

Lab2:

Q1:tags

```
Dockerfile  Inventory  playbook.yml  index.html  ansible.cfg  Welcome
playbook.yml
1 - name: Lab2
2
3 tasks:
4   - name: update_cache
5     apt:
6       update_cache: true
7       tags: update
8   - name: instsall Nginx
9     apt:
10      name: nginx
11      state: latest
12      tags: instsall
13   - name: Restart nginx service
14     service:
15       name: nginx
16       state: restarted
17       use: service
18   tags: always
19
20
PROBLEMS  OUTPUT  TERMINAL  PORTS  COMMENTS
>  TERMINAL
[menna@localhost ansible_lab1]$ ansible-playbook playbook.yml --tags update
BECOME password:

PLAY [Lab2] *****

TASK [Gathering Facts] *****
ok: [172.17.0.2]

TASK [update_cache] *****
ok: [172.17.0.2]

TASK [Restart nginx service] *****
changed: [172.17.0.2]

PLAY RECAP *****
172.17.0.2 : ok=3  changed=1  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

Another time using instal tag:

```
[menna@localhost ansible_lab1]$ ansible-playbook playbook.yml --tags instal
BECOME password:

PLAY [Lab2] *****

TASK [Gathering Facts] *****
ok: [172.17.0.2]

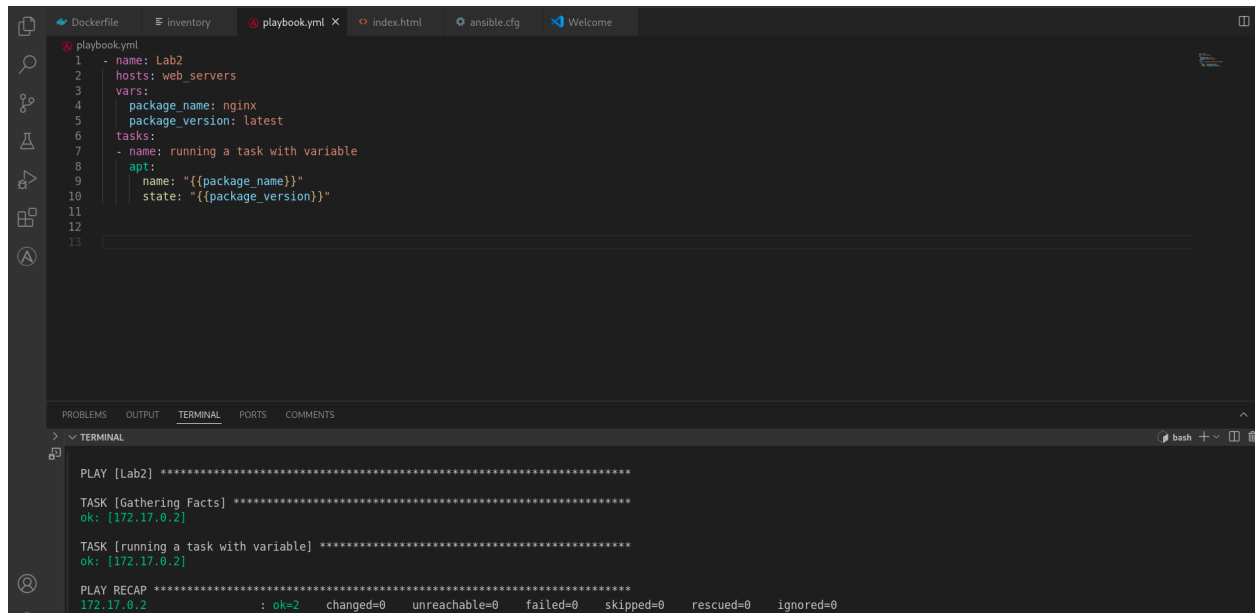
TASK [instsall Nginx] *****
ok: [172.17.0.2]

