in local git repo?  18. git commit command’s purpose?  19. How to sync local git repo data with github?  20. git fetch?  21. git merge?  22. git pull?  23. How to change branches in local git repo?  24. What is the difference between ‘git remote’ and ‘git clone’?  25. git status?  26. What is the function of ‘git rm’?  27. What is the function of ‘git checkout’ in git?  28. What is the use of ‘git log’?  29. Explain what is commit message?  30. How to setup github ssh authentication?  31. What is git webhooks?  33.Tell me the branching strategies you followed for your project?  34. What is git rebase?  35. What is git stash?  36. What is the difference between git pull and git fetch?  37. Differnce between git and svn?  38. Advantages of git compare with svn?  39. Explain about staging area in git?  40. How do you check after git merge …where merging is correctly or not?  41. How to create branch and delete a branch?  42. Do you get any conflicts while merging..how can u resolve conflicts?  43. How can u implement authentication and authorization in git?  44. What is the difference between git rebase and git merge?  45. How to merge the code?  46. Pull request in git hub - ---git pull  47. Forking a repository in git hub?  48. Diff between svn and git ---svn is centralized version control and git is Distributed Version Control.?  49. What is forking in git repositories?  50. explain about soft and hard reset and their difference?  51. What kind of branching and merging you worked on GIT.  52. with this kind of environment and the git release branching, does it work in Agile methodology?  53. What is tagging in GIT and what is the diff between feature branching and tagging?  54. How can you restrict a user to push the code to the repository only if the code does not meet the quality ?  55. When will you perform Git rebase and git revert and git Merge ? Why?  **Topic- Jenkins**   1. What is Jenkins ? 2. Why we use Jenkins ? 3. What are the other tools/technologies present in market other than 4. Jenkins for CI/CD ? 5. How to move Jenkins from one server to another ? 6. How to create Jenkins backup ? 7. What are plugins in Jenkins ? 8. What are the default plugins installed in Jenkins ? 9. How to schedule builds in Jenkins ? 10. Difference between Ant, Maven, Gradle ? 11. Difference between Jenkins, Teamcity and Bamboo ? 12. How to configure a cloud access in Jenkins ? 13. What is Jenkins slaves ? 14. How to run a groovy script in Jenkins ? 15. What is Jenkins Pipeline ? 16. What are different types of Jenkins Pipeline ? 17. What is Declarative pipeline in Jenkins ? 18. Is Jenkins a CI tool or both CI/CD ? 19. How to install Jenkins with non root access in Linux ? 20. If you have 200 employees in your company, how you can assign Jenkins 21. access to these employee how you can give permission in Jenkins ? 22. **CI/CD** 23. What is CI & CD ? 24. What is CI/CD pipeline ? 25. Difference between Continuous Delivery & Deployment ? 26. List the important tools & technologies used in Devops ?   Jenkins Task  Task 1  Write the Jenkins pipeline code for Java & Php application  Task 2  Write the Jenkinsfile code to build a Java application with Maven with error handling  Task 3  Complete the following tasks:  1. Jenkins setup on linux  2. Setup app server with apache to deploy an app.  3. create three jobs on jenkins  4. Pull the code from git repo  5. Build the application  6. deploy an app on apache using ansible.  7. app deploy should work with single trigger hit(git pull job -> build app -> deploy on apache server)  8. job should get triggered on git push on git repo |
|  | **Topic- Ansible**   1. What is Ansible ? 2. What is Configuration Management ? 3. Is Ansible only a tool for Configuration Management ? 4. What are the components of Ansible ? 5. How Ansible works ? 6. What are the other tools in market other than Ansible ? 7. How Ansible is different from Chef & Puppet ? 8. What is Inventory in Ansible ? 9. What are the types of Inventories ? 10. What is play & playbook ? 11. Difference between hosts & groups ? 12. What is Roles ? 13. How to install a Role ? 14. How to install multiple roles ? 15. How to create roles ? 16. What is Dynamic Inventory & when we use it & for what ? 17. Where is the Ansible Configuration file located ? 18. What are the different ways other than SSH by which Ansible 19. can connect to remote hosts ? 20. What is variable in Ansible ? 21. What are different types of variables ? 22. How to assign variables in group vars & hosts vars ? 23. Difference between File & Template directory in Roles ? 24. Difference between default & vars directory in Roles ? 25. What is Jinja 2 template ? 26. What is modules in Ansible ? 27. Difference between COPY & FILE modules ? 28. Difference between SHELL & COMMAND modules ? 29. What is Setup module ? what it does ? 30. What is register & debug in Ansible ? 31. What is changed\_when in Ansible ? 32. Can we disable automatic facts gathering in Ansible ? 33. How error handling can be done in Ansible ? 34. How to ignore failed commands in Ansible ? 35. What is handlers ? Why we use Handlers in Ansible ? 36. What is Privilege Escalation in Ansible ? 37. Task to connect(SSH) Ansible to remote host using another user & 38. run the playbook to the remote host using with another user ? 39. What is Ansible vault ? 40. How to decrypt a vault file ? 41. How to encrypt a string in Ansible using Ansible Vault ? 