

Ansibleハンズオン * 2023年1月16日

◆目次

- [ハンズオン環境](#)
- 演習
 - [演習①: Ansible CLIにてPlaybookを実行](#)
 - [演習②: Ansible GUIにてテンプレート\(Playbook\)を実行](#)
 - [演習③: 対話形式の入力フォーム作成\(サーベイ\)を作成し、実行](#)
 - [演習④: 簡単なワークフローを作成し、実行](#)

◆ハンズオン環境

- ハンズオン環境ログインURL
<http://f67jk.example.opentlc.com>

1. ハンズオン環境へのログイン

- 1.1. [ハンズオン環境ログインURL](#)に記載されているURLへアクセス。NameとEmailの入力を求められるので、お名前(ローマ字)および会社のEmailアドレスを入力ください。入力後Submitボタンをクリック。

Workbench Information

Please enter your Name and Email address.

Name:	<input type="text"/>
Email:	<input type="text"/>
	<input type="submit" value="Submit"/>

- 1.2. 下記ページが表示されるので、VS CodeとAutomation Controllerへアクセスし、ログイン。(※Private Automation HubおよびSSH accessは、使用しないため不要です。)

Workbench Information

Student1

VS Code access


To login to Visual Studio Code via your web browser please go here:

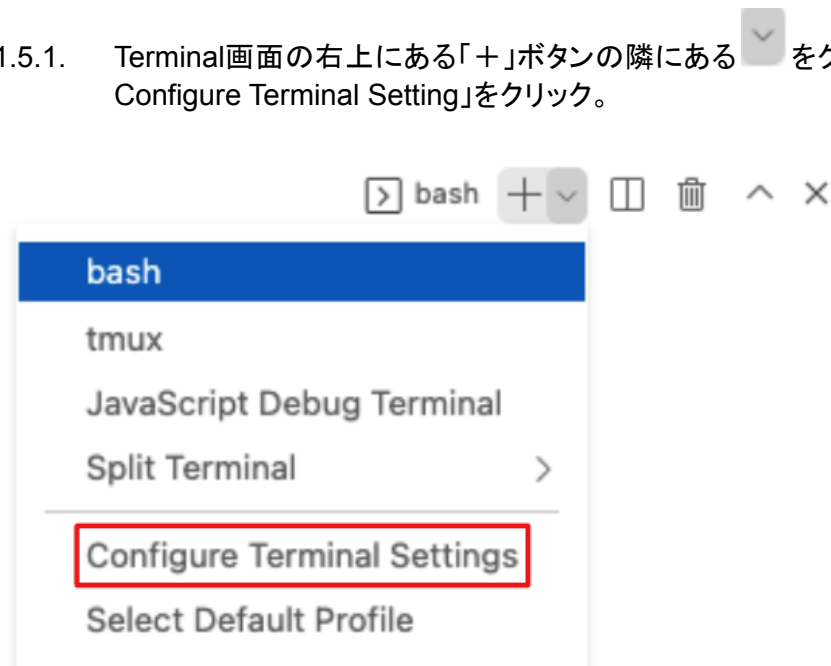
WebUI link:	https://student1-code.rhbeb3.example.opentlc.com
password:	<input type="password"/>

Automation controller

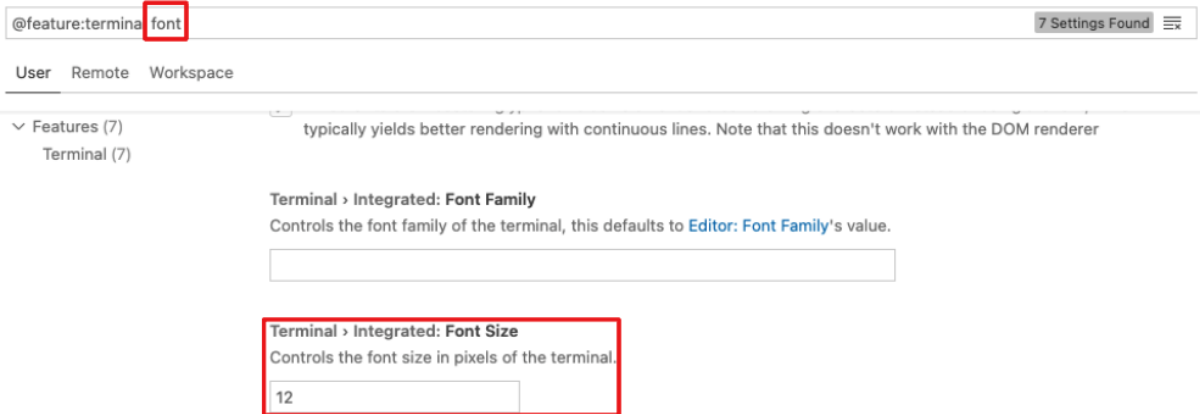
To login to the Automation controller WebUI use the following credentials:

WebUI link:	https://student1.rhbeb3.example.opentlc.com
username:	admin
password:	<input type="password"/>

- 1.3. Automation Controllerは、ログイン完了したら、そのままにしておいてください。
- 1.4. VS Codeに関しては、下記ページを参照して設定をお願いします。
https://aap2.demoredhat.com/exercises/ansible_rhel/1.1-setup/README.ja.html#step-2---using-the-terminal
「ステップ 1 - 環境へのアクセス」と「ステップ 2 - ターミナルの使用」の実施をお願いします。ステップ1 & 2以外の実施は不要です。
- 1.5. (Option) Terminalの文字が小さい場合、下記設定でフォントの大きさを変更可能です。
 - 1.5.1. Terminal画面の右上にある「+」ボタンの隣にあるをクリックし、「Configure Terminal Setting」をクリック。



- 1.5.2. 上部の検索欄に「@feature:terminal」の後ろにスペースを空けて「font」と入力。Font Sizeを入力できる項目がでてくるので、最適なフォントサイズを入力。



The screenshot shows the Red Hat settings application. At the top, there is a search bar with the text '@feature:terminal font' entered. Below the search bar, there are tabs for 'User', 'Remote', and 'Workspace'. The 'Features (7)' section is expanded, showing 'Terminal (7)'. Under 'Terminal (7)', there are two settings: 'Terminal > Integrated: Font Family' and 'Terminal > Integrated: Font Size'. The 'Font Size' setting is highlighted with a red box and shows a value of '12'.

事前準備は、以上となります。お疲れさまでした！！

◆ 演習

演習①: Ansible CLIにてPlaybookを実行

1. インベントリファイルの確認
 - 1.1. VSCodeからTerminalを開く
 - 1.1.1. Terminalを開くとAnsible Controllerへ自動的に接続されます
 - 1.2. `$ cat /home/student/lab_inventory/hosts`
 - 1.3. 出力例: * Webグループの対象を確認。今回Webグループに対して実行

```
[web]
node1 ansible_host=18.139.163.182
node2 ansible_host=54.254.223.63
node3 ansible_host=13.215.207.145

[control]
ansible-1 ansible_host=18.136.104.184
```
2. 実行するPlaybookの確認
 - 2.1. `$ cd /home/student/rhel-workshop/1.3-playbook`
 - 2.2. `$ cat apache.yml`
実行するPlaybookの内容を確認する。
-name: copy index.htmlでsrc(ソース)で指定されている"web.html"の内容は、次のステップで確認する。

```
---
- name: Apache server installed
  hosts: web
  become: true
  tasks:

  - name: latest Apache version installed
    yum:
      name: httpd
      state: latest

  - name: Apache enabled and running
    service:
      name: httpd
      enabled: true
      state: started

  - name: copy index.html
    copy:
      src: web.html
      dest: /var/www/html/index.html
```

3. Web.htmlの確認

```
<body>
<h1>Apache is running fine</h1>
</body>
```

4. Playbookの実行

4.1. \$ ansible-navigator run apache.yml -m stdout

※failed=0で完了していることを確認

補足:

- ok → タスクが成功
- changed → タスクでなにか変化が起きた時のステータス

```
PLAY [Apache server installed] *****

TASK [Gathering Facts] *****
ok: [node2]
ok: [node3]
ok: [node1]

TASK [latest Apache version installed] *****
changed: [node1]
changed: [node2]
changed: [node3]

TASK [Apache enabled and running] *****
changed: [node1]
changed: [node3]
changed: [node2]

TASK [copy index.html] *****
changed: [node3]
changed: [node1]
changed: [node2]

PLAY RECAP *****
node1      : ok=4  changed=3  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
node2      : ok=4  changed=3  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
node3      : ok=4  changed=3  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

5. VSCodeのTerminalから対象ノード(RHEL)へcurlコマンドを実施し、“3.Web.htmlの確認”で確認した内容が表示されることを確認

5.1. \$ curl http://node1

5.1.1. (時間があれば)node2,node3でも実行

早く終わった方は以下参考ページをご確認ください。

参考ページ:

https://aap2.demoredhat.com/exercises/ansible_rhel/1.3-playbook/README.ja.html

演習①は以上となります。お疲れさまでした！！

演習②: Ansible GUIにてテンプレート(Playbook)を実行

1. 事前準備の2.3でログインしたAnsible Controllerページを開く
 - 1.1. 自動的にログアウトされている場合は、再度パスワードを使用してログイン
2. 演習②では、以下のページに沿って実施
 - 2.1. https://aap2.demoredhat.com/exercises/ansible_rhel/2.3-projects/README.html
 - 2.2. チャレンジラボまで実施

目次

- 目的
- ガイド
- Git リポジトリのセットアップ
- プロジェクトの作成
- ジョブテンプレートの作成とジョブの実行
- チャレンジラボ: 結果のチェック

- 2.3. 演習のゴール/実行結果としては、下記のように、Active項目にactive(running)となっていれば成功

```
node1 | CHANGED | rc=0 >>
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2022-08-22 05:00:42 UTC; 6h ago
     Docs: man:httpd.service(8)
  Main PID: 25334 (httpd)
    Status: "Total requests: 22; Idle/Busy workers 100/0;Requests/sec: 0.000908; Bytes served/sec: 0 B/sec"
    Tasks: 278 (limit: 4539)
   Memory: 39.6M
    CGroup: /system.slice/httpd.service
            └─25334 /usr/sbin/httpd -DFOREGROUND
            └─25335 /usr/sbin/httpd -DFOREGROUND
            └─25336 /usr/sbin/httpd -DFOREGROUND
            └─25337 /usr/sbin/httpd -DFOREGROUND
            └─25338 /usr/sbin/httpd -DFOREGROUND
            └─25835 /usr/sbin/httpd -DFOREGROUND
```

演習②は以上となります。お疲れさまでした！！！！

演習③: 対話形式の入力フォーム作成(サーベイ)を作成し、実行

1. 事前準備の2.3でログインしたAnsible Controllerページを開く
 - 1.1. 自動的にログアウトされている場合は、再度パスワードを使用してログイン
2. 演習③では、以下のページに沿って実施
 - 2.1. https://aap2.demoredhat.com/exercises/ansible_rhel/2.4-surveys/README.ja.html

目次

- 目的
- ガイド
- Apache-configuration ロール
- Survey によるテンプレートの作成
 - テンプレートの作成
 - Survey の追加
- テンプレートの起動

注意点:「Survey を持つテンプレートの作成」の一番下に「Survey の Preview をクリックします。」の記載がありますが、ここは無視してください。

- 2.2. 演習のゴール/実行結果としては、下記のように表示されれば成功
Surveyでは、以下の通り入力した場合(例)
First_Line :Red
Second_Line:Hat

VSCodeのTerminalから以下のコマンド実行

\$curl http://node1

