

Algumas funções específicas de bibliotecas, como a `django-admin`, não estão presentes nativamente no Ansible, então podemos utilizar o parâmetro `shell:` para executar as funções que queremos.

Para utilizarmos o `shell:` devemos criar uma nova tarefa e colocar todos os comandos que queremos executar dentro de aspas, simples ou duplas, ou de acentos graves, assim o Ansible consegue distinguir o que deve ser executado.

Então a linha de comando pode ser escrita das seguintes formas:

```
- name: Iniciando o projeto
  shell: 'comando 1; comando 2; comando 3'
```

[COPIAR CÓDIGO](#)

```
- name: Iniciando o projeto
  shell: 'comando 1'
  shell: 'comando 2'
```

[COPIAR CÓDIGO](#)

Com todos os comandos sendo escritos em uma única linha, e separados por “;”, eles são executados em sucessão, então a tarefa:

```
- name: Iniciando o projeto
  shell: 'cd /home/ubuntu; cat index.html'
```

[COPIAR CÓDIGO](#)

Começará indo até a pasta `/home/ubuntu`, e em seguida mostrará o conteúdo do `index.html` que está na pasta `/home/ubuntu`.

Já se escrevermos o código em mais de uma linha, desta forma:

```
- name: Iniciando o projeto
  shell: 'cd /home/ubuntu'
  shell: 'cat index.html'
```

[COPIAR CÓDIGO](#)

Os comandos anteriores não interferem no novo, então o primeiro comando irá até a pasta `/home/ubuntu`, enquanto o segundo comando será executado em `/`, então devemos tomar cuidado com como escrevemos os comandos `shell`.