Loop task

 Loop over a list of packages and install latest versions.

 Loop over a list of packages and use different states as per input.

```
root@ip-10-2-0-220:~/lab1# ansible-playbook task2.yaml
BECOME password:

PLAY [looptask] ****************************************************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************************************************
ok: [10.2.0.67]

TASK [Install latest packages on ubuntu] ***************************************************************************************************************************
ok: [10.2.0.67] => (item=nginx)
changed: [10.2.0.67] => (item=docker)
changed: [10.2.0.67] => (item=curl)
ok: [10.2.0.67] => (item=sudo)
changed: [10.2.0.67] => (item=mysql-server)

TASK [Install packages on ubuntu with different states] ************************************************************************************************************
changed: [10.2.0.67] => (item={'package_name': 'docker', 'package_state': 'absent'})
changed: [10.2.0.67] => (item={'package_name': 'mysql-server', 'package_state': 'absent'})
ok: [10.2.0.67] => (item={'package_name': 'nginx', 'package_state': 'latest'})
changed: [10.2.0.67] => (item={'package_name': 'curl', 'package_state': 'absent'})
ok: [10.2.0.67] => (item={'package_name': 'sudo', 'package_state': 'present'})

PLAY RECAP *********************************************************************************************************************************************************
10.2.0.67                  : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

root@ip-10-2-0-220:~/lab1#
```

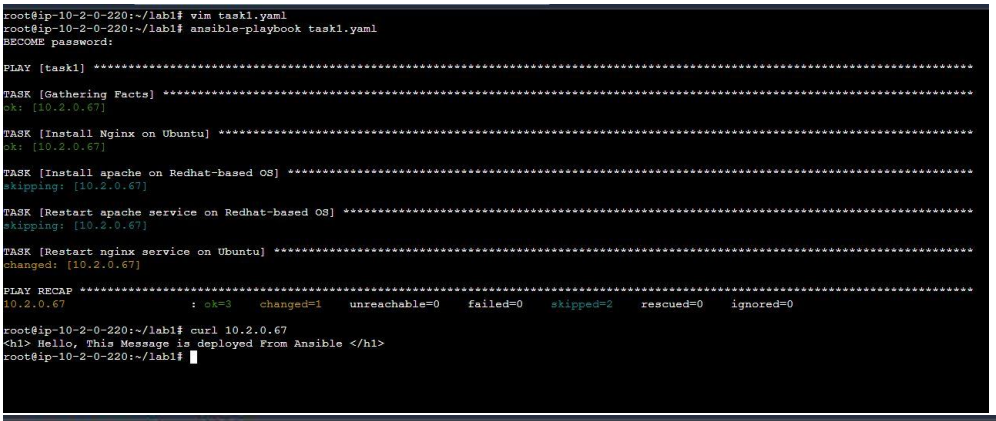
```yaml
- name: looptask
  hosts: servers
  ignore_errors: true
  tasks:
    - name: Install latest packages on ubuntu
      ansible.builtin.apt:
        name: "{{ item }}"
        state: "latest"
        update_cache: true
      when: ansible_facts.os_family == "Debian"
      loop:
        - nginx
        - docker
        - curl
        - sudo
        - mysql-server
    - name: Install packages on ubuntu with different states
      ansible.builtin.apt:
        name: "{{ item.package_name }}"
        state: "{{ item.package_state }}"
        update_cache: true
      when: ansible_facts.os_family == "Debian"
      loop:
        - {package_name: "docker", package_state: "absent"}
        - {package_name: "mysql-server", package_state: "absent"}
        - { package_name: "nginx", package_state: "latest" }
        - { package_name: "curl", package_state: "absent" }
        - { package_name: "sudo", package_state: "present" }
```