42. If a string is encrypted in a file with a password then how to pass 43. the password using parameter while decrypting ? 44. If a file is encrypted using password & password is stored in a file 45. how to pass the file to decrypt the file ? 46. If a file is encrypted using password & password is also encrypted 47. then how to provide the password while decrypting the file ? 48. What is Ansible galaxy ? 49. What is Tags in Ansible ? Why it is used ? 50. What is lookup in Ansible playbook ? 51. How to control the command failure in Ansible ? 52. How to debug your playbook ? 53. What is diff mode ? 54. What is Dry Run in Ansible & how to do that ? 55. What is pre task & post task ? 56. How you can run your all tasks at once ? 57. What is block in Ansible ? 58. What are different variable scopes ? 59. How variable precedence takes place ? 60. Difference between include & import ? 61. How to include custom modules in Ansible ? 62. Describe the role directory structure ?   Ansible Task  Task 1  Part 1. Write Ansible playbook to automate Jenkins deployment  Part 2. Write Ansible role to install Docker & setup Kubernetes cluster  Automate the pipeline creation in Jenkins to create docker container & deploy on Kubernetes cluster  Task 2  Write ansible playbook for below tasks:  1. Install apache server and deploy sample html application  2. Create /var/www/example.com  3. deploy a sample application to the above directory  4. create a virtual host for deploy application and set it as default virtualhost  1. What is inventory file used for and default inventory host location?  The Ansible inventory file defines the hosts and groups of hosts upon which commands, modules, and tasks in a playbook operate. The file can be in one of many formats depending on your Ansible environment and plugins. The default location for the inventory file is /etc/ansible/hosts. If necessary, you can also create project-specific inventory files in alternate locations.  2. What is ansible configuration file used for and its default path?    If you want to modify some settings in ansible we will use ansible configuration file. The default path will be /etc/ansible/ansible.cfg    3. Do you write your inventory file?  Yes  4. How many types of variables and precedence?  There are almost 16 variable precedence in ansible where extra vars have highist precedence  5. Write the command to find the python version on nodes?  6. What is the file structure of ansible roles?  Ansible roles can be as complex or as simple as you need. Sometimes, it is helpful to start simple and iterate into a more complex role as you shore up the base functionality.  7. What happen when one node or instance is unreachable?  8. What happen when one task is failed in playbook?  9. I have 20 servers , I want to install one package in 5 servers and other package in next 5 servers..like that How to write in ansible script for that…Explain?  10. What is the architecture of ansible?  11. How do I supply variables while executing the playbook in ansible?  12. Explain about the tags in ansible?  13. How to execute failure playbook again?  14. How is ansible diffrent than any other configuration management tool?  15. Ansible playbooks are written in what format?  16. What is a module in Ansible?  17. What is inventory used for in ansible?  18. Name three places where ansible variables can be stored?  19. What connection anible establishes with linux and windows node.  20. Diffrence between Remote and local execution in ansible.  21. What is the purpose and location of ansible.cfg file.  22. Name any two settings from anible.cfg file.  23. Please write below a sample inventory file with host, group & group of groups syntax in it.  24. What is the diffrence between group\_vars & host\_vars directory?  25. What are adhocs commands used for & write below syntax of a ad hocs command?  26. Write below ad hoc command to gather fact varibales on all the hosts from the inventory file?  27. What format does ansible ad hoc command returns the output?  28. Name three types of modules in ansible?  29. Name any 10 modules in ansible that you have used.  30. How to list all the ansible core modules from command line.  31. How to display all the options/attributes for apt module from command line.  32. How can you check mandatory option for any module from command line?  33. What is setup module used for?  34. Write down sample global play declaration  35. Write down any two tasks from playbook with its proper format and names.  36. Whats is the diffrence between sudo and become module and its purpose.  37. Write down playbook syntax of starting ntp service on webserver and dbservers host group at once.  38. How to take user input from a playbook?  39. What is debug module used for in playbooks?  40. How to store output of any task into a varible from playbook?  41. What are handlers used for in ansible and how is it different from task?  42. Conditional execution in ansible is used for what purpose and write down its syntax with small description?  43. What are templates used for and its format?  44. What are ansible roles and its purpose?  45. Command to generate ansible roles directory structure.  46. Name 5 directories from ansible roles.  Ansible interview Questions  47. What is {{ }} symbol used for in ansible.  48. Write down all the places where we can define variables according to its precedence.  49. Write down syntax of dictionary variable?  