```
<html>
<body>
<h1>Apache is running fine</h1>
<h1>This is survey field "First Line": Red </h1>
<h1>This is survey field "Second Line": Hat</h1>
</body>
</html>
```

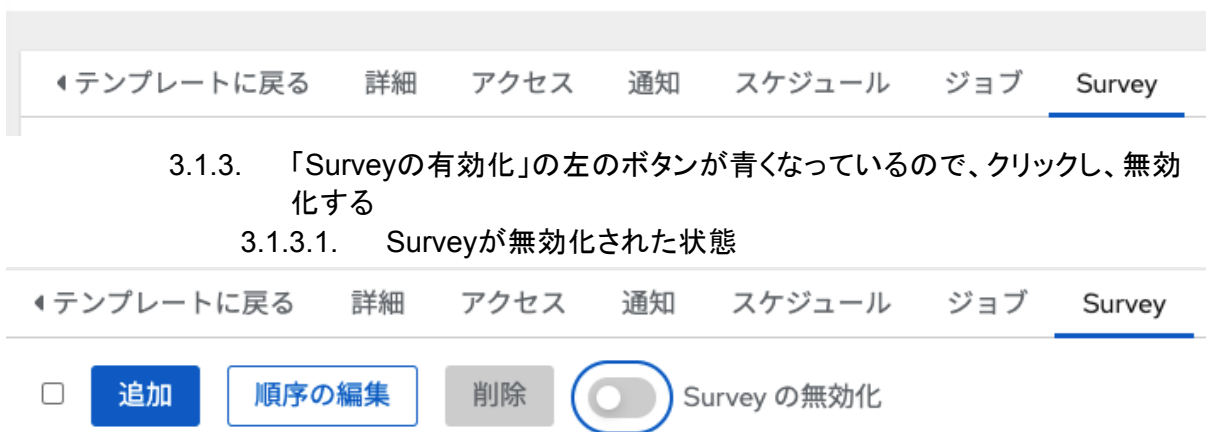
演習③は以上となります。お疲れさまでした！！！！

演習④: 簡単なワークフローを作成し、実行

1. 事前準備の2.3でログインしたAnsible Controllerページを開く
 - 1.1. 自動的にログアウトされている場合は、再度パスワードを使用してログイン
2. 現状のテンプレートの確認
 - 2.1. 演習②で作成した「Install Apache」テンプレートがあることを確認
 - 2.2. 演習②で作成した「Create index.html」テンプレートがあることを確認
3. 事前作業
 - 3.1. 演習②で作成した「Create index.html」テンプレートのSurveyを無効化
無効化する理由としては、ワークフローのSuveryで設定するためです。
 - 3.1.1. 「Create index.html」テンプレートをクリック
 - 3.1.2. 詳細が表示されてるので、「Survey」タブをクリック

[テンプレート](#) > [Create index.html](#)

Survey



4. ワークフロー作成
 - 4.1. テンプレートに戻り、「追加」をクリックし、「ワークフローテンプレートの追加」を選択

テンプレート



4.2. 「新規ワークフローテンプレートの作成」が表示される

テンプレート

新規ワークフローテンプレートの作成

🔍

名前 *	説明	組織
<input type="text"/>	<input type="text"/>	<input type="text"/>
インベントリ ②	制限 ②	ソースコントロールのプランチ ②
<input type="text"/>	<input type="text"/>	<input type="text"/>
ラベル ②	<input type="text"/>	

4.3. 以下情報を記入し、画面したにある”保存”をクリック

4.3.1. 名前: Workflow

4.3.2. インベントリ: Workshop Inventoryを選択

4.4. “保存”を押した後、下記画面が表示されるので”開始”をクリック

開始ボタンをクリックして開始してください。

[開始](#)

4.5. 下記が表示される。

ノードの追加 ×

ノードタイプ ジョブテンプレート ▼

名前 ▼ 🔍

名前 1

- ☐ Create index.html
- ☐ INFRASTRUCTURE / Deploy Application
- ☐ Install Apache

<< < 1 / 1 ページ > >>

収束 (コンバージェンス) * ②

任意 ▼

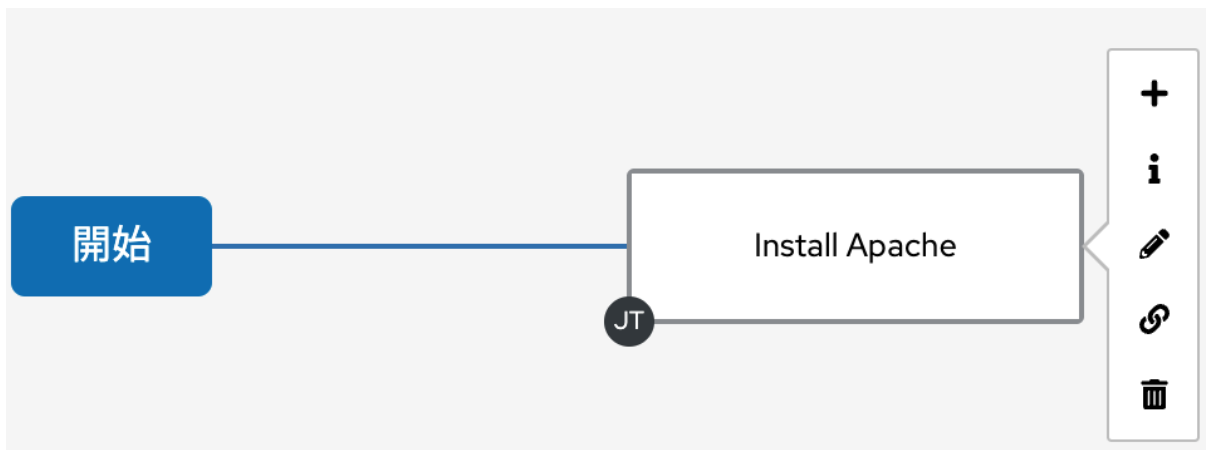
保存 取り消し

4.6. 1つ目のテンプレートを追加

4.6.1. 作成したテンプレートが表示されているので、「Install Apache」テンプレートを選択し、保存