TASK [Restart nginx service] *****
changed: [172.17.0.2]

PLAY RECAP *****
172.17.0.2 : ok=3  changed=1  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

Q2. variables

using playbook



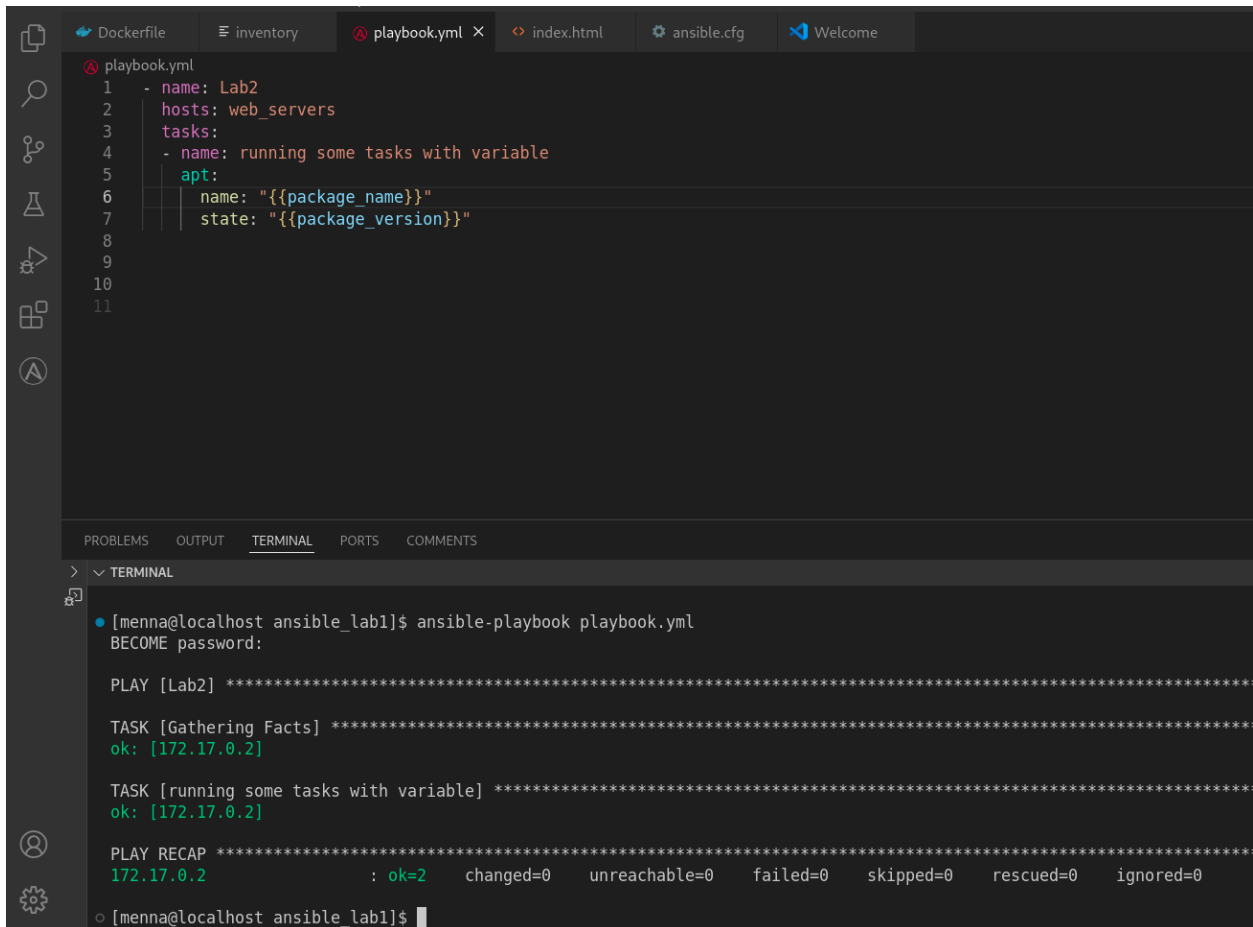
The screenshot shows an IDE with a dark theme. The top panel displays a file explorer with tabs for 'playbook.yml', 'index.html', 'ansible.cfg', and 'Welcome'. The 'playbook.yml' file is open, showing the following content:

```
1 - name: Lab2
2   hosts: web_servers
3   vars:
4     package_name: nginx
5     package_version: latest
6   tasks:
7     - name: running a task with variable
8       apt:
9         name: "{{package_name}}"
10        state: "{{package_version}}"
11
12
13
```

