

## Ansible

### Lab 3

Name: Muhammad Osama

#### Task1

- Loop over a list of package and install latest versions

```
A loops.ansible.yaml
1  - name: installing packages
2    gather_facts: false
3    hosts: web-servers
4    tasks:
5      - name: install packages using loop
6        apt:
7          name: "{{ item }}"
8          state: latest
9        loop:
10         - nginx
11         - curl
12         - sudo
13
```

- Loop over a list of packages and use different states

```
A loops.ansible.yaml
1  - name: installing packages
2    gather_facts: false
3    hosts: web-servers
4    vars:
5      packages:
6        - { package_name: "nginx", package_state: "latest" }
7        - { package_name: "curl", package_state: "present" }
8        - { package_name: "vim", package_state: "absent" }
9
10   tasks:
11     - name: install packages using loop
12       apt:
13         name: "{{ item.package_name }}"
14         state: "{{ item.package_state }}"
15       loop: "{{ packages }}"
16
17
```

## Task2

- Install nginx or httpd depending on distribution
- Restart nginx service if distribution is ubuntu.
- Restart httpd service if distribution is centos.


```
1 - name: installing packages
2   gather_facts: true
3   hosts: web-servers
4   vars:
5     tasks:
6       - name: install nginx when distribution is ubuntu
7         apt:
8           name: nginx
9           state: latest
10        when: ansible_facts['distribution'] == "Ubuntu"
11       - name: install httpd when distribution is CentOS
12         apt:
13           name: httpd
14           state: latest
15        when: ansible_facts['distribution'] == "CentOS"
16       - name: restart nginx when distribution is ubuntu
17         sysvinit:
18           name: nginx
19           state: restarted
20        when: ansible_facts['distribution'] == "Ubuntu"
21       - name: restart httpd when distribution is CentOS
22         sysvinit:
23           name: httpd
24           state: restarted
25        when: ansible_facts['distribution'] == "CentOS"
```

### Task3

- View the value of your register variable using debug module
- Restart service if the installation task was changed or was not failed

```
1 ---
2 - name: Installing and Restarting Web Servers
3   hosts: web-servers
4   gather_facts: true
5
6   tasks:
7     - name: Install Nginx on Ubuntu or httpd on CentOS
8       apt:
9         name: "{{ 'nginx' if ansible_facts['distribution'] == 'Ubuntu' else 'httpd' }}"
10        state: latest
11        register: result
12
13     - name: Restart Nginx on Ubuntu
14       sysvinit:
15         name: nginx
16         state: restarted
17         when: ansible_facts['distribution'] == "Ubuntu"
18
19     - name: Restart httpd on CentOS
20       sysvinit:
21         name: httpd
22         state: restarted
23         when: ansible_facts['distribution'] == "CentOS"
24
25
26     - name: Print return information from the installation task
27       debug:
28         msg: "{{ result }}"
29
30
31     - name: Restart Service if the installation task was changed
32       sysvinit:
33         name: "{{ 'nginx' if ansible_facts['distribution'] == 'Ubuntu' else 'httpd' }}"
34         state: restarted
35         when: result.changed == true
36
37
```

## Task4



```
1  - name: install package
2    gather_facts: true
3    hosts: web-servers
4    vars:
5      package_name: nginx
6      package_state: latest
7      file_names:
8        - ./index.html
9        - ./file.css
10       - ./file.js
11
12    tasks:
13      - name: install package
14        apt:
15          name: "{{ package_name }}"
16          state: "{{ package_state }}"
17          notify: copy_files_handler
18          changed_when: true
19
20
21    handlers:
22      - name: copy_files_handler
23        copy:
24          src: "{{ item }}"
25          dest: /var/www/html/
26          loop: "{{ file_names }}"
27          notify: restart_service
28
29      - name: restart_service
30        sysvinit:
31          name: "{{ package_name }}"
32          state: restarted
33
```

## Task5

```
1  - name: install package
2    gather_facts: true
3    hosts: web-servers
4    vars:
5      package_name: nginx
6      package_state: latest
7      file_names:
8        - ./file.css
9        - ./file.js
10     message: "DevOps"
11
12   tasks:
13     - name: install package
14       apt:
15         name: "{{ package_name }}"
16         state: "{{ package_state }}"
17         notify: copy_files_handler
18         changed_when: true
19
20     - name: flush
21       meta: flush_handlers
22
23     - name: html_template
24       template:
25         src: ./index.html.j2
26         dest: /var/www/html/index.html
27         notify: restart_service
28
29
30   handlers:
31     - name: copy_files_handler
32       copy:
33         src: "{{ item }}"
34         dest: /var/www/html/
35         loop: "{{ file_names }}"
36
37     - name: restart_service
38       sysvinit:
39         name: "{{ package_name }}"
40         state: restarted
41
```