In this lab exercise you will use below hosts. Please note down some details about these hosts as given below :

student-node :- This host will act as an Ansible master node where you will create playbooks, inventory, roles etc and you will be running your playbooks from this host itself.

node01 :- This host will act as an Ansible client/remote host where you will setup/install some stuff using Ansible playbooks. Below are the SSH credentials for this host:

User: bob
Password: caleston123

node02 :- This host will also act as an Ansible client/remote host where you will setup/install some stuff using Ansible playbooks. Below are the SSH credentials for this host:

User: bob
Password: caleston123

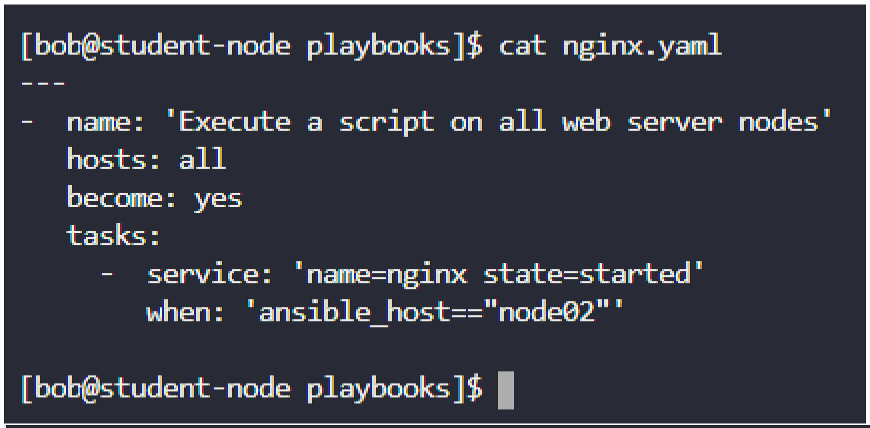
Which of the following Ansible built-in variable populates the flavour of the operating system?
ansible_os_family

Which keyword is used to define a condition in an Ansible playbook? when

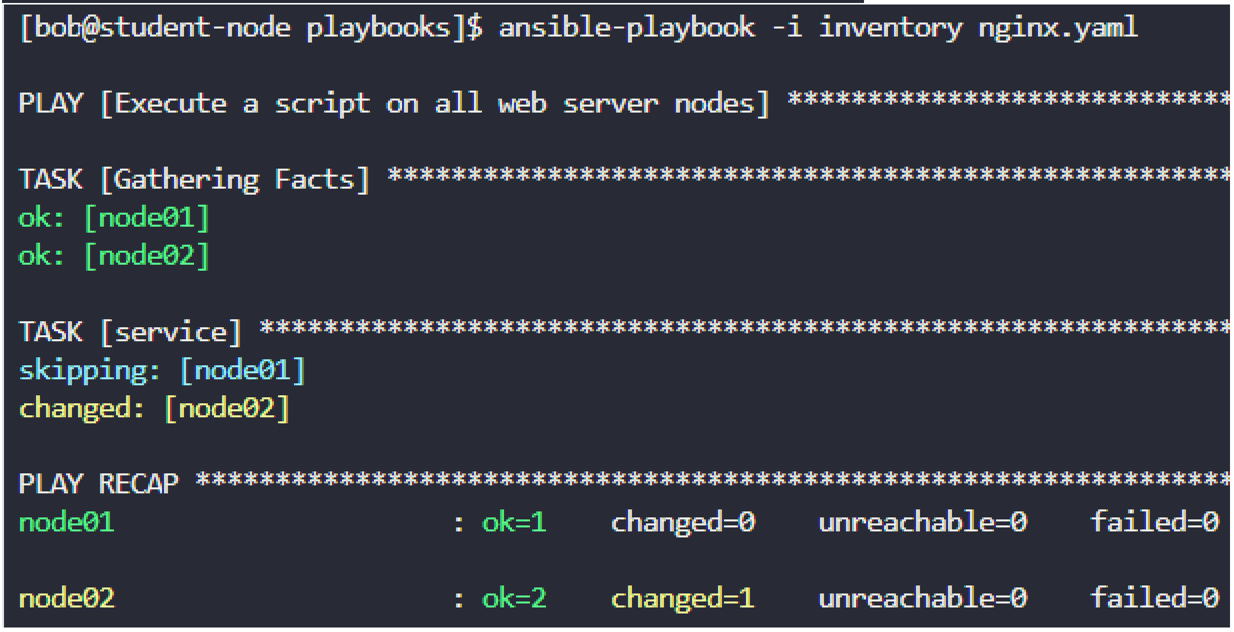
As per the given playbook, will Ansible install the vim package on a RedHat based machine? No

```
---
- name: Install package
  hosts: app1
  tasks:
    - name: Install
      package:
        name: vim
        state: present
      when: ansible_os_family != "RedHat"
```

