



INDIAN INSTITUTE OF
INFORMATION
TECHNOLOGY

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.

```
sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ vi hello.txt
sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
Task 4
The Assignment is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$ vi ex1.yml
sumith306@Hello-World:~/Team1$ cat ex1.yml
---
- name: Examples of lineinfile
  hosts: localhost

  tasks:
    - name: "Validate if a String or line is present in the file"
      become: yes
      become_user: root
      tags: example1
      lineinfile:
        path: /home/sumith306/Team1/hello.txt
        line: "Task 4"
        state: present
        backup: yes
        check_mode: yes
        register: example1out

sumith306@Hello-World:~/Team1$ echo "Lets validate if a string or line with word Task is present in the file and fetch the line"
Lets validate if a string or line with word Task is present in the file and fetch the line
sumith306@Hello-World:~/Team1$ ansible localhost -m shell -a "grep -i Task /home/sumith306/Team1/hello.txt" -i ansible_hosts
[WARNING]: Unable to parse /home/sumith306/Team1/ansible_hosts as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
localhost | CHANGED | rc=0 >>
Task 4
sumith306@Hello-World:~/Team1$
```

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

```
sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ vi hello.txt
sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
Task 4
The Assignment is to work with add , modify , delete , replace a line using lineinfile.
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ vi ex.yml
sumith306@Hello-World:~/Team1$ cat ex.yml
---
- name: Examples of lineinfile
  hosts: localhost

  tasks:
    - name: "Validate if a String or line is present in the file and add if it does not exist"
      become: yes
      become_user: sumith306
      tags: example2
      lineinfile:
        path: /home/sumith306/Team1/hello.txt
        line: "Task 4 is being implemented now."
        state: present
        backup: yes
        register: example2out
sumith306@Hello-World:~/Team1$ echo "Lets validate if a string or line is present in the file and add if it doesnot exist"
Lets validate if a string or line is present in the file and add if it doesnot exist
sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Validate if a String or line is present in the file and add if it does not exist] *****
changed: [localhost]

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

sumith306@Hello-World:~/Team1$ echo "Lets check hello.txt"
Lets check hello.txt
sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
Task 4
The Assignment is to work with add , modify , delete , replace a line using lineinfile.
Task 4 is being implemented now.
sumith306@Hello-World:~/Team1$ echo "Yay! the line is added to our file"
Yay! the line is added to our file
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
```

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

```
Select sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ vi hello.txt
sumith306@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 is being implemented now.
sumith306@Hello-World:~/Team1$ vi ex.yml
sumith306@Hello-World:~/Team1$ cat ex.yml
---
- name: Examples of lineinfile
  hosts: localhost

  tasks:
    - name: "Replace a line with ansible lineinfile"
      become: yes
      become_user: sumith306
      tags: example3
      lineinfile:
        path: /home/sumith306/Team1/hello.txt
        #The string to search:
        regexp: "Task 4"
        #The string to replace
        line: "Assignment 2"
        state: present
        backup: yes
        register: example3out
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ echo "Lets try replacing a line"
Lets try replacing a line
sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****
```

```
Sumith306@Hello-World:~/Team1
Sumith306@Hello-World:~/Team1$
Sumith306@Hello-World:~/Team1$ echo "Lets try replacing a line"
Lets try replacing a line
Sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Replace a line with ansible lineinfile] *****
changed: [localhost]

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

Sumith306@Hello-World:~/Team1$ echo "Lets check if the lines are replaced"
Lets check if the lines are replaced
Sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
Task 4
Assignment 2
Sumith306@Hello-World:~/Team1$ echo "You can see that only the last matching line is replaced using lineinfile"
You can see that only the last matching line is replaced using lineinfile
Sumith306@Hello-World:~/Team1$
```

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

```
sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$ vi hello.txt
sumith306@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
sumith306@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
```

```
sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Remove a line from the file if it is found] *****
changed: [localhost]

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

sumith306@Hello-World:~/Team1$ echo "Lets check if the lines are removed"
Lets check if the lines are removed
sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
It is to work with add , modify , delete , replace a line using lineinfile.
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ echo "You can see that all the lines with string Task 4 are removed"
You can see that all the lines with string Task 4 are removed
sumith306@Hello-World:~/Team1$
```

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

```
sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ vi hello.txt
sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$ vi ex.yml
sumith306@Hello-World:~/Team1$ cat ex.yml
---
- name: Examples of lineinfile
  hosts: localhost

  tasks:
    - name: "Insert after a matching line using insertafter parameter"
      become: yes
      become_user: sumith306
      tags: example5
      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
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
TASK [Gathering Facts] *****
*****
PLAY [Examples of lineinfile] *****
TASK [Gathering Facts] *****
ok: [localhost]
TASK [Insert after a matching line using insertafter parameter] *****
changed: [localhost]
PLAY RECAP *****
localhost                : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

sumith306@Hello-World:~/Team1$ echo "Lets check the file"
Lets check the file
sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
We are from Team 1
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$ echo "We have inserted the line right below the matched line in contrast with add feature which adds at the end"
We have inserted the line right below the matched line in contrast with add feature which adds at the end
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
```

```
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
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
PLAY [Examples of lineinfile] *****
TASK [Gathering Facts] *****
ok: [localhost]
TASK [Insert after a matching line using insertafter parameter] *****
changed: [localhost]
PLAY RECAP *****
localhost                : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

sumith306@Hello-World:~/Team1$ echo "Lets check the file"
Lets check the file
sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
We are from Team 1
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$ echo "We have inserted the line right below the matched line in contrast with add feature which adds at the end"
We have inserted the line right below the matched line in contrast with add feature which adds at the end
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
```

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.

