

REST API

- hosts: all

tasks:

- uri:

url: https://reqres.in/api/users?page={{ page_no }}

method: GET

register: result

failed_when: result.status != 200

- debug:

msg="{{ result.json.data[0] | json_query('first_name') }}"

```
[root@ip-172-31-27-155 Ansible]# ansible-playbook 45.yml -e "page_no=1"

PLAY [all] *****

TASK [Gathering Facts] *****
ok: [13.127.244.80]

TASK [uri] *****
ok: [13.127.244.80]

TASK [debug] *****
ok: [13.127.244.80] => {
  "msg": [
    "George",
    "Janet",
    "Emma"
  ]
}

PLAY RECAP *****
13.127.244.80 : ok=3 changed=0 unreachable=0 failed=0
```

ansible-playbook 45.yml -e "page_no=1"

ansible-playbook 45.yml -e "page_no=2"

Mail Communication

- hosts: localhost

tasks:

- name: Sending an e-mail using Gmail SMTP servers

mail:

host: smtp.gmail.com

port: 587

username: nepolean

password: pass123

to: kumar@gmail.com

subject: Ansible-report

body: System {{ ansible_hostname }} has been successfully provisioned.

delegate_to: localhost

ROLES

- Roles are ways of automatically loading certain vars_files, tasks, and handlers based on a known file structure. Grouping content by roles also allows easy sharing of roles with other users.
- Ansible also supports encapsulating Playbook tasks into reusable units called 'roles.'
- Roles are a packages of closely related Ansible content that can be shared more easily than plays alone.
 - ❖ Improves readability and maintainability of complex plays
 - ❖ Eases sharing, reuse and standardization of automation processes
 - ❖ Enables Ansible content to exist independently of playbooks, projects -- even organizations
 - ❖ Provides functional conveniences such as file path resolution and default values
- Ansible roles can be used to easily apply common configurations in different scenarios, such as having a common web server configuration role that may be used in development, test, and production automation.
- The Ansible Galaxy community site contains thousands of roles that can be used and customized to build Playbooks.

Ansible Galaxy

<https://galaxy.ansible.com/ui/standalone/roles/>

To download roles from the Ansible community
ansible-galaxy install rolename

Role Creation

Example Project Structure:

ansible-galaxy init Ansible_Role --offline # Ansible_Role => role_name

Roles expect files to be in certain directory names. Roles must include at least one of these directories, however it is perfectly fine to exclude any which are not being used. When in use, each directory must contain a main.yml file, which contains the relevant content:

- tasks** - contains the main list of tasks to be executed by the role.
- handlers** - contains handlers, which may be used by this role or even anywhere outside this role.
- defaults** - default variables for the role.
- vars** - other variables for the role.
- files** - contains files which can be deployed via this role.
- templates** - contains templates which can be deployed via this role.
- meta** - defines some meta data for this role.

```
site.yml
webservers.yml
fooservers.yml
roles/
  common/
    tasks/
    handlers/
    files/
    templates/
    vars/
    defaults/
    meta/
  webservers/
    tasks/
    defaults/
    meta/
```