

1:

workstation:

```
hostnamectl set-hostname workstation
```

```
bash
```

```
ip a
```

```
nmcli con mod ens160 ipv4.addresses 192.1.48.100/24 ipv4.gateway 192.1.48.2 ipv4.dns  
192.1.48.2 ipv4.method manual autoconnect yes
```

```
nmcli con up ens160
```

```
ip a
```

agent1:

```
hostnamectl set-hostname agent1
```

```
bash
```

```
ip a
```

```
nmcli con mod ens160 ipv4.addresses 192.1.48.101/24 ipv4.gateway 192.1.48.2 ipv4.dns  
192.1.48.2 ipv4.method manual autoconnect yes
```

```
nmcli con up ens160
```

```
ip a
```

agent2:

```
hostnamectl set-hostname agent2
```

```
bash
```

```
ip a
```

```
nmcli con mod ens160 ipv4.addresses 192.1.48.102/24 ipv4.gateway 192.1.48.2 ipv4.dns  
192.1.48.2 ipv4.method manual autoconnect yes
```

```
nmcli con up ens160
```

```
ip a
```

2:

workstation:

```
mkdir /ansible
```

```
cd /ansible
```

```
vim hosts
```

```
[dbservers]
```

```
agen1
```

```
[webservers]
```

```
agent1
```

```
agent2
```

```
vim ansible.cfg
```

```
[defaults]
```

```
inventory = /ansible/hosts
```

```
roles_path = /home/greg/ansible/roles
```

```
vim /etc/hosts
```

```
192.148.100 workstation
```

```
192.148.101 agent1
```

```
192.148.102 agent2
```

```
ansible -m ping all
```

```
ssh -keygen
```

```
ssh-copy-id root@agent1
```

```
ssh-copy-id root@agent2
```

```
ssh root@agent1
```

```
exit
```

```
ssh root@agent2
```

```
exit
```

```
3:
```

```
ansible}# mkdir playbooks
```

```
vim playbooks/users.yml
```

```
- hosts: agent1
```

```
  tasks:
```

```
    - group:
```

```
      name: manager
```

```
      state: present
```

```
    - user:
```

```
      name: ma{{item}}
```

```
      home: /home/ma{{item}}
```

```
      password: 123456
```

```
      loop: "{{ range(1,16) | list }}"
```

```
- hosts: agent2
```

```
  tasks:
```

```
    - group:
```

```
      name: consumer
```

```
      state: present
```

```
    - user:
```

```
      name: con{{item}}
```

```
      home: /home/con{{item}}
```

```
        password: 123456
        loop: "{{ range(1,13,2) | list }}"
- hosts: agent1
  tasks:
  - user:
      name: dbtest
      state: absent
      remove: yes
```

ansible-playbook playbooks/users.yml
ansible all -a 'tail -17 /etc/passwd' 检查

4:
vim /root/hwreport.txt
HardWare details:
Memory: xxMB 以 MB 表示的总内存大小
BIOS: xx 版本
Disk: sda1 xxMB, sda2 xxMB.

```
vim /ansible/playbooks/hwreport.yml
- hosts: all
  vars:
  - 1:
    - 3: Memory
      4: "{{ansible_memtotal_mb}}MB"
    - 3: BIOS
      4: "{{ansible_bios_version}}"
  - 2:
    - 3: sda1
      4: "{{ansible_devices.sda.partitions.sda1.size}}"
    - 3: sda2
      4: "{{ansible_devices.sda.partitons.sda2.size}}"
  tasks:
  - copy:
      src: /root/hwreport.txt
      dest: /root/hwreport.txt
  - replace:
      path: /root/hwreport.txt
      regexp: "{{item.3}}:.*"
      replace: "{{item.3}}:  {{item.4}}"
      loop: "{{1}}"
  - replace:
      path: /root/hwreport.txt
      regexp: "{{item.3}} xxMB"
```

```
    replace: "{{item.3}} {{item.4}}"
  loop: "{{2}}"
```

```
ansible-playbook playbooks/hwreport.yml
ansible all -a 'cat /root/hwreport.txt'
```

5:

```
vim playbooks/cron.yml
```

```
- hosts: dbrowsers
  tasks:
    - cron:
        name: "shell crond"
        minute: "*/2"
        job: date +%Y-%m-%d %H:%M "ansible in progress"
    - service:
        name: crond
        state: restarted
```

```
ansible-playbook playbook/cron.yml
```

6:

```
- hosts: webrowsers
  tasks:
    - service:
        name: firewalld
        state: stopped
        enabled: no
    - shell: 'setenforce 0'
    - replace:
        path: /etc/selinux/config
        regexp: 'SELINUX=enforcing'
        replace: 'SELINUX=disabled'
```

```
ansible-playbook playbooks/safty.yml
```