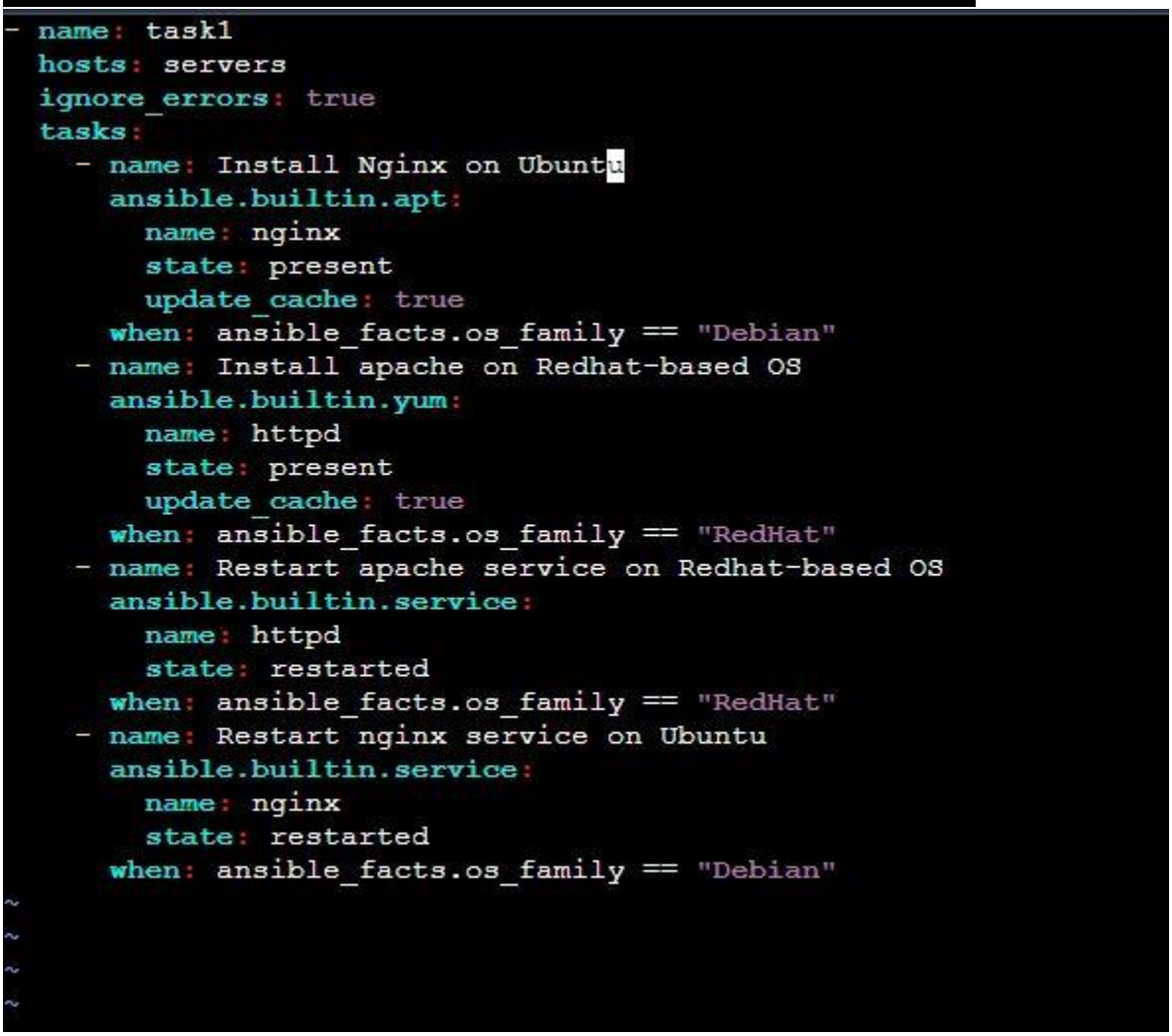
When task

 Install nginx or httpd depending on distribution

 Restart nginx service if distribution is ubuntu.

 Restart httpd service if distribution is centos





```
[defaults]
inventory = ./inventory
private_key_file = /root/ansible.pem
remote_user = ubuntu

[privilege_escalation]
become_ask_pass = true
become = true
```

```
[servers]
10.2.0.67
```

REGISTER & WHEN task

 View the value of your register variable using debug module

 Restart service if the installation task was changed or was not failed

```
- name: Install and restart nginx
  hosts: servers
  ignore_errors: true
  tasks:
    - name: Install Nginx on ubuntu
      ansible.builtin.apt:
        name: nginx
        state: present
        update_cache: true
      when: ansible_facts.os_family == "Debian"
      register: installation_info
    - name: View the output of the previous task
      ansible.builtin.debug:
        msg: "{{ installation_info }}"
    - name: Restart nginx service on ubuntu
      ansible.builtin.service:
        name: nginx
        state: restarted
      when: ansible_facts.os_family == "Debian" and installation_info.changed and not installation_info.failed
```

```
root@ip-10-2-0-220:~/lab1# ansible-playbook task3.yaml
BECOME password:

PLAY [Install and restart nginx] ***********************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************************
ok: [10.2.0.67]

TASK [Install Nginx on ubuntu] *************************************************************************************************************
ok: [10.2.0.67]

TASK [View the output of the previous task] *************************************************************************************************
ok: [10.2.0.67] => {
    "msg": {
        "cache_update_time": 1694894212,
        "cache_updated": true,
        "changed": false,
        "failed": false
    }
}

TASK [Restart nginx service on ubuntu] *****************************************************************************************************
skipping: [10.2.0.67]

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

root@ip-10-2-0-220:~/lab1#
```

HANDLERS task

1. install a package

(get the package name from vars)