The bottom panel shows the 'TERMINAL' output, which displays the execution results of the playbook:

```
PLAY [Lab2] *****
TASK [Gathering Facts] *****
ok: [172.17.0.2]
TASK [running a task with variable] *****
ok: [172.17.0.2]
PLAY RECAP *****
172.17.0.2 : ok=2  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

Using inventory level



The image shows a VS Code editor with a file explorer on the left containing icons for Explorer, Search, Source Control, Run and Debug, Dockerfile, Inventory, Playbook, Index, Ansible Config, and Welcome. The main editor displays a file named `playbook.yml` with the following content:

```
1 - name: Lab2
2   hosts: web_servers
3   tasks:
4     - name: running some tasks with variable
5       apt:
6         name: "{{package_name}}"
7         state: "{{package_version}}"
8
9
10
11
```

Below the editor is a panel with tabs for PROBLEMS, OUTPUT, TERMINAL, PORTS, and COMMENTS. The TERMINAL tab is active, showing the output of the command `ansible-playbook playbook.yml` executed in a terminal window. The output shows the playbook running on host `172.17.0.2` with the following details:

```
[menna@localhost ansible_lab1]$ ansible-playbook playbook.yml
BECOME password:

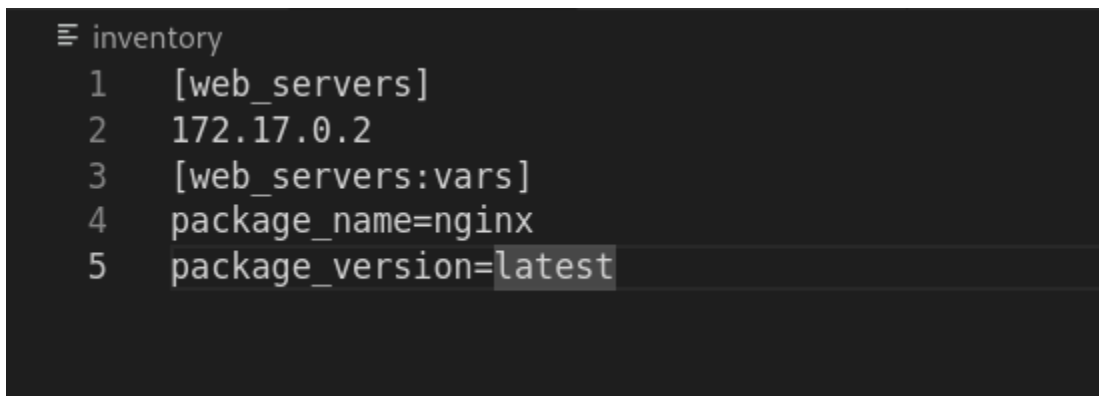
PLAY [Lab2] *****

TASK [Gathering Facts] *****
ok: [172.17.0.2]

TASK [running some tasks with variable] *****
ok: [172.17.0.2]

PLAY RECAP *****
172.17.0.2      : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[menna@localhost ansible_lab1]$
```



The image shows the content of the `inventory` file, which is displayed in a code editor. The content is as follows:

```
inventory
1  [web_servers]
2  172.17.0.2
3  [web_servers:vars]
4  package_name=nginx
5  package_version=latest
```

Using command line level

