

DevOps (CS457)

ASSIGNMENT 2: Task 4

Ansible lineinfile Examples – Add, Modify, Delete, Replace lines

Submitted to:

Dr. Uma S

Submitted by:

Team 1

Sumith Sai Budde (18BCS101)

Syed Sufyan Ahmed (18BCS103)

Shaik Fharook (18BCS091)

Parvati Jayakumar (18BEC036)

P Chethan Krishna (18BEC040)

G Rithika (18BCS031)

Pokala Dattatreya (18BCS067)

Rama Dundi Saketh (18BCS076)

Ansible lineinfile module is helpful when we want to add, remove, modify a single line in a file. We can also use conditions to match the line before modifying or removing using the regular expressions. We can reuse and modify the matched line using the back reference parameter.

Objectives:

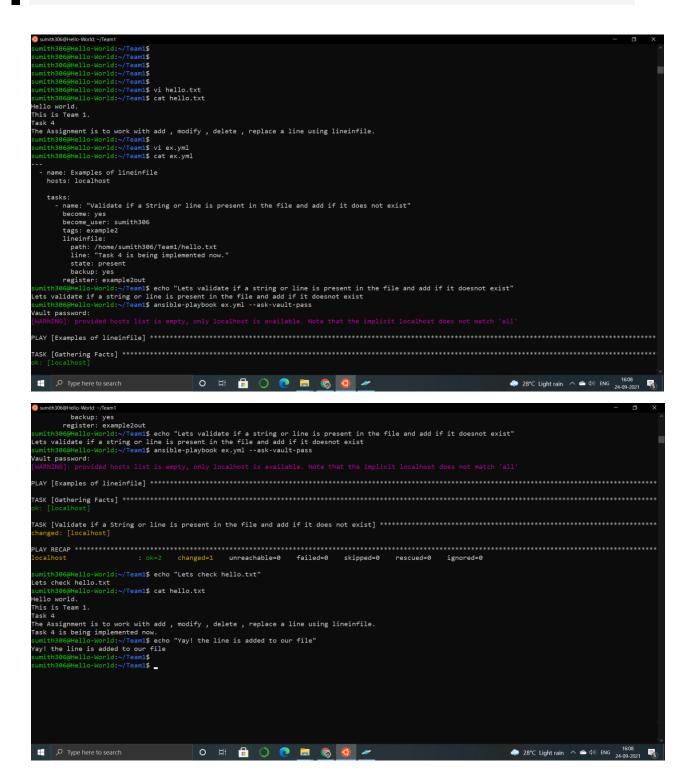
- Validate if a line is present without any modification
- Validate if a line is present in the file and add if it does not exist
- Replace a Line in a file
- **Remove** a line from the file if it is found All the instances
- Commenting a Line
- Insert before a matching line using insertbefore parameter
- Insert after the matching line using **insertafter** parameter
- Replacing Multiple lines at once using lineinfile module

1: Validate if a line is present in the file without any modification

Note: This does not modify the file and just checks if the line is present and returns the line.

```
## A STATE OF THE PROPERTY OF
```

2: Validate if a String or line is present in the file and add if it does not exist

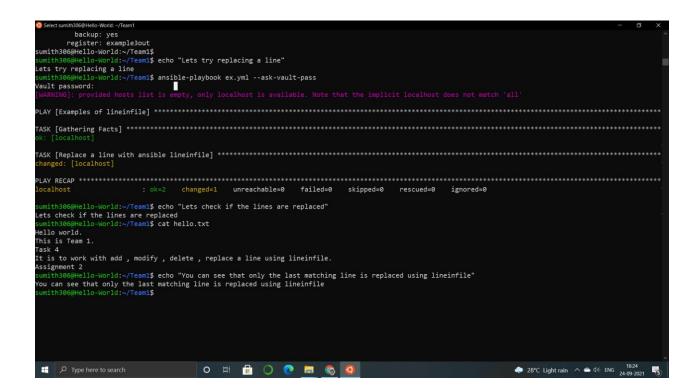


The playbook searches for a line and adds it when there are no matches found.