TRUE or FALSE  50. Setup module gets executed after executing any ad hoc command.  True  False  51. Anible playbooks are written in XML format.  True  False  52. ansible.cfg file is list of plays.  True  False  53. Fact variables is generated by ansible and we do not need to create fact varibales.  True  False  54. Anible is written in Python langueage.  True  False  55. Ansible works on Master and client relationship, where ansible control server package is installed on master and ansible client package is installed on all the clients.  True  False  56. Ansible loop module is with\_items.  True  False  57. All the templates in ansible roles has to be defined in main.yml file inside templates directory.  True  False  58. All the roles uploaded in ansible galaxy website is written and owned by Ansible inc(organization).  True  False  59. Ansible can only get installed on Linux machine but can manage Linux and windows nodes.  True  False  60. Write a Ansible task that can copy the file to remote location with the ownership of Jboss?  61. What is local action in Ansible?  62. What are the roles in Ansible?  63. Write a playbook for installation of apache in ubuntu and centos?  64.How to create a role in ansible?  65. Different types of inventories in ansible?  66. if we want system information of machine how we will get the data with ansible like we use facter in puppet ?  67. In which language we write playbooks in ansible?  68. Activity that we use with ansibles or advantage Functionality or purpose of ansible using in your project?  69. How to launch an LDAP server using ansible?  70. How ansible works.. and what are the playbook written other than the basic playbooks ?  71. For example there are 4 aps server running and 4 websers running and we have a new code change and this change should automatically copy to these different servers using ansible, how I will come to know new code is generated.  Use Jenkins with POLL SCM option and after artifact is generated using ansible to push suing copy module.  72. Have u done any automation like app goes down then for self healing of the app kind of the thing. 73 .Have u written any playbook. Explain it  74. what is a dynamic inventory in ansible?  75. How can you manage cloud services with ansible?  76. How can you protect sensitive information in ansible?  77. How can you save the output of module execution in ansible?  78. How can you manage error handling?  79. what are conditionals in ansible?  80. write a playbook to create AMI on AWS?  81. How can you the fact varibales in ansible?  82. can we create an ansible module ?  83 . How can you do provisioning with ansible?  84 . BASH Vs Python Vs Configuration Management Tools?  85. How can you update a single table in a Database with ansible?  86. What type of roles you used in your playbook explain ?  87. How to create and use the Dynamic inventory ?  88. Write a playbook for installing AppDynamics and configure the agents for the same ?  **Topic- Dockers and Containers**  1) What is docker why we are using docker difference between vm?  2) What is docker cloud & how it is different from docker hub what is the features over docker hub ?  3) What is docker compose ?  4) How you will link when the docker containers is in different virtual machine is there any configuration in docker compose file are any command or any variable?  5) What you will do if one master got corrupted, can we create multiple masters?  6) What you will do in case any pod deleted?  7) What is namespaces in kubernetes can you tell me some?  8) Can you tell me some commands using in kubernetes ?  9) Tell me the command to create cluster?  10) Difference between rc and rs?  11) What is kubernetes?  12) What is the difference between kubctl & kops?  13) Why you are using kubectl can you explin why we are using ?  14) Why we using kops?  15) What is the difference between docker cloud and docker swarm?  16) How to attach a volume in cluster at some time the container will be deleted then rs will re-create new container then how to attach that container automatically and how to restore the volume automatically to re-created container?  17) How many projects you used kubernetes?  18) N number of docker containers deployed to different vms how will you manage there is no kubernetes installed?  19) If installed kubernetes how you will deploy this containers into kubernetes cluster?  20) Can you tell me the command for creating kubernetes cluster in vm?  21) How many nodes we required to create kubernetes cluster?  22) We have nearly 15 nodes in my organization all are decentralized so which node I need to create as a master? Is their any possibility to make all the machines as masters?  23) Our applications are decentralized I don’t want distributed environment if any thing happens to the master all will collapse , can we create multiple masters?  24) What is the difference between kubectl and minikube?  25) If any container down in my cluster how you will rectify?  26) How do you manage your application secrets in k8s?  Echo $0  27) How to write a script when the first command is executed then execute the below script?  A) Java --version  If [ $? –eq 0 ]  Then  Echo “print the variable ”  Else  Echo “prin the variable”  fi  28) How to print the exact file name by using command ?  A) First assign path as a variable  awk -F '/' '{print $(NF-1)}' <<< "$a"  29) Can you tell me the syntax for for loop and while loop?  30) How to dictionary in python?  