ANSIBLE DYNAMIC ASSIGNMENTS (INCLUDE) AND COMMUNITY ROLES

Last 2 projects have already equipped us with some knowledge and skills on Ansible, so we can perform configurations using ***playbooks***, ***roles*** and ***imports***. Now we will continue configuring our UAT servers, learning and practicing new Ansible concepts and modules.

In this project we will introduce ***dynamic assignments*** by using ***include*** module.

Now you may be wondering, what is the difference between **static** and **dynamic** assignments?

Well, from ***Project 12***, you can already tell that static assignments use ***import*** Ansible module. The module that enables dynamic assignments is ***include***.

Hence,

```
import = Static
include = Dynamic
```

When the **import** module is used, all statements are pre-processed at the time playbooks are ***parsed***. Meaning, when you execute ***site.yml*** playbook, Ansible will process all the playbooks referenced during the time it is parsing the statements. This also means that, during actual execution, if any statement changes, such statements will not be considered. Hence, it is static.

On the other hand, when **include** module is used, all statements are processed only during execution of the playbook. Meaning, after the statements are **parsed**, any changes to the statements encountered during execution will be used.

INTRODUCING DYNAMIC ASSIGNMENT INTO OUR STRUCTURE

In your **https://github.com/<your-name>/ansible-config-mgt** GitHub repository start a new branch and call it **dynamic-assignments**.

Create a new folder, name it **dynamic-assignments**. Then inside this folder, create a new file and name it ***env-vars.yml***

```
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ git checkout dynamic-assignments
Branch 'dynamic-assignments' set up to track remote branch 'dynamic-assignments' from 'origin'.
Switched to a new branch 'dynamic-assignments'
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ git branch
* dynamic-assignments
  feature/proj-45
  main
  refactor
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ mkdir dynamic-assignments
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ ls
Ansible.md  dynamic-assignments  inventory  playbooks  roles  static-assignments
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ cd dynamic-assignments/
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$ sudo vi env-vars.yml
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$ ls
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$ sudo touch env-vars.yml
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$ ls
env-vars.yml
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$
```

Note: Depending on what method you used in the previous project you may have or not have roles folder in your GitHub repository – if you used ansible-galaxy, then roles directory was only created on your Jenkins-Ansible server locally. It is recommended to have all the codes managed and tracked in GitHub, so you might want to recreate this structure manually in this case – it is up to you.

Our GitHub should have following structure by now.

```
├── dynamic-assignments
│   └── env-vars.yml
├── inventory
│   └── dev
    └── stage
    └── uat
    └── prod
└── playbooks
    └── site.yml
└── roles (optional folder)
    └──defaults
    └── handlers
    └── meta
    └── tasks
    └── templates
    └── README.md
└── static-assignments
    └── common-del.yml
    └── common.yml
    └── uat-webservers.yml
```

Since we will be using the same Ansible to configure multiple environments, and each of these environments will have certain unique attributes, such as

servername, ip-address etc., we will need a way to set values to variables per specific environment.

For this reason, we will now create a folder to keep each environment's variables file. Therefore, create a new folder env-vars, then for each environment, create new **YAML** files which we will use to set variables.