Note: The line we are mentioning to be added would be added only at the End Of File or Last line.

3: Replacing a line in a file

```
## Second Provided How Provided
```



We are using regular expressions to search for the line and replace it .

Note: Only the last matching line will be replaced using lineinfile as you can see above.

4: Remove a line from the file, if it is found

```
umith306@Hello-World:~/Team1$
 umith306@Hello-World:~/Team1$ vi hello.txt
          Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
Task 4
It is to work with add , modify , delete , replace a line using lineinfile.
Task 4 on the way
sumith306@Hello-World:~/Team1$ vi ex.yml
 umith306@Hello-World:~/Team1$ cat ex.yml
 - name: Examples of lineinfile
   hosts: localhost
    tasks:
      - name: "Remove a line from the file if it is found"
       become: yes
       become_user: sumith306
        tags: example4
        lineinfile:
          path: /home/sumith306/Team1/hello.txt
          #The string to search:
regexp: "Task 4"
#The string to replace
          state: absent
          backup: yes
        register: example4out
```

```
world:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
  NING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all
hanged: [localhost]
ocalhost
              : ok=2 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
umith306@Hello-World:~/Team1$ echo "Lets check if the lines are removed"
Lets check if the lines are removed
     Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
t is to work with add , modify , delete , replace a line using lineinfile.
umith306@Hello-World:~/Team1$
umith306@Hello-World:~/Team1$ echo "You can see that all the lines with string Task 4 are removed"
ou can see that all the lines with string Task 4 are removed
                                                        O # 🗄 🔾 🧖 🧞 🧕 🛩
Type here to search
```

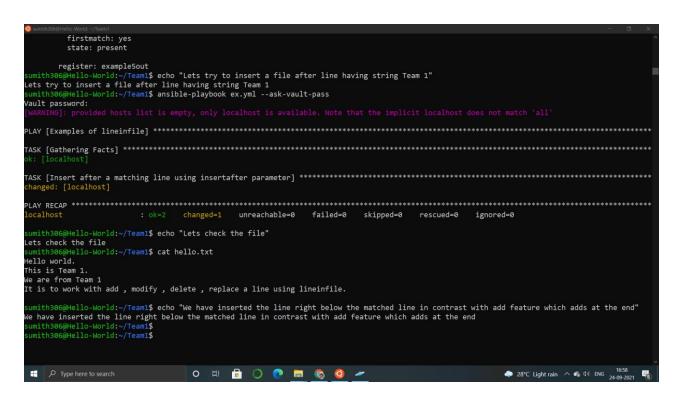
We are using regular expression to search for the string and remove the lines.

Note: All instances or lines which match the regexp are removed.

5: Insert After a matching line using insertafter parameter

```
umith306@Hello-World:~/Team1$
umith306@Hello-World:~/Team1$
    umith306@Hello-World:~/Team1$
    umith306@Hello-World:~/Team1$ vi hello.txt
umith306@Hello-World:~/Team1$ cat hello.txt
 it is to work with add , modify , delete , replace a line using lineinfile.
 cumith306@Hello-World:~/Team1$ vi ex.yml
cumith306@Hello-World:~/Team1$ cat ex.yml
      - name: Examples of lineinfile
             hosts: localhost
                          - name: "Insert after a matching line using insertafter parameter"
                               become user: sumith306
                                 tags: example5
                            lags: examples
lineinfile:
  path: /home/sumith306/Team1/hello.txt
  insertafter: "^This is Team 1."
  line: "We are from Team 1"
                                       firstmatch: yes
                                       state: present
                             register: example5out
sumith306@Hello-World:-/Team1$ echo "Lets try to insert a file after line having string Team 1"
Lets try to insert a file after line having string Team 1
sumith306@Hello-World:∼/Team1$ ansible-playbook ex.yml --ask-vault-pass
                   Password:

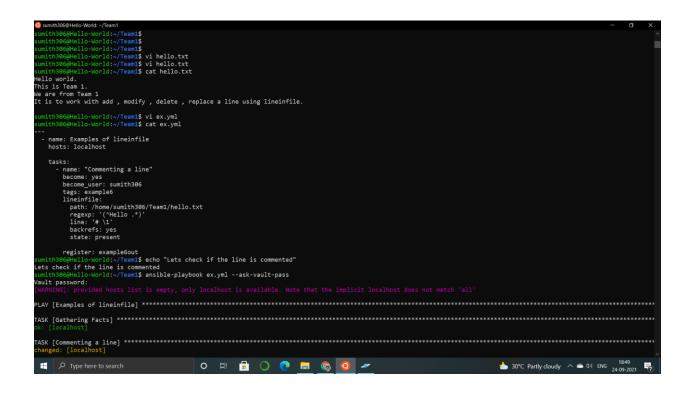
| Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Password: | Pass
```