A) Dict={a:10,b:10}  31) How to add another variable c, with key 10 to the above dictionary ?  32) How to replace existing dictionary?  Dict=[c:10]  33) How to print shell name?  34) How to assign a all the arguments to a single variable?  35) How to print the current processid of current shell?  36) How to know the file that is entering randomly to my script?  37) How to divide two variables in shell script?  38) What is trap?  39) What is shift in shell script?  40) How to run our script in background?  41) How to know the running back ground process id ?  42) What is $\*, $$ and $@?  43) How to print only directories?  44) How to print the directory only started with number?  45) How to grep two strings at a time?  46) How to grep a string that is started with some string and ends with some string like a…..b ?  47) How to print string that starts with a?  48) Did you worked on arrays?  49) How will you give access of your script to a particular user?  50) How to access background running scripts and their pid?  51) How to run our script in foreground?  1) Current roles & responsibilities?  2) What is Docker compose?  3) What is Docker server version?  4) What are the advantages of Docker?  5) How do you setup Docker in Jenkins in production environment?  6) How do create Docker image from Docker file?  7) How do you deploy the Docker image generated in the testing environment to  production environment in Jenkins?  8) What is Docker data centre?  9) What is Docker hub & uses?  10) What are the types of Docker networks?  11) How do you define network in the Docker compose file?  12) What are the basic parameters required in the Docker compose file?  13) What is Docker interlock?  14) What is overlay networking?  15) How to communicate between 2 containers present in separate network?  16) How to store the data present in the Docker container in the AWS?  17) If we define the Docker volume in the docker compose file is it possible to share  data with the EFS, NFS?  18) Differnce between image and container  19) How to Run containers  20) Why we need to mention dual ports(8080:8080) in docker run command  21) Differnce between Copy and Add  22) How to write a docker file to deploy a war file    i. We have private repository and don’t have base images .how can deploy a war file?  ii. Write a simple docker file to deploy a war file by using base images?  23) Difference between docker compose and docker swarm  24) Why we need to use docker compose?  25) Tell me about Docker Network  26) What is the difference between Docker Swarm ,Compose and Kubernetes ?  27) How does the containers communicate when the containers are running in different hosts?  28) What is the difference between CMD and RUN  29) How can you transfer the data from one container to another ?  30) How do you monitor your containerized applications?   1. Whats is docker ? 2. Difference between container & VMs ? 3. Difference between Docker & Virtualization ? 4. Difference between container and image ? 5. How image builds ? 6. What are image layers ? 7. How image layers work ? 8. What is overlayfs ? 9. Where the image layers can be found in which directory ? 10. How can we check the content of each layer ? 11. How to check the layers stacked with image ? 12. What is Union Mount & AUFS ? 13. Why use Union mount system for Docker ? 14. What are the 3 different directories in /var/lib/docker/aufs ? 15. How to run an image ? 16. How to tag an image ? 17. How to Link one container with another ? 18. How do you sequence the containers? A first then B should execute after that ? 19. How to create a volume in docker container to store data ? 20. How to mount a local directory into a container ? 21. How to expose a port no to access container ? 22. What is entrypoint in docker ? 23. What is dockerfile ? 24. Difference between ADD & COPY parameters in dockerfile ? 25. How to create a bridge in container ? 26. How a container gets an internal IP ? 27. Can we check the process of a container inside as well as outside the container ? 28. Can we check the container process on docker host ? 29. How kernel isolates to run the container and how resources managed by the kernel ? 30. What is namespace and cgroups ? 31. What is docker-compose and docker-swarm ? 32. How you can give different network IP to the container ? 33. What are the parameters of dockerfile ? 34. Is there any windows container also available ? 35. How to stop a container ? 36. How to run a container in background ? 37. How to go inside a container if container is running in background ? 38. How to check running containers ? 39. How to remove an image ? 40. How to run an image which is in tar format ? 41. Command to check the process of a container ? 42. How to check resource utilisation of a container ? 43. How to create an image ? 44. How to save changes of a container ? 45. What are registries ? 46. Difference between docker commands: up, run & start ? 47. Can we run more than one process in a container ? 48. How to remove a paticular layer from a container ? 49. Give me a command to remove all exited conatiners ?   Docker Task  Part 1. Write a Docker file to create a Docker image which should have Wordpress installed  Part 2. Write a Docker file to create a Docker image for database  Now, use Docker compose to bring up the above Docker images as containers. Database container should mount the local host's “/etc/mysql” volume into it's (containers) /etc/mysql directory. |

