

# TESTPASSPORT

## 練習題



質 量 更 高

服 務 更 好

<https://www.testpassport.net>

免費一年更新服務

**Exam : EX447**

**Title : Red Hat Certified Specialist  
in Advanced Automation:  
Ansible Best Practices**

**Version : DEMO**

### 1.CORRECT TEXT

Create a file called specs.empty in home/bob/ansible on the local machine as follows:

HOST=

MEMORY=

BIOS=

VDA\_DISK\_SIZE=

VDB\_DISK\_SIZE=

Create the playbook /home/bob/ansible/specs.yml which copies specs.empty to all remote nodes' path /root/specs.txt. Using the specs.yml playbook then edit specs.txt on the remote machines to reflect the appropriate ansible facts.

**Answer:**

Solution as:

```
- name: edit file
hosts: all
tasks:
  - name: copy file
    copy: report.txt
    dest: /root/report.txt
  - name: change host
    lineinfile:
      regex: ^HOST
      line: HOST={{ansible_hostname}}
      state: present
      path: /root/report.txt
  - name: change mem
    lineinfile:
      line: MEMORY={{ansible_memtotal_mb}}
      regex: ^MEMORY
      state: present
      path: /root/report.txt
```

```

- name: change bios
  lineinfile:
    line: BIOS={{ansible_bios_version}}
    regex: ^BIOS
    state: present
    path: /root/report.txt
- name: change vda
  lineinfile:
    line: VDA_DISK_SIZE ={%if ansible_devices.vda is defined%}{{ansible_devices.vda.size}}{%else%}NONE{%endif%}
    regex: ^VDA_DISK_SIZE
    state: present
    path: /root/report.txt
- name: change vdb
  lineinfile:
    line: VDB_DISK_SIZE ={%if ansible_devices.vdb is defined%}{{ansible_devices.vdb.size}}{%else%}NONE{%endif%}
    regex: ^VDB_DISK_SIZE
    state: present
    path: /root/report.txt

```

## 2.CORRECT TEXT

Create a file called `adhoc.sh` in `/home/sandy/ansible` which will use `ad-hoc` commands to set up a new repository. The name of the repo will be 'EPEL' the description 'RHEL8' the baseurl is `https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm` there is no `gpgcheck`, but you should enable the repo.

\* You should be able to use an `bash` script using `ad-hoc` commands to enable repos. Depending on your lab setup, you may need to make this repo "state=absent" after you pass this task.

### Answer:

```
chmod 0777 adhoc.sh
```

```
vim adhoc.sh
```

```
#!/bin/bash
```

```
ansible all -m yum_repository -a 'name=EPEL description=RHEL8
```

```
baseurl=https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm gpgcheck=no
enabled=yes'
```

## 3.CORRECT TEXT

Create a file called `requirements.yml` in `/home/sandy/ansible/roles` to install two roles. The source for the first role is `geerlingguy.haproxy` and `geerlingguy.php`. Name the first `haproxy-role` and the second `php-role`. The roles should be installed in `/home/sandy/ansible/roles`.

**Answer:**

```
in /home/sandy/ansible/roles
vim requirements.yml
```

```
- src: geerlingguy.haproxy
  name: haproxy-role
- src: geerlingguy.php_role
  name: php_role
```

Run the requirements file from the roles directory:

```
ansible-galaxy install -r requirements.yml -p /home/sandy/ansible/roles
```

**4.CORRECT TEXT**

Create an empty encrypted file called myvault.yml in /home/sandy/ansible and set the password to notsafepw. Rekey the password to iwejff2221.

**Answer:**

```
ansible-vault create myvault.yml
```

```
Create new password: notsafepw Confirm password: notsafepw
ansible-vault rekey myvault.yml
```

```
Current password: notsafepw New password: iwejff2221 Confirm password: iwejff2221
```

**5.CORRECT TEXT**

Create a playbook called webdev.yml in /home/sandy/ansible. The playbook will create a directory /var/www/html/webdev on dev host. The permission of the directory are 755 and owner is webdev. Create a symbolic link from /var/www/html/webdev to /var/www/html/webdev7. Serve a file from /var/www/html/webdev7/index.html which displays the text "Development". Curl http://node1.example.com/webdev/index.html to test

**Answer:**

Solution as:

```
- name: webdev
hosts: dev
tasks:
  - name: create webdev user
    user:
      name: webdev
      state: present
  - name: create a directory
    file:
      mode: '2755'
      path: /webdev
      state: directory
  - name: create symbolic link
    file:
      src: /webdev
      path: /var/www/html/webdev
      state: link
  - name: create index.html
    copy:
      content: Development
      dest: /webdev/ index.html
  - name: Install selinux policies
    yum:
      name: python3-policycoreutils
      state: present
  - name: allow httpd from this directory
    sefcontext:
      target: '/webdev(/.*)?'
      setype: httpd_sys_content_t
      state: present
  - name: restore the context
    shell: restorecon -vR /webdev
```