- There will be one controller machine (ansible server) and 5 managed nodes.
- You will be provided with the root login and root password, and you have to create playbooks via vikasnehra user.
- You have to login on controller node with root user account then switch to vikasnehra user account.
- Create all your playbooks in the /home/vikasnehra/ansible directory if the playbook path is not well-defined in the question.
- Create ansible.cfg and inventory file in the /home/vikasnehra/ansible directory
- Create a roles directory in the /home/vikasnehra/ansible directory.

Question 1:

1. Install and configure Ansible on the control node (ansible-server):
a) Install the required packages.
b) Create a static inventory file called /home/vikasnehra/ansible/inventory so that:
i) node1 is a member of the dev host group.
il) node2 is a member of the test host group.
ill) node3 and node4 are the members of the prod host group.
iv) node5 is a member of the balancers host group.
c)The prod group is a member of the webservers host group.
d) Create a configuration file called /home/vikasnehra/ansible/ansible.cfg so that:
1) The host inventory file is /home/vikasnehra/ansible/inventory
il) The default content collections directory is /home/vikasnehra/ansible/mycollection
iii) The default roles directory is /home/vikasnehra/ansible/roles

Question 2:

2. Create and run an Ansible ad-hoc command. As a system administrator, you will need to install software on the managed nodes:
a) Create a shell script called yum-repo.sh that runs Ansible ad-hoc commands to create the yum repositories on each of the managed nodes as per the following details:
b) NOTE: you need to create 2 repos (BaseOS & AppStream) in the managed nodes.
BaseOS:
name: BaseOS
baseurl: file:///mnt/BaseOS/
description: Base OS Repo
gpgcheck: yes
enabled: yes
key: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
AppStream:
name: AppStream
baseurl: file:///mnt/AppStream/
description: AppStream Repo
gpgcheck: yes
enabled: yes
key: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release

Question 3:

3. Create a playbook called /home/vikasnehra/ansible/packages.yml that:
a) Installs the php and mariadb packages on hosts in the dev, test, and prod host groups only.
b) Installs the RPM Development Tools package group on hosts in the dev host group only.
c) Updates all packages to the latest version on hosts in the dev host group only.

Question 4:
4. Install the RHEL system roles /home/vikasnehra/ansible/timesync.yml that:
a) Runs on all the managed hosts.
b) Uses the timesync role.
c) Configures the role to use the time server 172.25.254.250
d) Configures the role to set the iburst parameter as enabled.
Question 5:
5. Create a role called apache in /home/vikasnehra/ansible/roles with the following requirements:
a) The httpd package should be installed, httpd service should be enabled on boot, and started.
b) The firewall is enabled and running with a rule to allow access to the web server.
c) A template file index.html.j2 exists (you have to create this file) and is used to create the file /var/www/html/index.html with the following output: Welcome to HOSTNAME on IPADDRESS where HOSTNAME is the fully qualified domain name of the managed node and IPADDRESS is the ip address of the managed node.
Question 6:
6. Use Ansible Galaxy with the requirements file called /home/vikasnehra/ansible/roles/requirements.yml
to download and install roles to /home/admin/ansible/roles from the following URLs:
a) https://galaxy.ansible.com/download/zabbix-zabbix-1.0.6.tar.gz
The name of this role should be zabbix.
b) https://galaxy.ansible.com/download/openafs_contrib-openafs-1.9.0.tar.gz The name of this role should be security
c) https://galaxy.ansible.com/download/mafalb-squid-0.2.0.tar.gz The name of this role should be squid.
Question 7:
7. Create a playbook called squid.yaml as per the following details:
a) The playbook contains a play that runs on hosts in the balancers host group and uses the squid role present in your machine
Question 8:
8. Create a playbook called test.yml as per the following details:
a) The playbook runs on managed nodes in the test host group.
b) Create the directory /webtest with the group ownership webtest group and having the regular permissions rwx for the owner and group and rx for the others.
c)Apply the special permissions: set group ID
d)Symbolically link /var/www/html/webtest to /webtest directory.
e) Create the file / webtest/index.html with a single line of text that reads: Testing.