**Topic- Kubernetes**

What is Kubernetes ?

1. What are Kubernetes Components ?
2. What is etcd ?
3. What is master & minion ?
4. How to make quorum of cluster?
5. What is Replication controller & what it does ?
6. What is ingress ?
7. What is RBAC and how can you configure RBAC in a k8's Cluster ?
8. Difference between Kubernetes & Docker Swarm ?
9. How can you rollbck the previous version of application in Kuberntes?
10. Scenario: There are 2 tables, emp, empsal if there schema changes,
11. How does that deployment happens into containers/POD automatically?
12. How does container know that application is getting failure ?
13. Difference between nodeport, clusterIP, load balancer & ingress ?
14. What is kubectl & kubelet ?
15. What is the use of Kube-controller manager ?
16. What is pod ?
17. How many containers can run in a pod ?
18. How many containers can be launched in a node ?
19. What is the role of Kube-Scheduler ?
20. How the 2 pods communicate with each other ?
21. How 2 containers inside a pod communicate with each other ?
22. What is Flannel & why we use it ?
23. Difference between Flannel & Calico ?

**Topic-Openshift**

1. What is Openshift ?
2. Difference between Openshift & Kubernetes ?
3. What is Services Layer ?
4. How to expose a service in Openshift ?
5. What are the 3 components of any created project ?
6. What is router in default or in any project while creating project ?
7. What do you mean by identity provider in Openshift?
8. How do you create a New user identity ?
9. Where is the user identity located ?
10. What is project in Openshift ?
11. What are the types of permissions/role bindings in Openshift ?
12. How to check the permission of user ?
13. How to describe anything in Openshift ?
14. How to check no of projects ?
15. How to assign a role/permission to a user ?
16. What is clusterrolebinding in openshift ?
17. What is the process/working of POD creation ?
18. What is Builder POD ?
19. What is deployer POD ?
20. How to create a New application POD ?
21. How to check logs of POD ?
22. What is Deployment Configuration & why we need DC ?
23. What is SVC & why we need SVC ?
24. What is RC (Replication Controller) ?
25. How to check DC of POD & how to edit DC ?
26. How to create route ?
27. How to expose svc ?
28. How to do rollout ?
29. How to increase replica ?
30. What is Source to Image in Openshift ?
31. What is builder image ?
32. What are the process to create source to image ?
33. How to give the Cluster role/permission to the user ?
34. How to create secure route ?
35. What is PV & PVC ?
36. What are access modes in PV ?
37. What is node selector ?
38. What are the two regions in projects ?
39. Difference between template & Deployment Configuration ?
40. How to migrate whole cluster to another ?
41. How to manually migrate container ?