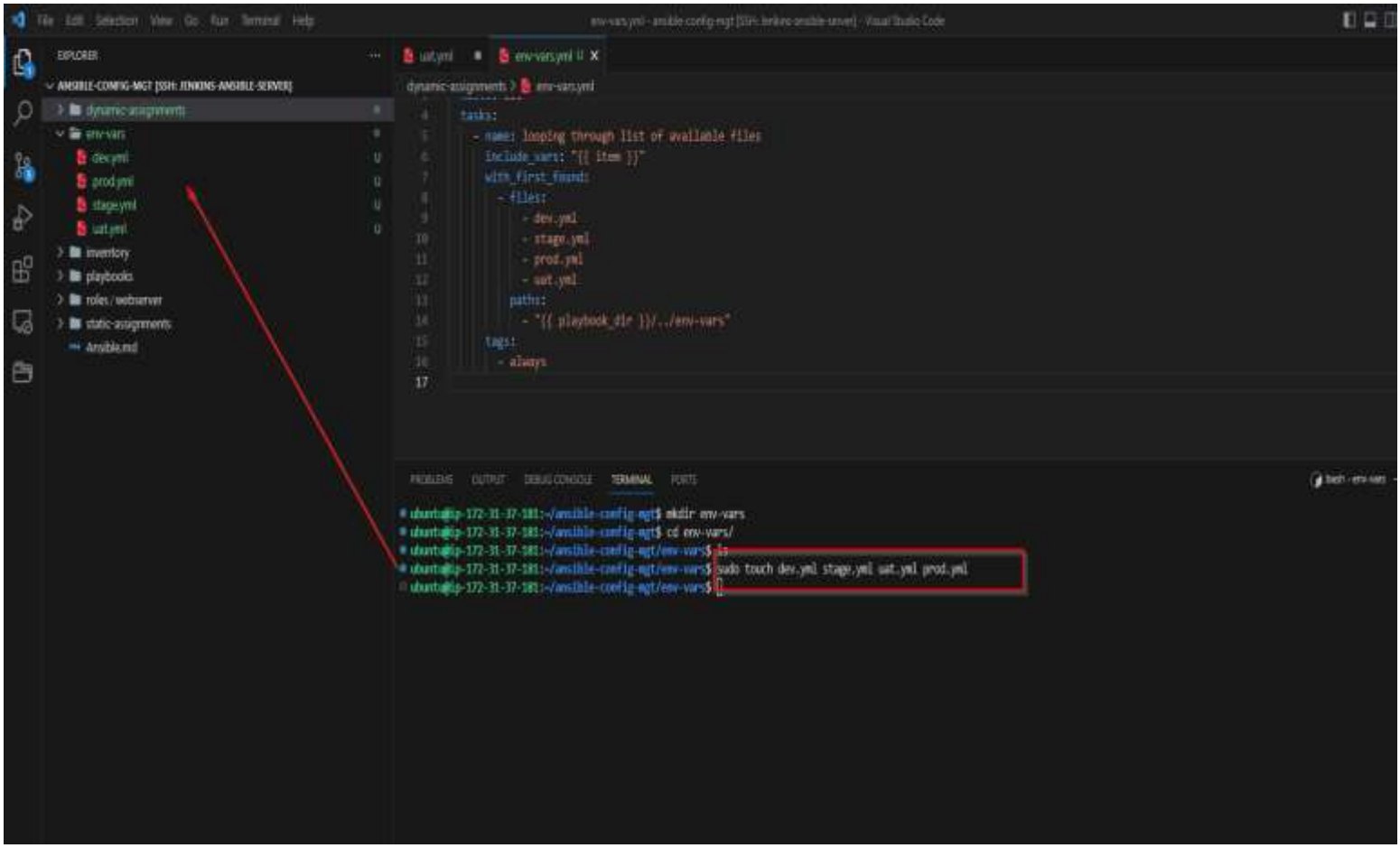
$ cd ansible-config-mgt

$ sudo mkdir env-vars

$ sudo touch dev.yml stage.yml uat.yml prod.yml

Our layout should now look like this.

```
├── dynamic-assignments
│   └── env-vars.yml
├── env-vars
    └── dev.yml
    └── stage.yml
    └── uat.yml
    └── prod.yml
├── inventory
    └── dev
    └── stage
    └── uat
    └── prod
├── playbooks
    └── site.yml
└── static-assignments
    └── common.yml
    └── webservers.yml
```



Now paste the instruction below into the ***env-vars.yml*** file.

```
---
- name: collate variables from env specific file, if it exists
  hosts: all
  tasks:
    - name: looping through list of available files
      include_vars: "{{ item }}"
      with_first_found:
        - files:
            - dev.yml
            - stage.yml
            - prod.yml
            - uat.yml
          paths:
            - "{{ playbook_dir }}/../env-vars"
      tags:
        - always
```

```
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$ sudo vi env-vars.yml
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$ ls
env-vars.yml
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$ cat env-vars.yml
---
- name: collate variables from env specific file, if it exists
  hosts: all
  tasks:
    - name: looping through list of available files
      include_vars: "{{ item }}"
      with_first_found:
        - files:
            - dev.yml
            - stage.yml
            - prod.yml
            - uat.yml
          paths:
            - "{{ playbook_dir }}/../env-vars"
      tags:
        - always
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/dynamic-assignments$
```

Notice 3 things to notice here:

1. We used `include_vars` syntax instead of `include`, this is because Ansible developers decided to separate different features of the module. From Ansible version **2.8**, the `include` module is deprecated and variants of `include_*` must be used. These are:

- include_role
- include_tasks
- include_vars

In the same version, variants of **import** were also introduces, such as:

- import_role
- import_tasks

2. We made use of a special variables `{ playbook_dir }` and `{ inventory_file }`. `{ playbook_dir }` will help Ansible to determine the location of the running playbook, and from there navigate to other path on the filesystem. `{ inventory_file }` on the other hand will dynamically resolve to the name of the inventory file being used, then append `.yml` so that it picks up the required file within the `env-vars` folder.
3. We are including the variables using a loop. `with_first_found` implies that, looping through the list of files, the first one found is used. This is good so that we can always set default values in case an environment specific env file does not exist.