7:

```
echo 'test 7' > /opt/file.txt
vim playbooks/copyfile.yml
- hosts: webrowsers
  gather_facts: no
  tasks:
    - copy:
        src: /opt/file.txt
        dest: /var/file.txt
```

force: no

```
ansible-playbook playbooks/copyfile.yml
ansible all -a 'cat /var/file.txt'
```

8:

```
vim playbooks/install.yml
```

```
- hosts: webservers
  gather_facts: no
  tasks:
    - shell: 'cat /data/db.info'
      ignore_errors: yes
      register: result
    - yum:
        name: mariadb
        state: present
        when: "'mariadb' in result.stdout"
```

```
ansible-playbook playbooks/install.yml
```

9:

```
vim httpd.j2
```

Welcome, This is wxt's homepage!, The host is {{ansible_hostname}}

```
vim playbooks/httpd.yml
```

```
- hosts: webservers
  tasks:
    - yum:
        name: httpd
        state: present
    - service:
        name: httpd
        state: started
        enabled: yes
    - template:
        src: /ansible/httpd.j2
        dest: /var/www/html/index.html
    - replace:
        path: /etc/httpd/conf/httpd.conf
        regexp: 'AllowOverride None'
        replace: 'AllowOverride AuthConfig'
```

```
ansible-playbook playbooks/httpd.yml
```

```
curl 192.1.48.101
```

```
curl 192.1.48.102
```

10: 与 6 一样

11:

```
vim hosts.j2
{{ansible_hostname}}    {{ansible_processor_vcpus}}    {{ansible_facts.default_ipv4.address}}
{{ansible_default_ipv4.macaddress}}
mkdir data
vim playbooks/hosts.yml
- hosts: all
  tasks:
    - template:
        src: /ansible/hosts.j2
        dest: /var/www/html/ansible_details.html
    - fetch:
        src: /var/www/html/ansible_details.html
        dest: /ansible/data/{{ansible_hostname}}
        flat: yes
```

```
ansible-playbook playbooks/hosts.yml
cat data/agent1
cat data/agent2
```

12:

```
echo 'this is my test' > /var/tmp/ans.txt
vim playbooks/ping.yml
- hosts: agent1
  tasks:
    - copy:
        src: /var/tmp/ans.txt
        dest: /var/www/html/tag1/html
- hosts:
  tasks:
    - copy:
        src: /var/tmp/ans.txt
        dest: /var/www/html/tag2.html
    tags: backup
```

```
ansible-playbook playbooks/ping.yml --tags "production"
ansible-playbook playbooks/ping.yml --tags "backup"
ansible agent2 -a 'cat /var/www/html/tag2.html'
```

13:

```
vim 13node.j2
```

my node is also {{ansible_hostname}}

vim playbooks/saveabort.yml

```
- hosts: all
  tasks:
    - stat:
        path: /data/ansible_abort.txt
        register: result
    - template:
        src: /ansible/13node.j2
        dest: /data/ansible_abort.txt
        when: result.stat.exists == false
```

ansible-playbook playbooks/saveabort.yml

ansible all -a 'cat /data/ansible_abort.txt'

ansible-playbook playbooks/saveabort.yml

14:

vim playbooks/mariadb.yml

```
- hosts: agent1
  tasks:
    - block:
        - yum:
            name: mariadb,mariadb-server
            state: present
            register: result1
        rescue:
        - debug:
            msg: "{{result1}}"
        - debug:
            msg: "Installation failed"
    - block:
        - service:
            name: mariadb
            state: started
            enabled: yes
            register: result2
        rescue:
        - debug:
            msg: "{{result2}}"
        - debug:
            msg: "Starting failed"
```

ansible all -a 'mount /dev/cdrom /mnt'

```
ansible-playbook playbooks/mariadb.yml
ansible agent1 -a 'systemctl status mariadb'
```

15:

```
ansible-galaxy init /home/greg/ansible/roles/apache
ls /home/greg/ansible/roles/apache
cd /home/greg/ansible/roles
vim apache/templates/index.html.j2
Welcome to {{ansible_facts.fqdn}} on {{ansible_facts.default_ipv4.address}}
```

```
vim apache/tasks/main.yml
- yum:
    name: httpd
    state: present
- service:
    name: httpd
    state: started
    enabled: yes
- template:
    src: index.html.j2
    dest: /var/www/html/index.html
```

```
vim /home/greg/ansible/apache.yml
- hosts: all
  roles:
    - apache
```

```
cd /ansible
ansible-playbook /home/greg/ansible/apache.yml
curl agent1
curl agent2
```