There is a playbook named nginx.yaml under /home/bob/playbooks directory. It is starting nginx service on all hosts defined in /home/bob/playbooks/inventory inventory file. Use the when condition to run this task only on node02 host.

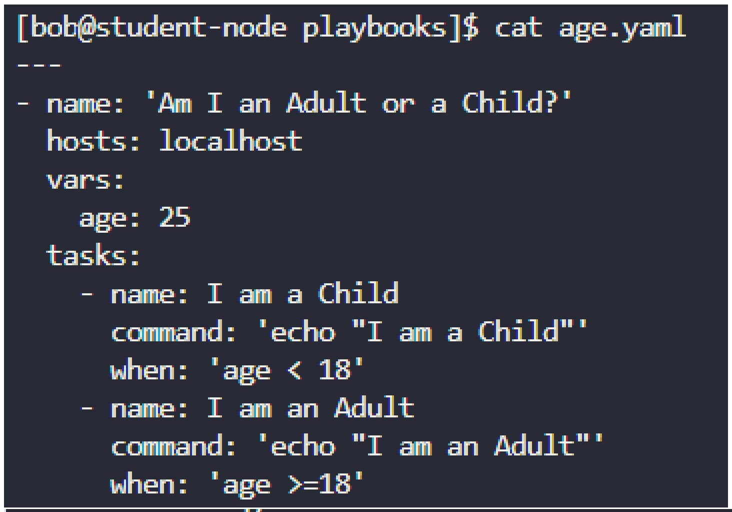


```
[bob@student-node playbooks]$ cat nginx.yaml
---
-  name: 'Execute a script on all web server nodes'
   hosts: all
   become: yes
   tasks:
     -  service: 'name=nginx state=started'
        when: 'ansible_host=="node02"'

[bob@student-node playbooks]$
```

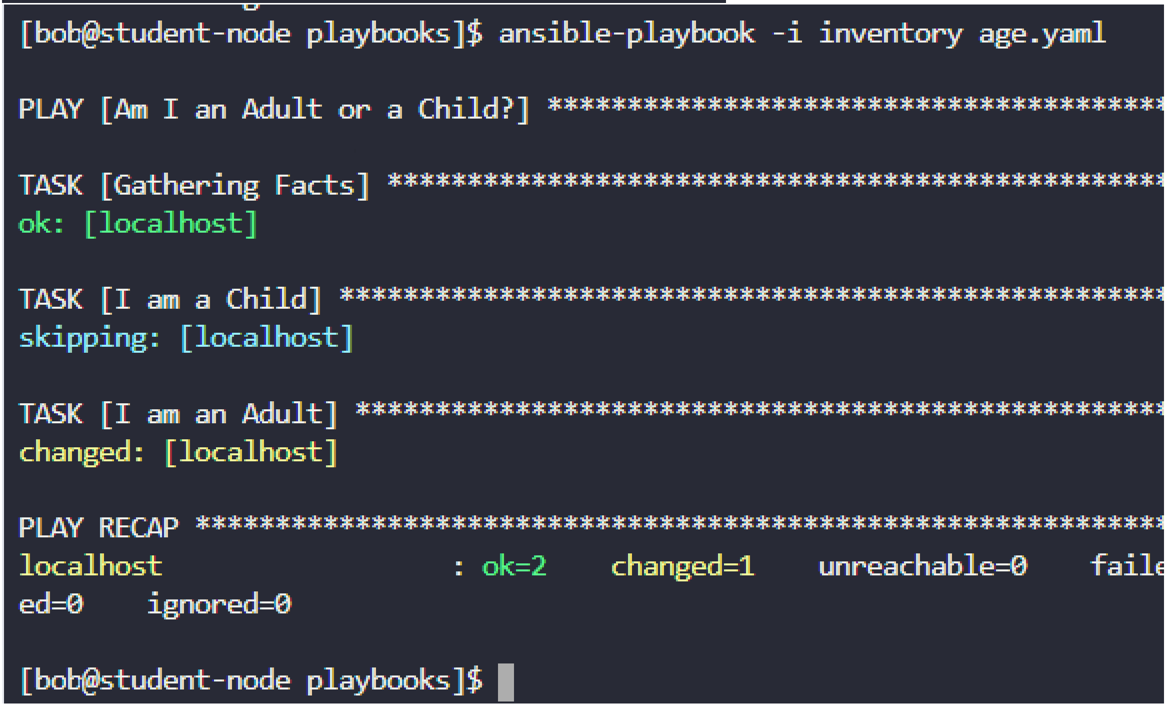


```
[bob@student-node playbooks]$ ansible-playbook -i inventory nginx.yaml

PLAY [Execute a script on all web server nodes] ****************************

TASK [Gathering Facts] *****************************************************
ok: [node01]
ok: [node02]

TASK [service] *************************************************************
skipping: [node01]
changed: [node02]

PLAY RECAP *****************************************************************
node01                     : ok=1    changed=0    unreachable=0    failed=0

node02                     : ok=2    changed=1    unreachable=0    failed=0
```