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

6: Commenting a line

```
sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ vi hello.txt
sumith306@Hello-World:~/Team1$ vi hello.txt
sumith306@Hello-World:~/Team1$ cat hello.txt
Hello world.
This is Team 1.
We are from Team 1
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$ vi ex.yml
sumith306@Hello-World:~/Team1$ cat ex.yml
---
- name: Examples of lineinfile
  hosts: localhost

  tasks:
    - name: "Commenting a line"
      become: yes
      become_user: sumith306
      tags: Example6
      lineinfile:
        path: /home/sumith306/Team1/hello.txt
        regexp: '^Hello .*'
        line: '# \1'
        backrefs: yes
        state: present

      register: example6out
sumith306@Hello-World:~/Team1$ echo "Lets check if the line is commented"
Lets check if the line is commented
sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Commenting a line] *****
changed: [localhost]
```

```
sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Commenting a line] *****
changed: [localhost]

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

sumith306@Hello-World:~/Team1$ echo "Lets check if the line is commented"
Lets check if the line is commented
sumith306@Hello-World:~/Team1$ cat hello.txt
# Hello world.
This is Team 1.
We are from Team 1
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$ echo "Line successfully commented"
Line successfully commented
sumith306@Hello-World:~/Team1$
```

7: Insert Before a matching line using insertbefore parameter

```
sumith306@Hello-World: ~/Team1
line successfully commented
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ cat hello.txt
# Hello world.
This is Team 1.
We are from Team 1
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$ vi ex.yml
sumith306@Hello-World:~/Team1$ cat ex.yml
---
- name: Examples of lineinfile
  hosts: localhost

  tasks:
    - name: "Inserting before a line"
      become: yes
      become_user: sumith306
      tags: example7
      lineinfile:
        path: /home/sumith306/Team1/hello.txt
        insertbefore: '# Hello .*'
        line: "The time is 6:51pm"
        backrefs: yes
        state: present

      register: example7out
sumith306@Hello-World:~/Team1$ echo "Lets insert a line before the commented line"
Lets insert a line before the commented line
sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Inserting before a line] *****
```

```
sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Inserting before a line] *****
changed: [localhost]

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

sumith306@Hello-World:~/Team1$ echo "Lets check if the line is added"
Lets check if the line is added
sumith306@Hello-World:~/Team1$ cat hello.txt
# Hello world.
The time is 6:51pm
This is Team 1.
We are from Team 1
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ echo "Line successfully added before commented line"
Line successfully added before commented line
sumith306@Hello-World:~/Team1$
```

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

```
sumith306@Hello-World: ~/Team1
sumith306@Hello-World:~/Team1$ echo "Now lets try replacing multiple lines in our file"
Now lets try replacing multiple lines in our file
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ cat hello.txt
The time is 6:51pm
# Hello world.
This is Team 1.
We are from Team 1
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$ vi ex.yml
sumith306@Hello-World:~/Team1$ cat ex.yml
---
- name: Examples of lineinfile
  hosts: localhost

  tasks:
    - name: "Ansible replacing multiple lines"
      become: yes
      become_user: sumith306
      tags: example8
      lineinfile:
        path: /home/sumith306/Team1/hello.txt
        # Line to search/match against
        regexp: '{{item.From}}'
        # Line to replace with
        line: '{{item.To}}'
        state: present

      with_items:
        - { From: 'The time is 6:51pm' ,
          To: 'The time is 7:15pm' }
        - { From: 'This is Team 1.' ,
          To: 'This is Group 1.' }

      register: example8out
sumith306@Hello-World:~/Team1$ echo "Lets try to replace both line 1 and line 3 of our file"
Lets try to replace both line 1 and line 3 of our file
```

```
sumith306@Hello-World: ~/Team1
To: 'This is Group 1.' }

register: example8out
sumith306@Hello-World:~/Team1$ echo "Lets try to replace both line 1 and line 3 of our file"
Lets try to replace both line 1 and line 3 of our file
sumith306@Hello-World:~/Team1$ ansible-playbook ex.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Examples of lineinfile] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Ansible replacing multiple lines] *****
changed: [localhost] => (item={'From': 'The time is 6:51pm', 'To': 'The time is 7:15pm'})
changed: [localhost] => (item={'From': 'This is Team 1.', 'To': 'This is Group 1.'})

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

sumith306@Hello-World:~/Team1$ echo "Lets check our file"
Lets check our file
sumith306@Hello-World:~/Team1$ cat hello.txt
The time is 7:15pm
# Hello world.
This is Group 1.
We are from Team 1
It is to work with add , modify , delete , replace a line using lineinfile.

sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$ echo "Yay! multiple lines are updated at once successfully"
Yay! multiple lines are updated at once successfully
sumith306@Hello-World:~/Team1$
sumith306@Hello-World:~/Team1$
```

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/>