```
playbook.yml
1  - name: Lab2
2    hosts: web_servers
3    tasks:
4      - name: running some tasks with variable
5        apt:
6          name: "{{package_name}}"
7          state: "{{package_version}}"
8
9
10
11 |

PROBLEMS OUTPUT TERMINAL PORTS COMMENTS
> TERMINAL
• [menna@localhost ansible_lab1]$ ansible-playbook playbook.yml -e "package_name=nginx package_version=latest"
BECOME password:

PLAY [Lab2] *****

TASK [Gathering Facts] *****
ok: [172.17.0.2]

TASK [running some tasks with variable] *****
ok: [172.17.0.2]

PLAY RECAP *****
172.17.0.2 : ok=2 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

[menna@localhost ansible_lab1]$
```

Q3.Loops:

Part1:

```
playbook.yml
1  - name: Lab2
2    hosts: web_servers
3    tasks:
4      - name: running some tasks with variable
5        apt:
6          name: "{{item}}"
7          state: latest
8        loop:
9          - nginx
10         - mariadb-server
11         - curl
12
13
14
15

PROBLEMS OUTPUT TERMINAL PORTS COMMENTS
> TERMINAL
• [menna@localhost ansible_lab1]$ ansible-playbook playbook.yml
BECOME password:

PLAY [Lab2] *****

TASK [Gathering Facts] *****
ok: [172.17.0.2]

TASK [running some tasks with variable] *****
ok: [172.17.0.2] => (item=nginx)
changed: [172.17.0.2] => (item=mariadb-server)
changed: [172.17.0.2] => (item=curl)

PLAY RECAP *****
172.17.0.2 : ok=2 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

[menna@localhost ansible_lab1]$
```

Part2:

```
playbook.yml
1 - name: Lab2
2   hosts: web_servers
3   tasks:
4     - name: running some tasks with loops
5       apt:
6         name: "{{item.package_name}}"
7         state: "{{item.package_state}}"
8       loop:
9         - { package_name: "nginx", package_state: "latest"}
10        - { package_name: "mariadb-server", package_state: "present"}
11        - { package_name: "curl", package_state: "absent"}
12
13
14
15
```

PROBLEMS OUTPUT **TERMINAL** PORTS COMMENTS

> TERMINAL

```
[menna@localhost ansible_lab1]$ ansible-playbook playbook.yml
BECOME password:

PLAY [Lab2] *****

TASK [Gathering Facts] *****
ok: [172.17.0.2]

TASK [running some tasks with loops] *****
ok: [172.17.0.2] => (item={'package_name': 'nginx', 'package_state': 'latest'})
ok: [172.17.0.2] => (item={'package_name': 'mariadb-server', 'package_state': 'present'})
changed: [172.17.0.2] => (item={'package_name': 'curl', 'package_state': 'absent'})

PLAY RECAP *****
172.17.0.2 : ok=2 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

Q4:When:

I pulled a centos docker image and entered the container then ran some commands to solve the yml issue which are:

```
sudo sed -i 's/mirrorlist/#mirrorlist/g' /etc/yum.repos.d/CentOS-*
```

```
sudo sed -i 's|#baseurl=http://mirror.centos.org|baseurl=http://vault.centos.org|g' /etc/yum.repos.d/CentOS-*
```

Then i made yum update

Then i installed the openssh-server in my centos container and i made a user named:centosManouna with password :1234

```
[menna@localhost ansible_lab1]$ sudo docker inspect d6c7247255dd
[
  {
```

```
    "Gateway": "172.17.0.1",
    "IPAddress": "172.17.0.2",
    "IPPrefixLen": 16,
```

