Let us explore the environment for our KodeKloud e-commerce LAMP stack application. There are 2 servers - lamp-web and lamp-db. Let us setup the inventory files for that. Create an inventory file at /home/thor/playbooks/lamp-stack-playbooks/inventory to include the following data:
Hosts: lamp-web, lamp-db
Groups: db_servers contains lamp-db; web_servers contains lamp-web
IP Addresses: lamp-web: 172.20.1.100; lamp-db: 172.20.1.101
Credentials for lamp-web: Username=john Password=john
Credentials for lamp-db Username=maria Password=maria

SOLUTION :
[thor@ansible-controller lamp-stack-playbooks]$ cat inventory
Inventory File
lamp-web ansible_host=172.20.1.100 ansible_user=john ansible_ssh_pass=john
lamp-db ansible_host=172.20.1.101 ansible_user=maria ansible_ssh_pass=maria
[web_servers]
lamp-web
[db_servers]
lamp-db

Let's add some additional data required for setting up the database and web servers. The data should be associated with the respective servers.
Database Info:
mysqlservice=mysqld
mysql_port=3306
dbname=ecomdb
dbuser=ecomuser
dbpassword=ecompassword
Web Info:
httpd_port=80

repository=https://github.com/kodekloudhub/learning-app-ecommerce.git

[thor@ansible-controller lamp-stack-playbooks]$ cat inventory
Inventory File
lamp-db ansible_host=172.20.1.101 ansible_user=maria ansible_ssh_pass=maria mysqlservice=mysqld mysql_port=3306 dbname=ecomdb dbuser=ecomuser dbpassword=ecompassword
lamp-web ansible_host=172.20.1.100 ansible_user=john ansible_ssh_pass=john httpd_port=80 repository=https://github.com/kodekloudhub/learning-app-ecommerce.git
[web_servers]
lamp-web
[db_servers]
lamp-db

Let us setup password less authentication between Ansible Controller and the web/db servers.
Create a pair of SSH keys for each user (without any passphrase) at /home/thor/.ssh/maria and /home/thor/.ssh/john
And distribute the public keys to the web and database servers - lamp-db and lamp-web.
DB server user is maria and its password is maria. Web server user is john and its password is john.

```
[thor@ansible-controller .ssh]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/thor/.ssh/id_rsa): /home/thor/.ssh/john
```

```
[thor@ansible-controller .ssh]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/thor/.ssh/id_rsa): /home/thor/.ssh/maria
```

ssh-copy-id -i /home/thor/.ssh/john/john.pub john@lamp-web
ssh-copy-id -i /home/thor/.ssh/maria.pub maria@lamp-db

Update the inventory file to use the newly created private keys for the respective hosts

[thor@ansible-controller lamp-stack-playbooks]$ cat inventory
Inventory File
lamp-db ansible_host=172.20.1.101 ansible_user=maria
ansible_ssh_private_key_file=/home/thor/.ssh/maria mysqlservice=mysqld mysql_port=3306
dbname=ecomdb dbuser=ecomuser dbpassword=ecompassword
lamp-web ansible_host=172.20.1.100 ansible_user=john
ansible_ssh_private_key_file=/home/thor/.ssh/john httpd_port=80
repository=https://github.com/kodekloudhub/learning-app-ecommerce.git
[web_servers]
lamp-web
[db_servers]
lamp-db

A playbook deploy-lamp-stack.yml is given with a basic tasks to install basic libraries. Execute the playbook and fix any issues.
You are not required to add any tasks or plays. Only fix the issue with execution.

[thor@ansible-controller lamp-stack-playbooks]$ cat deploy-lamp-stack.yml - name: Deploy lamp stack application
```
  hosts: all
  become: true
  tasks:
    - name: Install common dependencies
      yum:
        name:
          - libselinux-python
          - libsemanage-python
          - firewalld
        state: installed
```