1. Ansible sibelius - Try It Out-conditional (when clause)

In this scenario, you have to create a directory "test" at /home/user using ansible-adhoc command. Don't use any playbook.

Tasks to be done:
- Create a file named simplefile.txt
- write a task in the main.yml file in present in **fresco_when\tasks** folder.
- the task is to move the created simplefile.txt file to created directory i.e move the create file to /home/user/test folder.
- move only if the file doesn't exists. using **when** in playbook

Note:

- Run project ->install to install ansible.
- mainplaybook.yml file is provided to ansible-playbook.
- Use the localhost for the inventory for ansible-playbook.

In this scenario, you have to create a directory "test" at /home/user using ansible-adhoc command. Don't use any playbook.

```
user@workspacew9as5z00akhb7drn:~$ ansible localhost -m file -a 'path=/home/user/test state=directory'
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host localhost should use
/usr/bin/python3, but is using /usr/bin/python for backward compatibility with
prior Ansible releases. A future Ansible release will default to using the
discovered platform python for this host. See https://docs.ansible.com/ansible/
2.9/reference_appendices/interpreter_discovery.html for more information. This
feature will be removed in version 2.12. Deprecation warnings can be disabled
by setting deprecation_warnings=False in ansible.cfg.
localhost | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "gid": 0,
    "group": "root",
    "mode": "0755",
    "owner": "user",
    "path": "/home/user/test",
    "size": 4096,
    "state": "directory",
    "uid": 1000
}
```

```yaml
---
- hosts: localhost
  connection: local
  become: yes
  become_method: sudo
  roles:
    - fresco_when

```

- name: Solution for when clause
  hosts: localhost
  become: yes
  tasks:
  - name: check if file exists
    stat:
      path: “”
    register: f1
  - name: copy file
    copy:
      src:
      dest:
    when: not f1.stat.exists

…

```yaml
---
- name: Solution
  hosts: localhost
  become: yes
  tasks:
  - name: Check if file exists
    stat:
      path="/simplefile.txt"
    register: f1
  - name: Create file
    copy:
      src: /simplefile.txt
      dest: /home/user/test
    when: not f1.stat.exists
```

With above ansible-playbook command failed as full path for simplefile.txt was not provided .

Correct paybook :

main.yml × simplefile.txt

```yaml
---
- name: Solution
  hosts: localhost
  become: yes
  tasks:
  - name: Check if file exists
    stat:
      path="/simplefile.txt"
    register: f1
  - name: Create file
    copy:
      src: /projects/challenge/fresco_when/tasks/simplefile.txt
      dest: /home/user/test/
    when: not f1.stat.exists
...
```

- `stat:` Declares that we are using the stat module.
- `path:` Declares the path to the file or folder we want to check.
- `register:` Provides the name of the register where the stat module saves file and folder details.

```
user@workspacemu0xv3kwqt9cvtcv:/projects/challenge/fresco_when/tasks$ ansible-playbook main.yml

PLAY [Solution] ****************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host localhost should use /usr/bin/python3, but is using
/usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to
using the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This
feature will be removed in version 2.12. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
ok: [localhost]

TASK [Check if file exists] ****************************************************************************************
ok: [localhost]

TASK [Create file] *************************************************************************************************
changed: [localhost]

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

```
user@workspacemu0xv3kwqt9cvtcv:/projects/challenge/fresco_when/tasks$ cd /home/user/test
user@workspacemu0xv3kwqt9cvtcv:~/test$ ls
simplefile.txt
user@workspacemu0xv3kwqt9cvtcv:~/test$
```

File is copied .

As we know ansible is idempotent . If we again try to run playbook command changes will not be applied again .

```
user@workspacemu0xv3kwqt9cvtcv:/projects/challenge/fresco_when/tasks$ ansible-playbook main.yml

PLAY [Solution] ****************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host localhost should use /usr/bin/python3, but is using
/usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using
the discovered platform python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature
will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in
ansible.cfg.
ok: [localhost]

TASK [Check if file exists] ****************************************************************************************
ok: [localhost]

TASK [Create file] *************************************************************************************************
ok: [localhost]

PLAY RECAP *********************************************************************************************************
localhost                  : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

1. Ansible install nginx and postgresql

In this scenario, you have to Install nginx and postgresql using ansible-playbook.

- tasks for installing nginx and postgresql should be written in **fresco_nginx\tasks\main.yml** file.