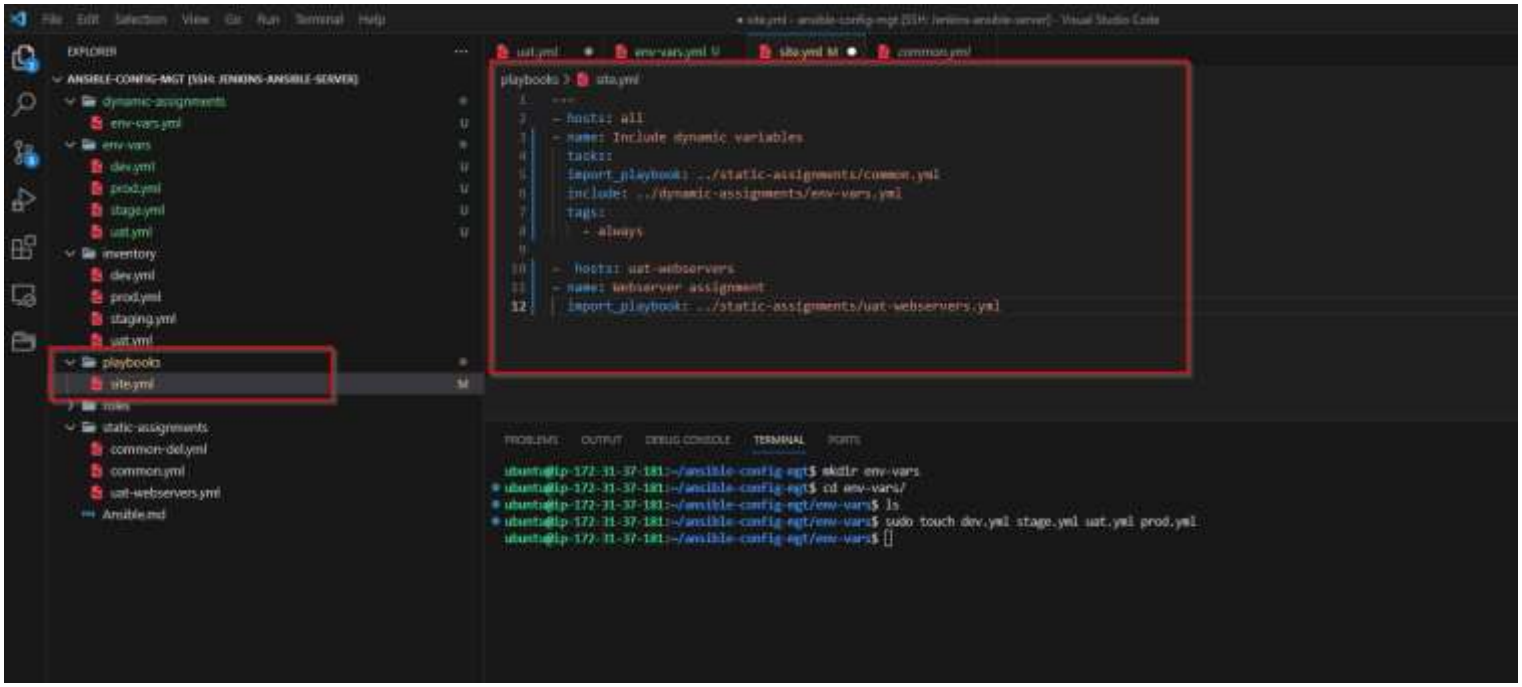
UPDATE SITE.YML WITH DYNAMIC ASSIGNMENTS

We will now update ***site.yml*** file to make use of the dynamic assignment. (*At this point, we cannot test it yet. We are just setting the stage for what is yet to come*)

Update the ***site.yml*** to look like below:

```
---
- hosts: all
- name: Include dynamic variables
  tasks:
  import_playbook: ../static-assignments/common.yml
  include: ../dynamic-assignments/env-vars.yml
  tags:
    - always

-  hosts: uat-webservers
- name: Webserver assignment
  import_playbook: ../static-assignments/uat-ebservers.yml
```



COMMUNITY ROLES

Before we proceed, let us commit and push new changes on **dynamic-assignment** branch to git.

```
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ git status
On branch dynamic-assignments
Your branch is up to date with 'origin/dynamic-assignments'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   playbooks/site.yml

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        dynamic-assignments/
        env-vars/

no changes added to commit (use "git add" and/or "git commit -a")
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ git add .
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ git commit -m "adding vars"
[dynamic-assignments ad78cf8] adding vars
 6 files changed, 25 insertions(+), 3 deletions(-)
 create mode 100644 dynamic-assignments/env-vars.yml
 create mode 100644 env-vars/dev.yml
 create mode 100644 env-vars/prod.yml
 create mode 100644 env-vars/stage.yml
 create mode 100644 env-vars/uat.yml
```

```
ubuntu@ip-172-31-37-181:~/ansible-config-mgt/env-vars$ git push
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (7/7), 988 bytes | 329.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To https://github.com/obaigbenaa/ansible-config-mgt.git
   df83a6e..ad78cf8  dynamic-assignments -> dynamic-assignments
```

Now it is time to create a role for MySQL database – it should install the MySQL package, create a database and configure users. But why should we re-invent the wheel? There are tons of roles that have already been developed by other open source engineers out there. These roles are actually production ready, and dynamic to accommodate most of Linux flavours. With Ansible Galaxy again, we can simply download a ready to use ansible role, and move on with other businesses.

