AAP QQ

Q1.

configure git in the control machine user name is 'gituser', user

email id 'gituser@example.com' push default method is 'simple'

Q2.

create a project user_project from http://git.lab.example.com:8081/git/user-project.git .

playbook is already created there with name create_user.yml with some task.

Users will create from a file user_list.yml where some user names are there with a group.

Add user greg to a group developer. Add node3 to a group prod inside the inventory and run the playbook.

After completing all tasks and configurations then push the changes to the git server.

Q3.

Install alias httpd install.clone the project name web_install from http://git.lab.example.com:8081/git/alias_httpd.git.

There is a file alias.conf with some data.playbook was created with a copy task.

if the task was executing httpd service should restart if not httpd service should not start.

After completing all tasks and configurations then push the changes to the git server.

Q4.

create a tag based playbook.

playbook name tags.yml pull from
http://git.lab.example.com:8081/git/tag-project.git.

Add 'hello' to /var/www/html/index.html in 'dev' host group with tag 'alpha'.

Add 'by by' to /var/www/html/index.html in same 'dev' host group with tag 'beta'.

if any tag is not given to the playbook then it should not execute any task.

After completing all tasks and configurations then push the changes to the git server.

Q5.

Tune ansible configuration file.

pull git project from http://git.lab.example.com:8081/git/fork-project.git

Modify ansible.cfg file with parallelization size is 45 and gather facts no.

After completing all tasks and configurations then push the changes to the git server.

Q6.

Create users.clone the project from <git server> already having a user_list.yaml file with content

create a playbook create_user.yaml which will create the users with the following details.

a). users will be created with the name and proper uid.

b). In the GECOS part it will be the 'firstname middlename lastname' and the first letter should be capital. (eg John Micheal Bobby).
c). for every user it should create a 6 digit random password with SHA512 encryption. the salt value of the password will be stored in a file filename-<username> in the directory on which the playbook will be executed in the control node

After completing all tasks and configurations then push the changes to the git server.

Q7.

Install an ansible.netcommon collection from hub.domain11.example.com into /home/student/mycollections/

Q8.

Create a custom collection.

a. Collection name rhel.user
b. Create a role in its name mymotd.
c. Copy a yml task file to mymotd/task/main.yml file.
d. Copy a conf file to mymotd/files/ .
e. Publish to a private hub.

Q9.

Create a custom execution environment .

a) Base image: hub.lab.example.com/ee-minimal-rhel8:latest
b) Build image : hub.lab.example.com/ansible-builder-rhel8:latest
c) Use rhel.user custom collection created in the previous question.
d) TLS certificate available in /etc/pki/tls/certs/classroom-ca.pem
---

Q10.

Create a custom execution environment with a python package.

a. Python=3 and python-ldap=3
b. Base image -:
c. Build image :-
d. Publish to a private hub.

Q11.

Run playbook in an execution environment (created in 10th question) by script.

a. The git project needs to be cloned.
b. A config file and dynamic inventory script there.
c. Create a playbook with a copy content task and run using the above eei.
d. Push to git.

Q12.

Run a variable playbook.

a. Clone project from git url.
b. Config file dynamic inventory script available.
c. Create a playbook which copies content from the CONTENT variable to a FILE and the file is inside the DIRECTORY variable.

Q13.

Use a custom collection role.

a. Clone git project from url.

b. Config file and dynamic inventory.

c. Use a custom role from the rhel.user collection in a playbook

---

Q14.

Run playbook.

a. Clone project from git url.

b. Copy a file to /var/www/html/

c. Dynamic inventory and config file available

Q15. Create 2 projects in the controller

Q16. Create 2 inventory

Q17. Create an execution environment resource in the controller.

Q18. Create 3 templates using the execution environment and execute it

---