4.7. 2つ目のテンプレートを追加

4.7.1. 下記画面が表示され、「Install Apache」テンプレートにカーソルを合わせる(もしくはクリック)すると、吹き出しがでる



- 4.8. 吹き出しの一番上にある、“+”をクリック
- 4.9. 下記画面が表示されるので、“成功時”を選択し、“次へ”をクリック
- 4.9.1. 成功時にのみ次のステップに進む設定です。

ノードの追加
×

- 1 実行タイプ
- 2 ノードタイプ

実行
このノードを実行する条件を指定

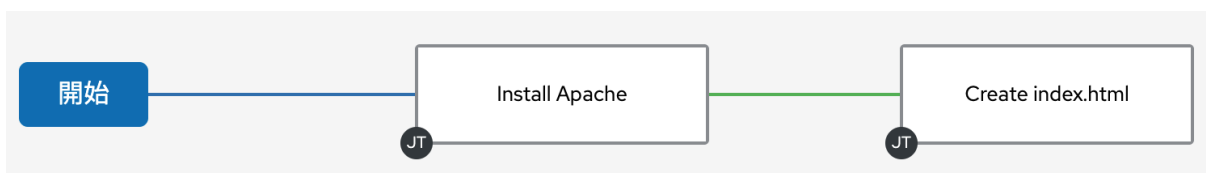
成功時
親ノードが正常な状態になったときに実行します。

障害発生時
親ノードが障害状態になったときに実行します。

常時
親ノードの最終状態に関係なく実行します。

次へ
取り消し

- 4.10. ノードの追加画面が表示されるので、「Create index.html」テンプレートを選択し、保存。



- 4.11. 画面右上に“保存”ボタンがあるのでクリック
- 4.12. Workflowのテンプレート詳細が表示されるので、演習③で実施した通り、Surveyを入力し、有効化。

Survey の追加

- テンプレートで **Survey** タブをクリックして、**Add** ボタンをクリックします。
- 次の情報を入力します。

パラメーター	値
Question	First Line
Answer Variable Name	first_line
Answer Type	Text

- **Save** をクリックします。
- **追加** ボタンをクリックします。

同じ方法で、2 番目の **Survey Question** を追加します。

パラメーター	値
Question	Second Line
Answer Variable Name	second_line
Answer Type	Text

- **Save** をクリックします。
- トグルをクリックして Survey の質問を **On** に切り替えます。
- Survey の **Preview** をクリックします。

5. ワークフローの実行

5.1. テンプレート画面に戻り、"Workflow"テンプレート実行。Workflow欄の右側にある



をクリック。

5.2. Surveyを入力し、"起動"ボタンを押して、"Workflow"テンプレートを実行

5.3. 出力画面が表示され、書くテンプレートが順々に実行されていることが確認できます。

6. 演習のゴール/実行結果としては、下記のように表示されれば成功

Surveyでは、以下の通り入力した場合(例)

First_Line : Red

Second_Line : Hat

VSCodeのTerminalから以下のコマンド実行

```
$curl http://node1
```

```
<html>
<body>
<h1>Apache is running fine</h1>
<h1>This is survey field "First Line": Red </h1>
<h1>This is survey field "Second Line": Hat</h1>
</body>
</html>
```

演習④は以上となります。お疲れさまでした！！

演習は、以上で全て終了となります。