Download Mysql Ansible Role

We can browse available community roles *here*

We will be using a *MySQL role developed by geerlingguy.*

Hint: To preserve your your GitHub in actual state after you install a new role – make a commit and push to master your **'ansible-config-mgt'** directory. Of course you must have **git** installed and configured on **Jenkins-Ansible** server and, for more

convenient work with codes, you can configure Visual Studio Code to work with this directory.

We will now create a new branch called **'roles-feature'** and move into it. This is where we will be creating the role for MySQL database.

$ git branch roles-feature

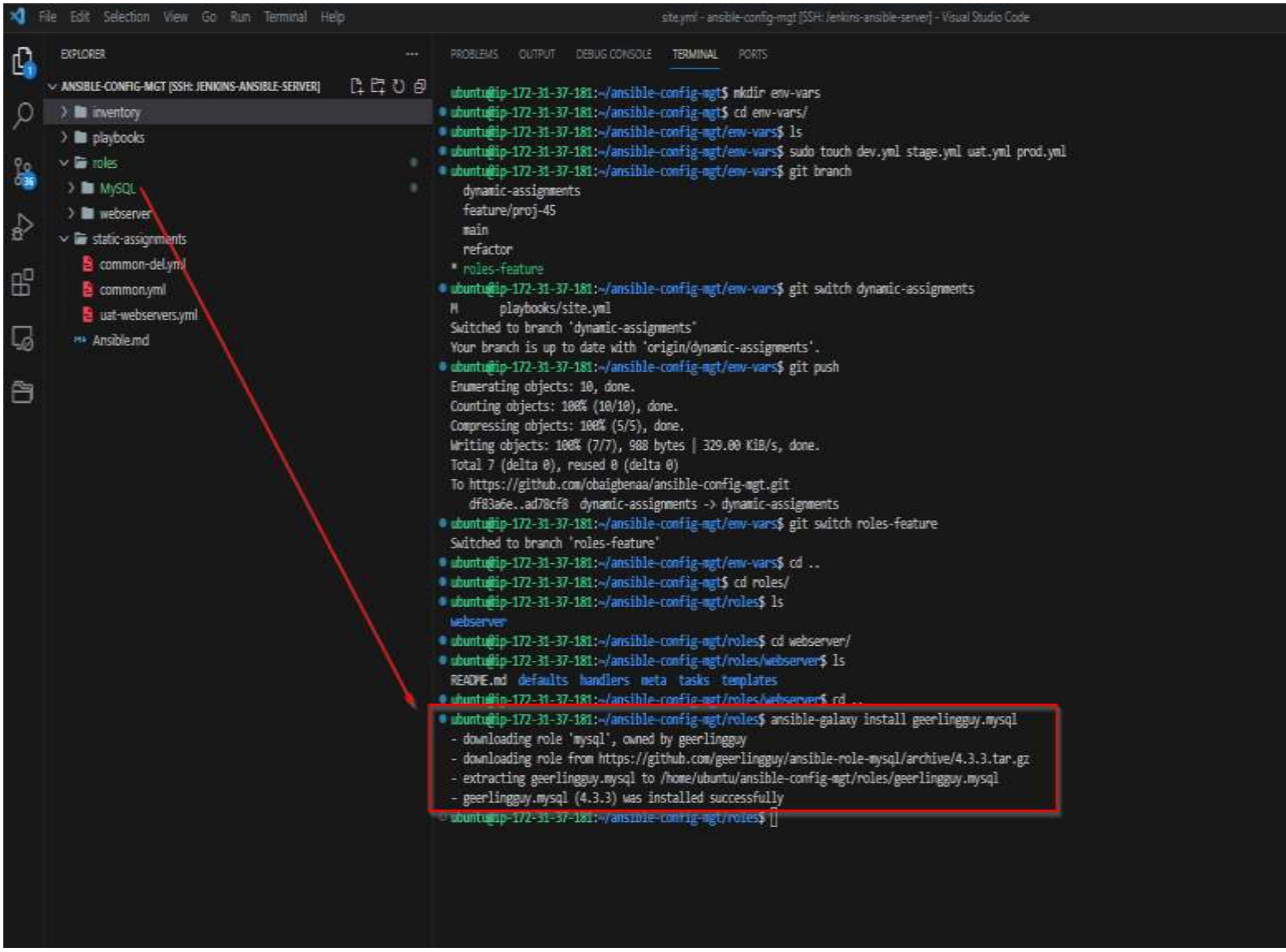
$ git switch roles-feature

Now we are in the **roles-feature** branch.

Inside ***roles*** directory we will now create a new **MySQL** role with **ansible-galaxy install geerlingguy.mysql** and rename the folder to ***mysql***

$ ansible-galaxy install geerlingguy.mysql

$ mv geerlingguy.mysql/ mysql



Read and check the README.md file for instructions and edit roles configuration to use correct credentials for MySQL required for the **tooling** website.

