**Create a playbook httpd.yml under ~/playbooks/ to install httpd package on web1 node using Ansible's yum module.**

```
[thor@ansible-controller playbooks]$ cat httpd.yml
---
- name: 'Install httpd package'
  hosts: web1
  tasks:
  - name: 'Install httpd package'
    yum:
      name: httpd
      state: installed
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory httpd.yml
```

---

**We have an rpm available for wget package on URL http://mirror.centos.org/centos/7/os/x86_64/Packages/wget-1.14-18.el7_6.1.x86_64.rpm. Create a playbook with name wget.yml under ~/playbooks to install that rpm on web1 node using yum module.**

```
[thor@ansible-controller playbooks]$ cat wget.yml
---
- name: 'Install rpm using url'
  hosts: all
  tasks:
  - name: 'Install rpm'
    yum:
      name: http://mirror.centos.org/centos/7/os/x86_64/Packages/wget-1.14-18.el7_6.1.x86_64.rpm
      state: installed
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory wget.yml
```

---

**There is a playbook under ~/playbooks named as unzip.yml to install unzip package on web1 node. We want to install unzip-5.52 version of this package so before running the playbook make the required changes.**

```
[thor@ansible-controller playbooks]$ cat unzip.yml
---
- hosts: all
  tasks:
    - name: Install unzip package
      yum:
        name: unzip-5.52
        state: present
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory unzip.yml
```

---

**Our playbook - iotop.yml - to install the latest version of iotop package keeps failing. Please fix the issue so that playbook can work.**
**The playbook is located under ~/playbooks directory. And the inventory file is inventory**

```
[thor@ansible-controller playbooks]$ cat iotop.yml
---
- hosts: all
  tasks:
    - name: Install iotop package
      yum:
        name: iotop
        state: latest
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory iotop.yml
```

---

**We want to install some more packages on web1 node. Create a playbook ~/playbooks/multi-pkgs.yml to install the latest version of sudo package, moreover we already have vsftpd v3.0.2 installed but due to some compatibility issues we want to install vsftpd v2.2.2 so add a task in same playbook to do so.**

```
[thor@ansible-controller playbooks]$ cat multi-pkgs.yml
---
- name: 'Install'
  hosts: web1
  tasks:
  - name: 'Install'
    yum:
      name: sudo
      state: latest
  - name: 'Install'
    yum:
      name: vsftpd-2.2.2
      state: present
      allow_downgrade: yes
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory multi-pkgs.yml
```

---

**We would like the httpd service on web1 node to always start automatically after the system reboots. Update the httpd.yml playbook you created earlier with the required changes.**

```
[thor@ansible-controller playbooks]$ cat httpd.yml
---
- name: 'Service status'
  hosts: all
  tasks:
  - name: 'Service status'
    service:
      name: httpd
      state: started
      enabled: true
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory httpd.yml
```

**Create a playbook httpd.yml under ~/playbooks directory to make sure httpd service is started on web1 node. You can use ~/playbooks/inventory**

```
[thor@ansible-controller playbooks]$ cat httpd.yml
---
- name: 'Service status'
  hosts: all
  tasks:
  - name: 'Service status'
    service:
      name: httpd
      state: started
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory httpd.yml
```

---

**We have a playbook ~/playbooks/file.yml to copy a file with a welcome message under httpd server's document root on web1 node. Make changes in the playbook so that httpd server reloads after copying the file, make sure it does not restart the httpd server.**

```
[thor@ansible-controller playbooks]$ cat file.yml
---
- hosts: all
  gather_facts: no
  tasks:
    - name: Copy Apache welcome file
      copy:
        src: index.html
        dest: /var/www/html/index.html
    - name: Reload server
      service:
        name: httpd
        state: reloaded
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory file.yml
```

---

**We would like the httpd service on web1 node to always start automatically after the system reboots. Update the httpd.yml playbook you created earlier with the required changes.**

```
[thor@ansible-controller playbooks]$ cat httpd.yml
---
- name: 'Service status'
  hosts: all
  tasks:
  - name: 'Service status'
    service:
      name: httpd
      state: started
      enabled: true
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory httpd.yml
```

---

**We created a playbook ~/playbooks/config.yml to enable port 443 for httpd on web1 node as we want to run nginx on the default port 80 so port 80 needs to be free. Make changes in the playbook so that httpd service restarts after making these change.**

```
[thor@ansible-controller playbooks]$ cat config.yml
---
- hosts: all
  gather_facts: no
  tasks:
    - name: Make changes in Apache config
      replace:
        path: /etc/httpd/conf/httpd.conf
        regexp: "^Listen 80"
        replace: "Listen 443"
    - name: Restart httpd
      service:
        name: httpd
        state: restarted
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory config.yml
```

---

**Create a playbook ~/playbooks/nginx.yml to install nginx on web1 node and make sure nginx service is started and should always start even after the system reboots.**

```
[thor@ansible-controller playbooks]$ cat nginx.yml
---
- name: Nginx service
  hosts: all
  tasks:
  - name: Nginx
    yum:
      name: nginx
      state: installed
  - name: Nginx
    service:
      name: nginx
      state: started
      enabled: true
[thor@ansible-controller playbooks]$ ansible-playbook -i inventory nginx.yml
```

---