The playbook under /home/bob/playbooks/age.yaml , has a variable defined called age. The two tasks attempt to print if I am a child or an Adult. Use the when conditional to print if I am a child or an Adult based on whether my age is < 18 (Child) or >= 18 (Adult).



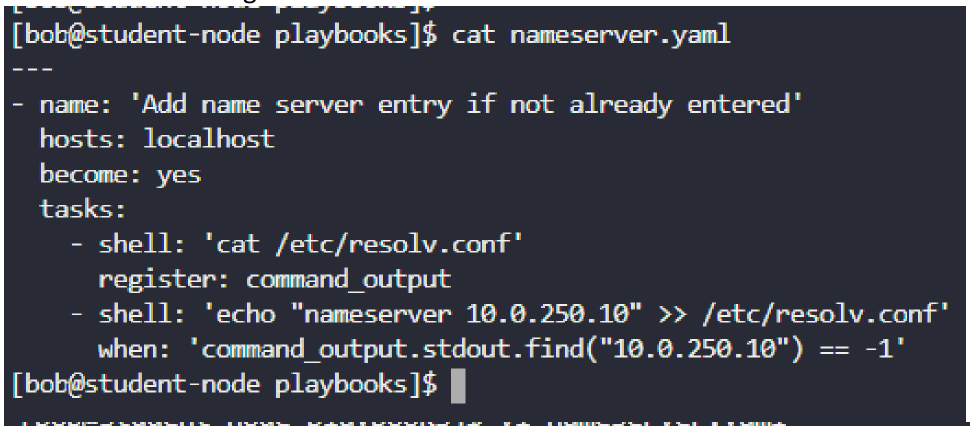
```
[bob@student-node playbooks]$ cat age.yaml
---
- name: 'Am I an Adult or a Child?'
  hosts: localhost
  vars:
    age: 25
  tasks:
    - name: I am a Child
      command: 'echo "I am a Child"'
      when: 'age < 18'
    - name: I am an Adult
      command: 'echo "I am an Adult"'
      when: 'age >=18'
```