2. copy a list of files (index.html & file.css & file.js) from controller to host using loop

(get the list of file names from vars)

(the actual files will be stored in ./your_current_directory)

(will be defined as Handler)

(will be triggered by step 1)

3. Restart the service of the installed package

(will be defined as Handler)

(will be triggered by step 2)

```
root@ip-10-2-0-220:~/lab1# ansible-playbook task4.yaml
BECOME password:

PLAY [handlertask] *************************************************************************************************************************

TASK [install nginx package] ***************************************************************************************************************
changed: [10.2.0.67]

RUNNING HANDLER [copyFiles_handler] ********************************************************************************************************
ok: [10.2.0.67] => (item=./handler/index.html)
ok: [10.2.0.67] => (item=./handler/styles.css)
ok: [10.2.0.67] => (item=./handler/script.js)

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

```
root@ip-10-2-0-220:~/lab1# curl 10.2.0.67
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Simple Website</title>
    <link rel="stylesheet" type="text/css" href="styles.css">
</head>
<body>
    <header>
        <h1>Simple Website</h1>
    </header>
    <div class="container">
        <h2>Welcome to my website!</h2>
        <p>This is a simple website with HTML, CSS, and JavaScript.</p>
        <button id="colorChangeButton">Change Background Color</button>
    </div>
    <script src="script.js"></script>
</body>
</html>

root@ip-10-2-0-220:~/lab1#
```

```
files_path: ./handler

files:
  - "{{ files_path }}/index.html"
  - "{{ files_path }}/styles.css"
  - "{{ files_path }}/script.js"
~
~
~
~
~
~
~
~
~
```

```yaml
- name: handlertask
  hosts: servers
  gather_facts: false
  tags: handler
  vars:
    package_name: nginx
  vars_files: ./handler.yaml
  tasks:

    - name: install {{ package_name }} package
      apt:
        name: "{{ package_name }}"
        state: present
      changed_when: true
      notify: copyFiles_handler

  handlers:
    - name: copyFiles_handler
      copy:
        src: "{{ item }}"
        dest: /var/www/html/
      loop: "{{ files }}"
      notify: topic

    - name: Restart service of {{ package_name }}
      listen: topic
      service:
        name: "{{ package_name }}"
        state: restarted
~
~
~
~
~
~
~
```

TEMPLATES task

1. installing a package

(get the package name from vars)

2. Copying index.html from controller to host using template

(get the template name & template message from vars)

(the actual template file will be stored in ./your_current_directory)

(will trigger step 4)

3. copying a list of files (file.css & file.js) from controller to host using loop

(get the list of file names from vars)

(the actual files will be stored in ./your_current_directory)

(will be defined as Handler)

(will be triggered by step 1)

4. Restart the service of the installed package

(will be defined as Handler)

(will be triggered by step 3

```
tamplete_name: index.html

tamplete_message: this is from ansible tamplete

tamplete_target_name: index.html
~
~
~
~
~
~
~
~
~
~
```

```
root@ip-10-2-0-220:~/lab1# ansible-playbook task5.yaml
BECOME password:

PLAY [tamplete] ****************************************************************************************************************************************************

TASK [install nginx package] ***************************************************************************************************************************************
changed: [10.2.0.67]

TASK [copy tamplete file index.html to the target machine] *********************************************************************************************************
ok: [10.2.0.67]

RUNNING HANDLER [copyFiles_handler] ********************************************************************************************************************************
changed: [10.2.0.67] => (item=./handler/index.html)
ok: [10.2.0.67] => (item=./handler/styles.css)
ok: [10.2.0.67] => (item=./handler/script.js)

RUNNING HANDLER [restartService_handler] ***************************************************************************************************************************
changed: [10.2.0.67]

PLAY RECAP *********************************************************************************************************************************************************
10.2.0.67                  : ok=4    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

root@ip-10-2-0-220:~/lab1#
```

```yaml
- name: tamplete
  hosts: servers
  gather_facts: false
  tags: tamplete
  vars:
    package_name: nginx
  vars_files: ./tamplete.yaml
  tasks:

    - name: install {{ package_name }} package
      apt:
        name: "{{ package_name }}"
        state: present
      changed_when: true
      notify: copyFiles_handler

    - name: copy tamplete file {{ tamplete_name }} to the target machine
      template:
        src: "./tamplete/{{ tamplete_name }}"
        dest: "/var/www/html/{{ tamplete_target_name }}"
      notify: restartService_handler
  handlers:
    - name: copyFiles_handler
      copy:
        src: "{{ item }}"
        dest: /var/www/html/
      loop: "{{ files }}"
      notify: restartService_handler

    - name: restartService_handler
      service:
        name: "{{ package_name }}"
        state: restarted
~
```