Question 9:
9. Create an Ansible vault to store user passwords with the following conditions:
a) The name of the vault is vault.yml
b) The vault contains two variables, dev_pass with value as redhat and mgr_pass with value as linux respectively.
c)The password to encrypt and decrypt the vault is nehraclasses
d) The password is stored in the file /home/vikasnehra/ansible/password.txt file.
Question 10:
10. Generate hosts files:

a) Download an initial template file called hosts.j2 from the below URL: http://classroom.example.com/content/hosts.j2 to /home/vikasnehra/ansible/directory. Complete the template so that it can be used to generate a file with a line for each inventory host in the same format as /etc/hosts.
b) Create a playbook called gen_hosts.yml that uses this template to generate the file /etc/myhosts on hosts in the dev host group.
c) When completed, the file /etc/myhosts on hosts in the dev host group should have a line for each managed host:
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.10. nodel.example.com node1
192.168.10.y node2.example.com node2
192.168.10.z node3.example.com node3
192.168.10.a node4.example.com node4
192.168.10.b nodes.example.com node5

Question 11:

11. Create a playbook called hwreport.yml that produces an output file called /root/hwreport on all of the managed nodes with the following information:
a) Inventory hostname
b) Total memory in MB
c) BIOS version
Each line of the output file contains a single key-value pair.

Question 12:

12. Create a playbook called /home/vikasnehra/ansible/issue.yml as per the following requirements:
a)The playbook runs on all inventory hosts.
b) The playbook replaces the contents of /etc/issue with a single line of text as:
i. On hosts in the dev host group, the line reads: Development
ii. On hosts in the test host group, the line reads: Test
iii. On hosts in the prod host group, the line reads: Production;

Question 13:

13. Rekey an existing Ansible vault as per the following conditions:
a) Use the vault.yml file which you have created earlier.
b) Set the new vault password as ansible.
c) The vault remains in an encrypted state with the new password.

Question 14:

14. Create user accounts. A list of users to be created can be found in the file called user_list.yml which you should download from "http://classroom.example.com/content/user_list.yml" and save to /home/vikasnehra/ansible/directory. Using the password vault created elsewhere in this exam, create a playbook called create_user.yml that creates user accounts as follows:
a) Users with a job description of developer should be created on managed nodes in the dev and test host groups assigned the password from the dev_pass variable and is a member of supplementary group devops.
Users with a job description of manager should be created on managed nodes in the prod host group assigned the password from the mgr _pass variable and is a member of supplementary group opsmgr.

c)Passwords should use the SHA512 hash format. Your playbook should work using the vault password file created elsewhere in this exam.

Question 15:

15. Configure cron jobs:
Create /home/vikasnehra/ansible/cron.yml playbook as per the following requirements:
a) This playbook runs on all managed nodes in the hostgroup.
b) Configure cronjob, which runs every 2 minutes and executes the following commands: logger "EX294 exam in progress" and run as user natasha.

Question 16:

16. Create & use a logical volume:
Create a playbook called /home/vikasnehra/ansible/lvm.yml that runs on all the managed nodes and does the following:
a) Creates a logical volume with the following requirements:
i.The logical volume is created in the volume group.
ii.The logical volume name is data.
ili.The logical volume size is 1200 Mib.
iv. Format the logical volume with the ext4 file-system.
v. if the requested logical volume size cannot be created, the error message "could not create logical volume of that size" should be displayed and size 800 MiB should be used instead.
vi. if the volume research does not exist, the error message "volume group does not exist" should be displayed.
vii. Don't mount the logical volume in any way.

Question 17:

17. Create and use partitions:
Create /home/vikasnehra/ansible/partition.yml, which will create partitions on all the managed nodes:
a) After vdb creating a 1200M primary partition, partition number 1, and format it into ext4 filesystem and mount it under /srv
c) If there is not enough disk space, give prompt information "Could not create partition of that size" and create a 800M partition.
d) If vdb does not exist, a prompt message will be given "this disk does not exist."

Question 18:

18. Using a selinux role create a selinux.yml playbook with the following conditions:
a) Configure on all managed hosts to set the default selinux mode as permissive.
b) Verify the selinux mode on all the nodes using ansible ad-hoc command.
c) Create another copy of the selinux.yml playbook with the name as selinux2.ml and make changes there in it to configure the selinux default mode as enforcing for all the managed nodes.
d) Execute the selinux2.ml playbook using ansible navigator.
e) Verify the selinux mode on all the node machines.