**Topic- AWS**

1. What is Amazon RDS ?
2. What is EC2, S3, EBS ?
   1. What is VPC & why we require to create VPC ?
   2. Is is possible to scale an Ec2 Instance vertically ?
   3. How is Amazon RDS, Redshift & DynamoDB different ?
   4. How is a spot Instance different from an On-demand Instance ?
   5. How Infrastructure As Code processed & executes in AWS ?
   6. If your Linux-build server getting slow down, what will you do to check ?
   7. Types of EBS storage ?
   8. How to backup a running instance ?
   9. How to secure s3 bucket ?
   10. What are the security available for users to access S3 ?
   11. How to create AMI ?
   12. What are the main components of CloudFormation ?
   13. What is mapping in cloudformation template ?
   14. How is YAML different from JSON ?
   15. Different types of ELB ?
   16. What is autoscaling group ?
   17. Which type of ELB is good for application load ?
   18. What is difference between application load balancer & classic load balancer ?
   19. What is metrics in cloudwatch ?
   20. Is it possible to recover your lost private key ?
   21. How can you connect your EC2 Instance if you lost your key ?
   22. While connecting to your EC2 instances, what are the possible connection issues one might face ?
   23. What is Subnet & how many subnets are there in a VPC ?
   24. Why do we make subnets ?
   25. What is routing table ?
   26. How you can connect a private subnet with a public subnet ?
   27. Can VPC peering possible in two different region ?

AWS Task

Task 1

Write a script which will based on “Number of requests” metric of the ALB/ELB scale up webapp EC2 instances under the Load Balancer, increase AWS Elasticsearch Nodes count, and change the instance size of a MongoDB EC2 instance from m4.large to m4.xlarge. (without using ASG).

Task 2

Architecture Diagram for a PHP/JAVA/Python based application to be hosted on AWS with all mentions like VPC, AWS/any other cloud platform services, well defined network segregation.

**Topic- Scripting {shell/python}**

Scripting (SHELL/Python)

Shell Task

Task1

Bash script to setup a whole LAMP stack, PHP app can be Wordpress and DB can be MySQL. This script should install all components needed for a Wordpress website.

We should be able to run this script on a local machine or server and after execution of the script it should have Wordpress Running via Nginx/Apache.

DB user for Wordpress should also be made automatically from within the script and same should be set in Wordpress conf file.

Task 2

Bash script to setup a whole JAVA application stack on a server.

This script should install all components needed for a Java/Grails application.

Once the script is run it should have the java application running and being served via Nginx on local machine or server. Sample java application can be simply a tomcat war etc