If you scroll to 'mysql_users: []' tab of the .md file, you will find the default setup of the configuration as below:

Update it with the correct credentials and privileges to want for username, host list, password, and database access.

```
97
98   # Logging settings.
99   mysql_log: ""
100  # The following variables have a default value depending on operating system.
101  # mysql_log_error: /var/log/mysql/mysql.err
102  # mysql_syslog_tag: mysql
103
104  mysql_config_include_files: []
105  #   - src: path/relative/to/playbook/file.cnf
106  #   - { src: path/relative/to/playbook/anotherfile.cnf, force: yes }
107
108  # Databases.
109  mysql_databases: []
110      - name: tooling
111        collation: utf8_general_ci
112        encoding: utf8
113        replicate: 1
114
115  # Users.
116  mysql_users: []
117      - name: webaccess
118        host: 0.0.0.0
119        password: password
120        priv: '*.*:ALL,GRANT'
121
122  mysql_disable_log_bin: false
123
124  # Replication settings (replication is only enabled if master/user have values).
125  mysql_server_id: "1"
126  mysql_max_binlog_size: "100M"
127  mysql_binlog_format: "ROW"
128  mysql_expire_logs_days: "10"
129  mysql_replication_role: ''
130  mysql_replication_master: ''
131  mysql_replication_master_inventory_host: "{{ mysql_replication_master }}"
```

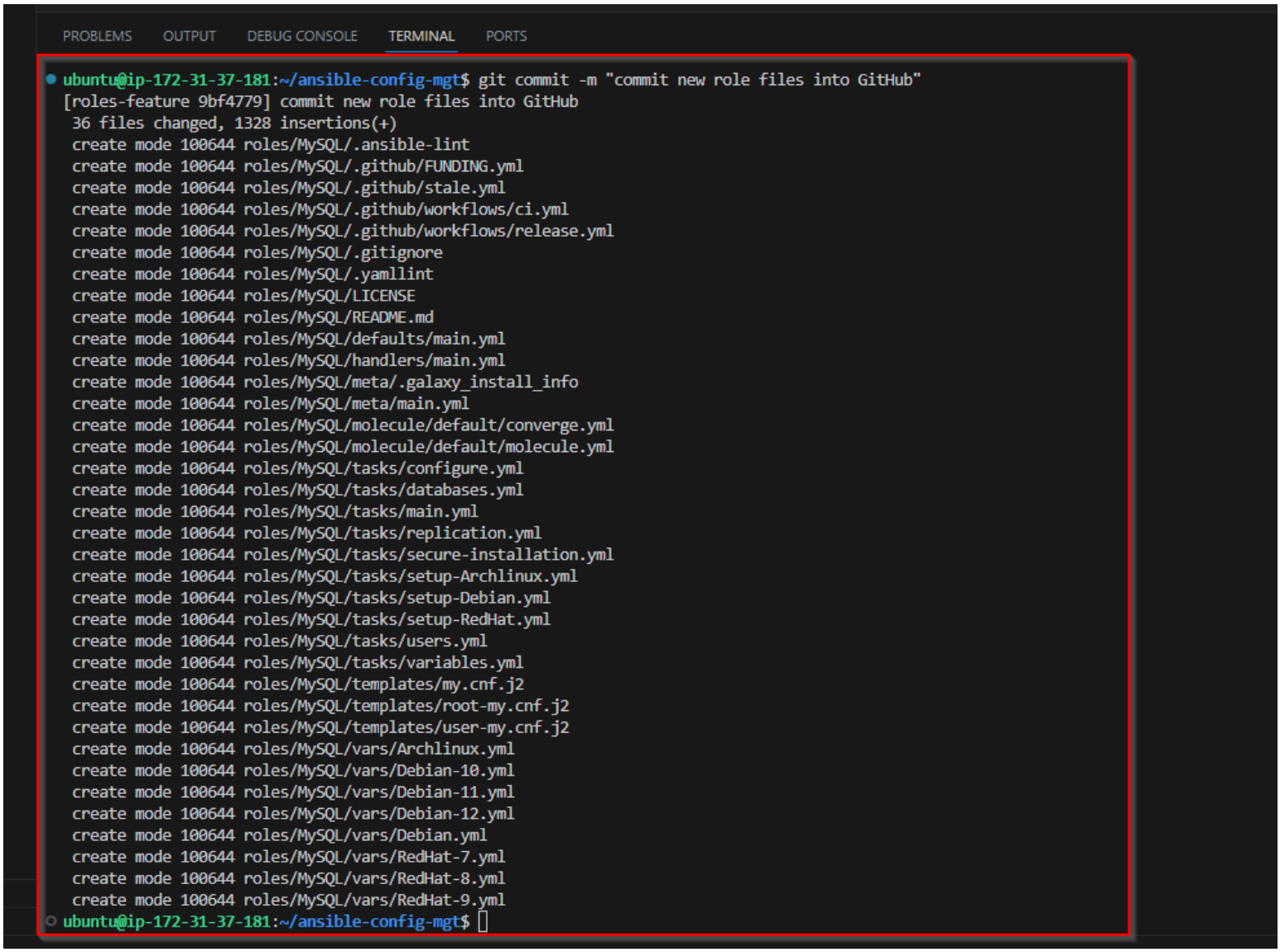
Save the updated configuration.