Note:

- Run project install to install ansible.
- mainplaybook.yml file is provided to ansible-playbook.
- Use the localhost for the inventory for ansible-playbook.

```
---
#installing nginx and postgresql
 - name: install nginx and postgresql
   hosts: localhost
   become: true
   tasks:
   - name: install nginx
     apt:
       name: nginx
       state: latest
   - name: install postgresql
     apt:
       name: postgresql
       state: latest
   - name: start nginx
     service:
       name: nginx
       state: started
   - name: start postgresql
     service:
       name: postgresql
       state: started
...
```

```
PLAY [install nginx and postgresql] ********************************************************

TASK [Gathering Facts] *********************************************************************
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host localhost should use
/usr/bin/python3, but is using /usr/bin/python for backward compatibility with prior
Ansible releases. A future Ansible release will default to using the discovered platform
python for this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for
more information. This feature will be removed in version 2.12. Deprecation warnings can be
 disabled by setting deprecation_warnings=False in ansible.cfg.
ok: [localhost]

TASK [install nginx] ***********************************************************************
changed: [localhost]

TASK [install postgresql] ******************************************************************
changed: [localhost]

TASK [start nginx] *************************************************************************
ok: [localhost]

TASK [start postgresql] ********************************************************************
changed: [localhost]

PLAY RECAP *********************************************************************************
localhost                  : ok=5    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0


user@workspaceyrrvv9h9iq6l4g8u:/projects/challenge/fresco_nginx/tasks$ ansible-playbook main.yml
```

---

1. Ansible Sibelius - Try It Out - Loops

In this scenario, you have to install apache2, sqlite3, git using loops,

- tasks for installing apache2, sqlite3, git should be written in fresco_loops\tasks\main.yml

Disclaimer: This scenario always installs the latest version of ansible, if Invoking "apt" only once while using a loop via squash_actions is not working or deprecated, Try to use the pkg module while looping.

Note:

- Run project install to install ansible.
- mainplaybook.yml file is provided to ansible-playbook.
- Use the localhost for the inventory for ansible-playbook.

```yaml
---
- name: install apache2, sqlite3, git pn remote server
  hosts: localhost
  become: yes
  tasks:
  - name: Install list of packages
    action: apt pkg={{item}} state=latest
    with_items:
      - apache2
      - sqlite3
      - git
```

```
user@workspaceghny3x64ufzp55gu:/projects/challenge/fresco_loops/tasks$ ansible-playbook main.yml

PLAY [install apache2, sqlite3, git pn remote server] ***************************************

TASK [Gathering Facts] **********************************************************************
[DEPRECATION WARNING]: Distribution Ubuntu 16.04 on host localhost should use
/usr/bin/python3, but is using /usr/bin/python for backward compatibility with prior Ansible
 releases. A future Ansible release will default to using the discovered platform python for
 this host. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for
more information. This feature will be removed in version 2.12. Deprecation warnings can be
disabled by setting deprecation_warnings=False in ansible.cfg.
ok: [localhost]

TASK [Install list of packages] *************************************************************
[DEPRECATION WARNING]: Invoking "apt" only once while using a loop via squash_actions is
deprecated. Instead of using a loop to supply multiple items and specifying `pkg:
"{{item}}"`, please use `pkg: ['apache2', 'sqlite3', 'git']` and remove the loop. This
feature will be removed in version 2.11. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
changed: [localhost] => (item=[u'apache2', u'sqlite3', u'git'])

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

Question Type: **Single-Select**

```
vars:
  score: 3
tasks:
  - shell: echo "Bang on!! You won."
    when: ??
```

How do you use a variable to apply condition?

- when: score == 3
- when: ' score ' == 3
- when: " score " == 3
- when: {{ score }} == 3

You cannot define multiple conditions.

- True
- False

Ansible modules will apply changes everytime you run Playbook.

- True
- False

Ansible can deploy virtualization and cloud environments, including _____

- Google Cloud platform
- Amazon Web Services
- Cloud Stack
- Open Stack
- All the options

Which module will you use to create a directory?

- fetch
- template
- copy
- file

Which module you can use to install Apache in Ubuntu OS?

- yum
- apt

```
tasks:
- name: test
  ## some code ##
  notify:
  - handler_1
  - handler_2
  - handler_3
- name: test1
  ## some code##
  notify:
  - handler_1

handlers:
- name: handler_2
## some code ##
- name: handler_3
- name: handler_1
```

test-handler_2-handler_3-handler_1-test1

test-test1-handler_2-handler_3-handler_1

test-handler_1-handler_2-handler_3-test1-handler_1

test-test1-handler_1-handler_2-handler_3

YAML uses tabs for indentation.

True

False

Question Type: **Single-Select**

_____ is a valid YAML syntax.

path: "F\test"\programs

path: F:

All the options

path: F:\test

Which command do you use to do a syntax check on your Playbook?

- ansible-playbook <playbook_name> --syntax--check
- ansible-playbook <playbook_name> -syntax-check
- ansible-playbook <playbook_name> -syntax--check
- ansible-playbook <playbook_name> --syntax-check

Using which module can you see the list of all the Ansible variables?

- ping
- copy
- fetch
- setup

Nodes are managed by a controlling machine over _____.

- HTTPs
- HTTP
- SSH
- FTP

ansible.cfg should be present in _____.

/etc/configurations

/ansible/

/etc/hosts

/etc/ansible

A Playbook starts with three dots ... .

False

True

You can write comments in Jinja2 as _____.

{# #}

{% %}

{{ }}

Ansible is a/an _____.

Orchestration Engine

All the options

Infrastructure as code

Configuration management

Ansible has two types of servers. Select the appropriate answer from the given options.

Controlling machines and nodes

Only node