And i used the already running docker container of ubuntu:

```
[menna@localhost ansible_lab1]$ sudo docker container run -itd target
99de0eba91acef3dd3c9132cf6102d6860546b977476a41400739f861d66498c

[menna@localhost ansible_lab1]$ sudo docker inspect 99de0eba91acef3dd3c9132cf6102d6860546b977476a41400739f861d66498c
[
  {
    "Gateway": "172.17.0.1",
    "IPAddress": "172.17.0.3",
    "IPPrefixLen": 16
  }
]
```

Then i used the ssh-copy-id for both containers to access them using ssh only without the password

```
[menna@localhost ansible_lab1]$ ssh centosManouna@172.17.0.2
Last login: Mon Feb  3 15:18:15 2025 from gateway
[centosManouna@d6c7247255dd ~]$ exit
```

```
[menna@localhost ansible_lab1]$ ssh manouna@172.17.0.3
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 5.14.0-554.el9.x86_64 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Mon Feb  3 15:29:42 2025 from 172.17.0.1
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

manouna@99de0eba91ac:~$ exit
```

```
1  [web_servers]
2  172.17.0.2
3  172.17.0.3
```

Playbook:

There is an unknown issue considering the centos container that i don't know how to solve:

```
[menna@localhost ansible_lab1]$ ansible-playbook playbook.yml
BECOME password:

PLAY [playbook with conditions] *****

TASK [Gathering Facts] *****
ok: [172.17.0.3]
```

```

playbook.yml
1  - name: playbook with conditions
2    hosts: all
3    tasks:
4      - name: ubuntu task with conditions
5        apt:
6          name: nginx
7          state: latest
8          when: ansible_facts['distribution'] == "Ubuntu"
9          remote_user: "{{ ansible_ssh_user_ubuntu }}"
10
11     - name: centos task with conditions
12       yum:
13         name: httpd
14         state: latest
15         when: ansible_facts['distribution'] == "CentOS"
16         remote_user: "{{ ansible_ssh_user_centos }}"
17

```

```

inventory
1  [centos]
2  172.17.0.2 ansible_ssh_user=centosManouna
3
4  [ubuntu]
5  172.17.0.3 ansible_ssh_user=manouna

```

```

ansible.cfg
1  [defaults]
2  inventory = ./inventory
3  private_key = /home/menna/.ssh/id_rsaManouna
4
5  [privilege_escalation]
6  become = true
7  become_ask_pass = true
8
9  [vars]
10 ansible_ssh_user_ubuntu = manouna
11 ansible_ssh_user_centos = centosManouna
12

```

Registers:

```
① playbook.yml
1  - name: playbook with conditions
2    hosts: webservers
3    tasks:
4      - name: ubuntu task with conditions
5        apt:
6          name: nginx
7          state: latest
8          update_cache: yes
9          register: install_result
10
11     - name: result
12       debug:
13         var: install_result
14
TASK [result] *****
ok: [172.17.0.3] => {
  "install_result": {
    "cache_update_time": 1738661517,
    "cache_updated": false,
    "changed": false,
    "failed": false
  }
}
PLAY RECAP *****
172.17.0.3      : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0
```

Question2:

```
① playbook.yml
1  - name: playbook with conditions
2    hosts: webservers
3    tasks:
4      - name: ubuntu task with conditions
5        apt:
6          name: nginx
7          state: latest
8          update_cache: yes
9          register: install_result
10
11     - name: result
12       debug:
13         var: install_result
14     - name: restart service
15       service:
16         name: nginx
17         state: restarted
18         use: service
19       when: install_result.changed or not install_result.failed
20
```



```
TASK [Gathering Facts] *****
ok: [172.17.0.3]

TASK [ubuntu task with conditions] *****
ok: [172.17.0.3]

TASK [result] *****
ok: [172.17.0.3] => {
  "install_result": {
    "cache_update_time": 1738661517,
    "cache_updated": false,
    "changed": false,
    "failed": false
  }
}

TASK [restart service] *****
changed: [172.17.0.3]

PLAY RECAP *****
172.17.0.3          : ok=4    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```