Create Directory

cd

mkdir apache-basic-playbook

cd apache-basic-playbook

|  |
| --- |
| ---  - hosts: web  name: This is a play within a playbook  become: yes  vars:  httpd\_packages:  - httpd  - wget  - vim |

vars: You’ve told Ansible the next thing it sees will be a variable name

httpd\_packages You are defining a list-type variable called httpd\_packages. What follows is a list of those packages

|  |
| --- |
| tasks:  - name: install httpd packages  yum:  name: "{{ item }}"  state: present  with\_items: "{{ httpd\_packages }}" |

{{ item }} You are telling Ansible that this will expand into a list item like httpd and mod\_wsgi.

with\_items: "{{ httpd\_packages }}” This is your loop which is instructing Ansible to perform this task on every item in httpd\_packages

|  |
| --- |
| - name: start apache  service:  name: httpd  state: started |

This is simple way of starting service

**Handlers:**

For handlers we will use previous code and use handler to start apache service

|  |
| --- |
| tasks:  - name: install apache  yum:  name: "{{ item }}"  state: present  with\_items: "{{ httpd\_packages }}"  notify: restart apache service  - name: Populate index file  shell: echo "Hello world 1234" >> /var/www/html/index.html  notify: restart apache service  changed\_when: false  - name: start apache  service:  name: httpd  state: started  enabled: yes  handlers:  - name: restart apache service  service:  name: httpd  state: restarted |

First task will install packages

Second task will insert hello world in index.html file

Third task will start and enable apache server

handler: This is telling the play that the tasks: are over, and now we are defining handlers:. Everything below that looks the same as any other task, i.e. you give it a name, a module, and the options for that module. This is the definition of a handler.

notify: restart apache service …​and here is your latter. Finally! The nofify statement is the invocation of a handler by name. Quite the reveal, we know. You already noticed that you’ve added a notify statement to the copy httpd.conf task, now you know why.

changed\_when: false: This will set changed flag to false all the time. This task will not count as change.