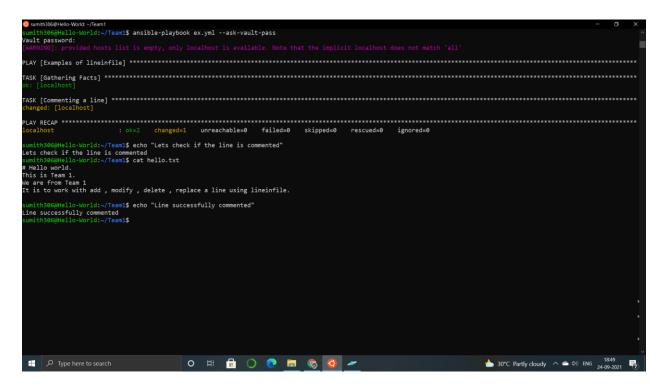


We can notice that there is no REGEX parameter. We do not need this as the insertafter parameter itself can have the Search String with the Regular Expression Syntax.

<u>Note</u>: If multiple entries of searching line is present then the last matched line would be considered.

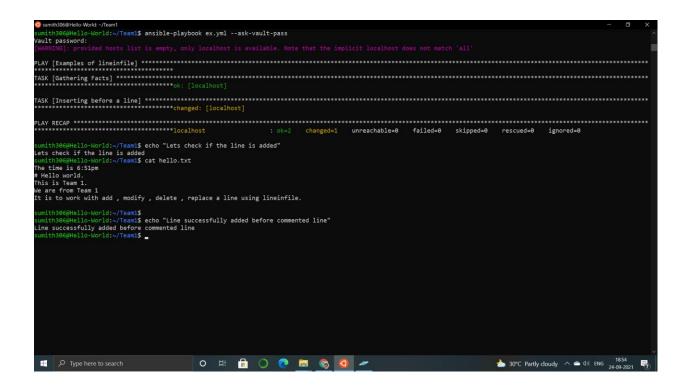
6: Commenting a line





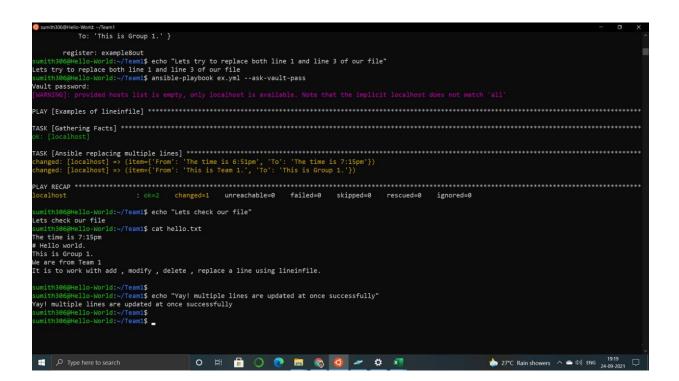
7: Insert Before a matching line using insertbefore parameter

```
### Description of the properties of the propert
```



We can notice that there is no REGEX parameter. We do not need this as the insertafter parameter itself can have the Search String with the Regular Expression Syntax.

8: Replacing multiple lines at once using lineinfile module



We have used with_items iteration/loop statement to pass multiple values to the regex and line parameters at the same time.

REFERENCES

- https://www.middlewareinventory.com/blog/ansible-lineinfile-examples/
- https://www.middlewareinventory.com/blog/ansible-lineinfile-multiple-lines-replace-multiple-lines/