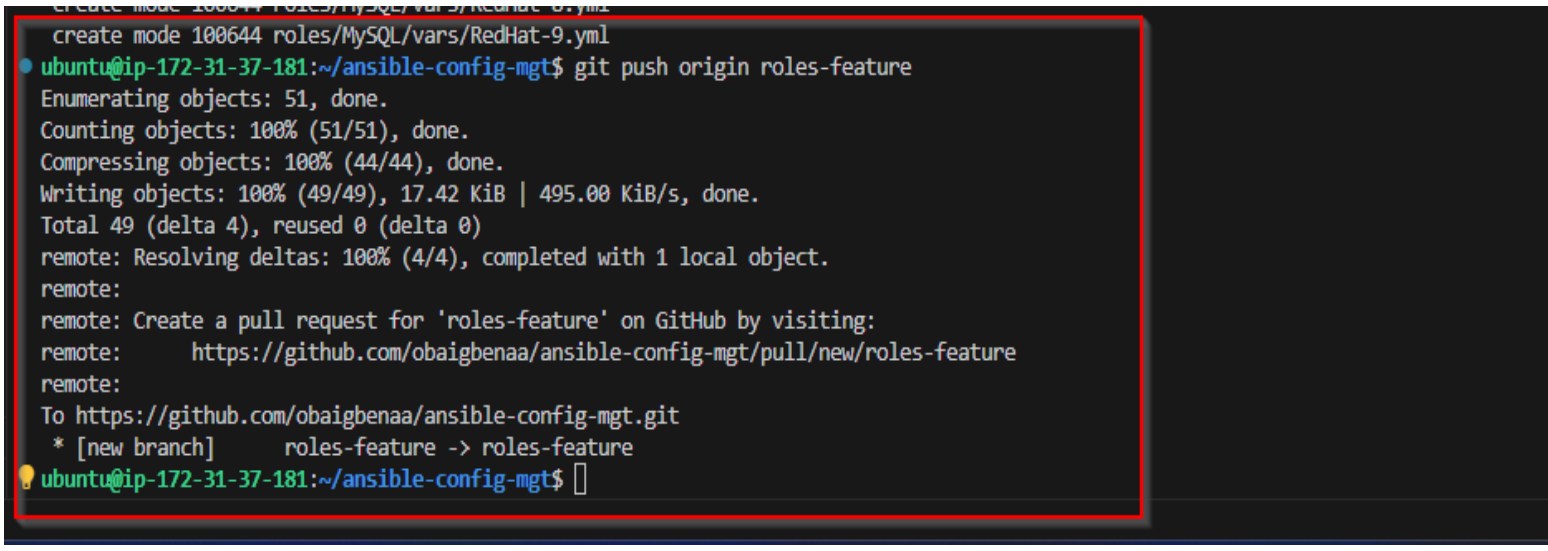
Now we will upload all code changes (on branches **roles-feature and dynamic-assignment**) into your GitHub:

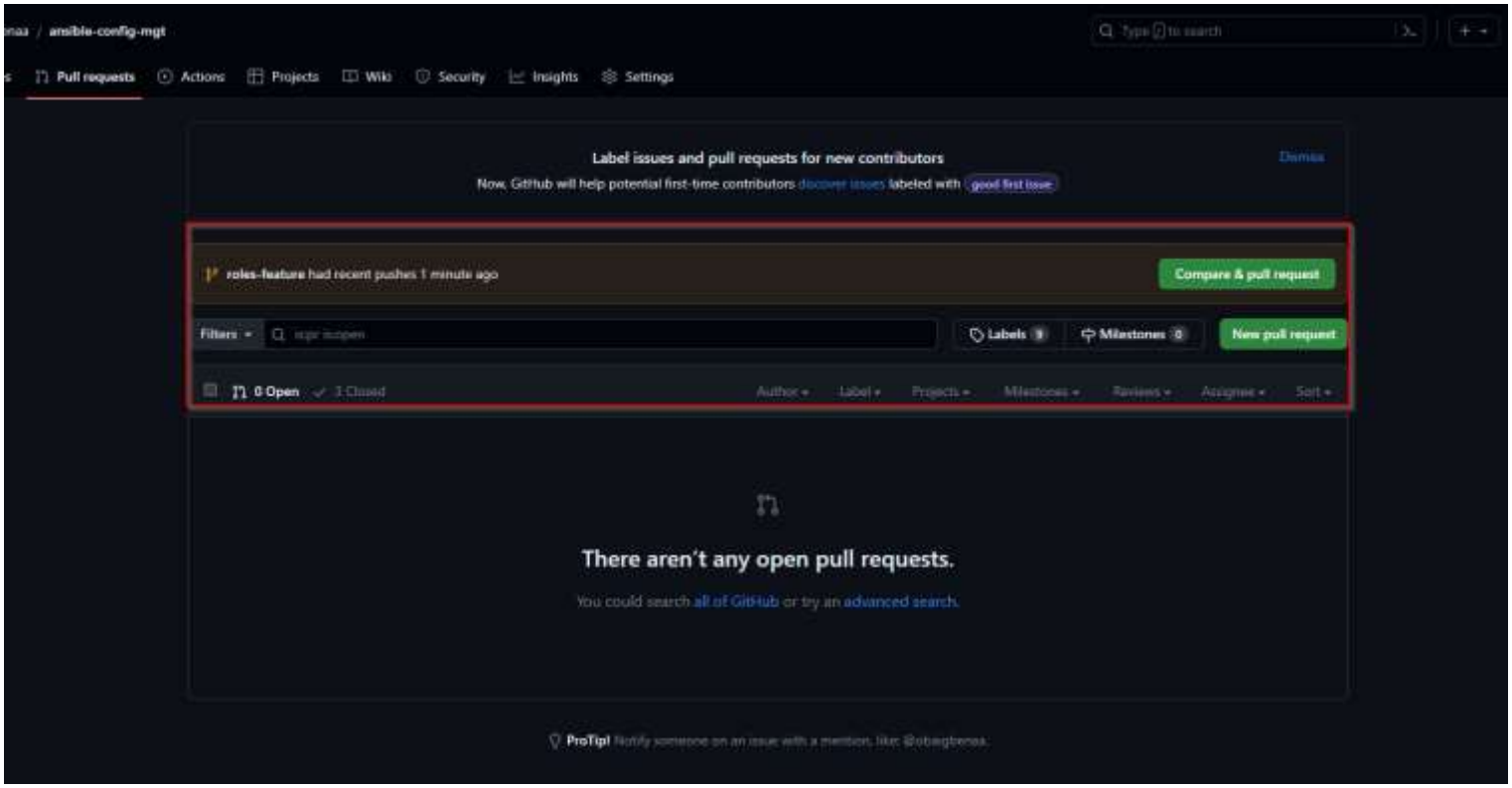
$ git add .

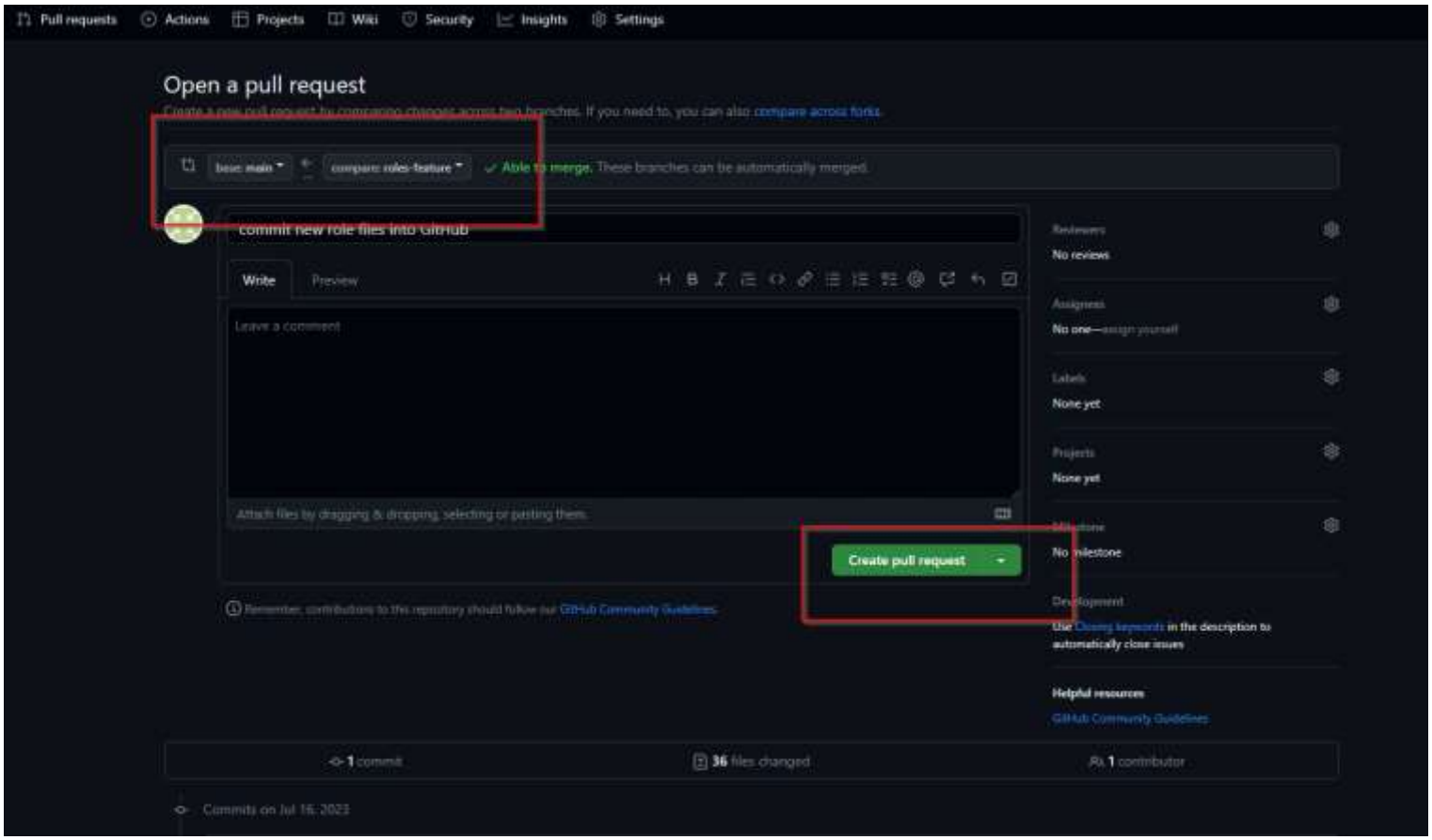
$ git commit -m "Commit new role files into GitHub"