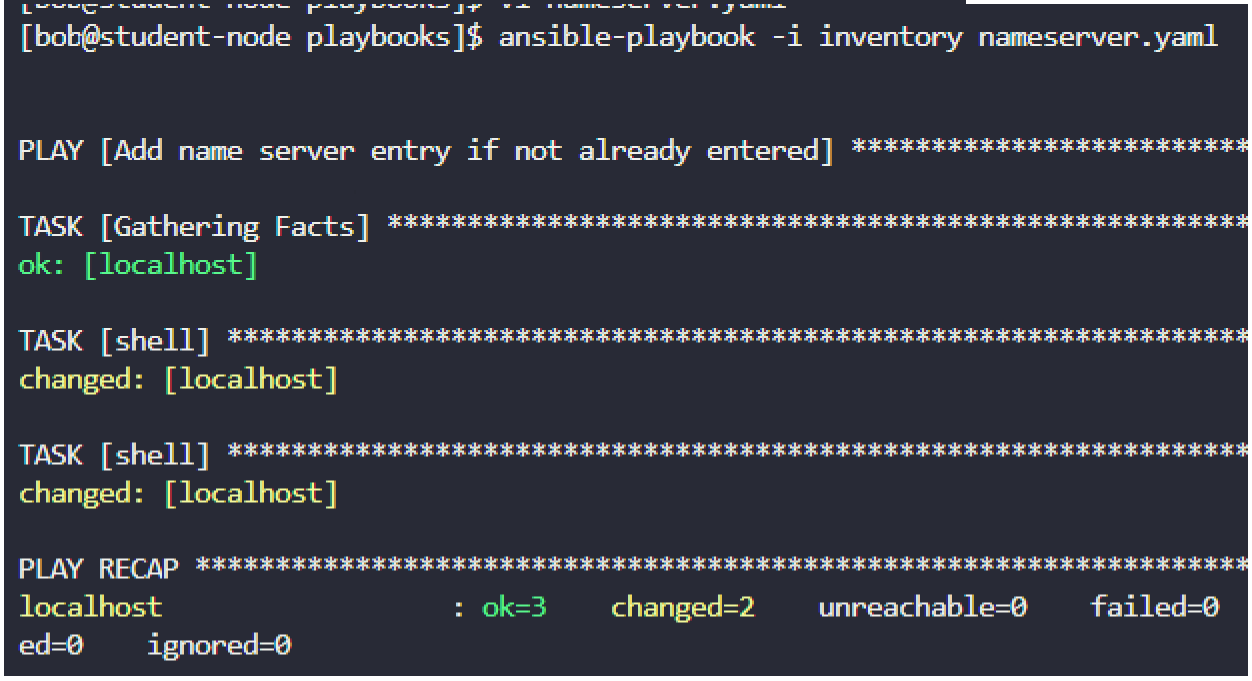
```
[bob@student-node playbooks]$ ansible-playbook -i inventory age.yaml

PLAY [Am I an Adult or a Child?] ***************************************

TASK [Gathering Facts] *************************************************
ok: [localhost]

TASK [I am a Child] ****************************************************
skipping: [localhost]

TASK [I am an Adult] ***************************************************
changed: [localhost]

PLAY RECAP *************************************************************
localhost                  : ok=2    changed=1    unreachable=0    faile
ed=0    ignored=0

[bob@student-node playbooks]$
```

Playbook /home/bob/playbooks/nameserver.yaml attempts to add an entry in /etc/resolv.conf file to add a new nameserver.
The first task in the playbook is using the shell module to display the existing contents of /etc/resolv.conf file and the second one is adding a new line containing the name server details into the file. However, when this playbook is run multiple times, it keeps adding new entries of same line into the resolv.conf file. To resolve this issue, update the playbook as per details mentioned below.
Add a register directive to store the output of the first task to a variable called command_output
Then add a conditional to the second task to check if the output already contains the name server (10.0.250.10). Use command_output.stdout.find(<IP>) == -1

Note:
a. A better way to do this would be to use the lineinfile module. This is just for practice.
b.shell and command modules are similar in a way that they are used to execute a command on the system. However, shell executes the command inside a shell giving us access to environment variables and redirection using >>.



```
[bob@student-node playbooks]$ cat nameserver.yaml
---
- name: 'Add name server entry if not already entered'
  hosts: localhost
  become: yes
  tasks:
    - shell: 'cat /etc/resolv.conf'
      register: command_output
    - shell: 'echo "nameserver 10.0.250.10" >> /etc/resolv.conf'
      when: 'command_output.stdout.find("10.0.250.10") == -1'
[bob@student-node playbooks]$
```



```
[bob@student-node playbooks]$ ansible-playbook -i inventory nameserver.yaml


PLAY [Add name server entry if not already entered] *************************

TASK [Gathering Facts] ******************************************************
ok: [localhost]

TASK [shell] ****************************************************************
changed: [localhost]

TASK [shell] ****************************************************************
changed: [localhost]

PLAY RECAP ******************************************************************
localhost                  : ok=3    changed=2    unreachable=0    failed=0
ed=0    ignored=0
```