**Topic- Linux-RHEL**

1. What is Linux ?
2. What are Linux OS Flavors ?
3. Difference between Debian & RPM based OS ?
4. What is Kernel ?
5. Explain the boot process of Linux OS ?
6. How is RHEL different from CentOS ?
7. What is the Latest version of RHEL ?
8. What is Grub ?
9. Difference between Grub & Grub2 ?
10. What is boot loader ?
11. Do you think the boot process in RHEL 7 is faster than RHEL 6 ? If yes, How ?
12. What is .rpm & .deb ?
13. What is RPM ?
14. What is YUM ?
15. Different methods to install the rpm based packages ?
16. What is Bash ?
17. What is SHell ?
18. How many types of SHells are there ? \* Explain the daily used basic commands like cp, mv, rm ?
19. What is the significance of touch command ?
20. In how many ways you can create a file ?
21. How to delete the content from a file ?
22. Explain the process/work behind hitting the google.com ? how you access google.com ?
23. How many types of permissions are there ? What is chmod ?
24. What is sticky bit ?
25. What is ACLs ?
26. What is SetGID, SetUID & Stickybit ?
27. Location where all the user information are stored ?
28. File where user password are stored ?
29. What is the default permission of a file ?
30. What is the significance of -rvf ?
31. What is PV, VG & LV ?
32. What are the types of file system ?
33. What is XFS ?
34. Can we reduce XFS file system ?
35. How can we extend LV ?
36. Command to check running process ?
37. Command to check RAM usage ?
38. Command to check Disk usage ?
39. Difference between ps -aux & top command ?
40. What are the ways to check CPU usage ?
41. How to check CPU details ?
42. Explain the steps to create a partition & how to format with file system ?
43. Explain the steps to create LV ?
44. Explain steps to reduce XFS & EXT files systems ?
45. Significance of .bashrc file ?
46. How you check the kernel version ?
47. How you check the Red hat release version ?
48. Significance of resolv.conf file ?
49. What is DNS ? How you resolve DNS ? Types of DNS records ?
50. Difference between Nginx & HTTP Server ?
51. Port no of HTTP, FTP, SSH, HTTPS ?
52. What is SSH ? How you generate SSH-keys ?
53. What is Private & public key ? How they authenticate ?
54. Configuration file of SSH ?
55. Configuration file of HTTP ?
56. What is Virtual Hosting ? How you configure virtual hosting ?
57. Explain ifconfig command ?
58. Difference between IPv4 & IPv6 ?
59. What is MAC address ? can we change the physical address ?
60. How to check system uptime ?
61. How to check memory information ?
62. What is SWAP ?
63. What is the exact memory free in your system ?
64. What is cache memory ?
65. What if you can do rm -rvf / ?
66. Kinds of permission in Linux ?
67. What is vim & vi ?
68. What is pipe | ?
69. What is grep command ?
70. What Find command does ?
71. How to redirect commands output ?
72. What is systemd in Linux ?
73. What does systemctl do ?
74. If you run a command like nautilus in terminal, whether it will block your terminal or not ?
75. If yes, whats the solution of this to not to unblock the terminal without closing the command application?
76. What is rsyslog ?
77. What is SSH-tunnel ?
78. How to set history size ?
79. How to extend VG ?
80. What are logical & extended partitions ?
81. Explain the steps to reset root password at boot time ?
82. What are run-levels ? How many types of run levels are there ?
83. How we change the run level ?
84. How to check the logs ?
85. Difference between Journalctl & tail command ?
86. What does the subscription-manager do ?
87. How to archive a file ?
88. What is umask ?
89. How to kill a process ?
90. How to assign IP address manually ?
91. How to assign static IP address to a system ?
92. Explain the different types of Linux process states ?
93. What is a Zombie process ?
94. What is KVM ?
95. What is hypervisor ?
96. Difference between MBR & GPT ?
97. How you can mount a file system permanently ?
98. What is cron ? How to setup a cron job ?
99. What is Kickstart ?
100. How to create a network bridge in Linux ?
101. Difference between iptables & firewalld
102. What is SElinux ?
103. What is ISCSI & targetcli ?
104. Difference between NFS & SAMBA ?
105. What is nfsnobody ?
106. What is SSHFS ?
107. What is Kerberos ?
108. How to secure NFS with Kerberos ?
109. What is the difference between telnet & SSH ?
110. What is DHCP ?
111. What is Kickstart file ?
112. What is NTP Server ? How to configure NTP ?

1. script/command to delete last word from every line in a file

awk '{gsub("[a-zA-Z0-9]\*$", "");print}' <filename>

To replace last word with hello in every line

awk '{gsub("[a-zA-Z0-9]\*", "hello");print}' <filename>

2.script/command to find the files with more than 1gb size

find <path for directory> -size +1G -type f

3. What is Swap Space?

4. What is the maximum length for a file name in Linux?

5.Which partition stores the system configuration files in Linux system?

6. Which command is used to uncompress gzip files?

7. What is the difference between soft and hard mounting points?

8. What are the file permissions in Linux?

9. more questions are from sed, find and awk .

10. How to check Memory stats and CPU stats as a Linux admin?

11.How to reduce or shrink the size of LVM partition?

12. How can you enhance the security of password file?

13. What is the difference between Cron and Anacron?

14. What command is used to check the number of files, disk space and each user’s defined quota?

15. how can you manage memory in linux machines?

16. 18. What is the name and path of the main system log?

17. how to manage logical volumes?

18. Explain /proc filesystem?

19. What are the fields in the/etc/passwd file?

20. How do you terminate an ongoing process?

21. How can you know the execution time of a command?

22. How can you append one file to another in Linux?

23. How you can run an Linux program in the background simultaneously when you start your Linux Server?

24. How to check previous command is success or not ?

25. Give me a command to create a file with specific size ?

26. How can you mount an extra disk to an existing filesystem ?

**Topic- Monitoring**

1. Why we use monitoring ?
2. What are the different tools & technologies for monitoring ?
3. How we monitor our applications & servers differently ?
4. What is Prometheus ? How is different from other monitoring tools ?
5. What is ELK stack ?
6. Why we use Grafana ?
7. How we query different outputs in Prometheus ?
8. What type of graph can be implemented in Grafana ?