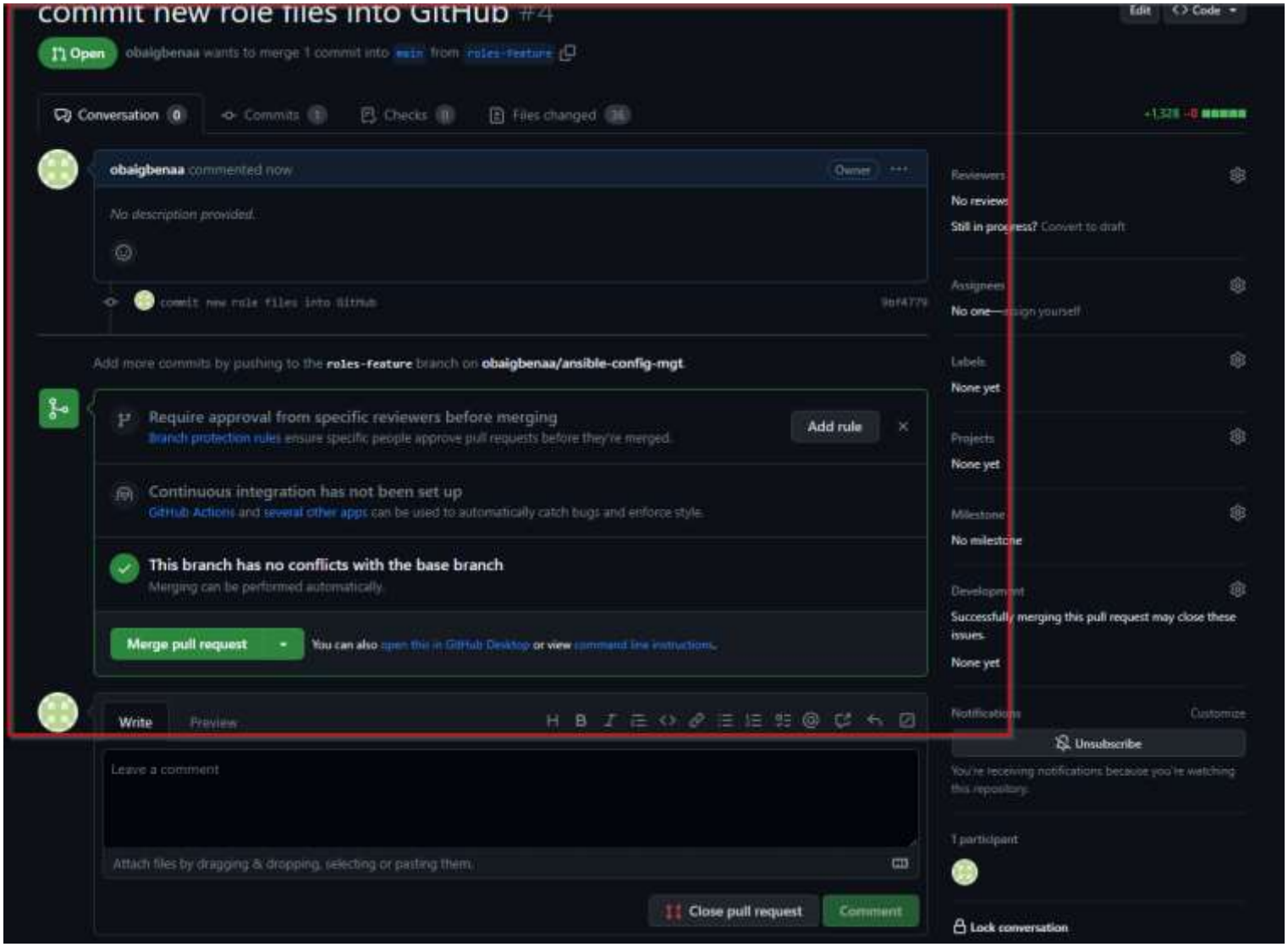
$ git push --set-upstream origin roles-feature

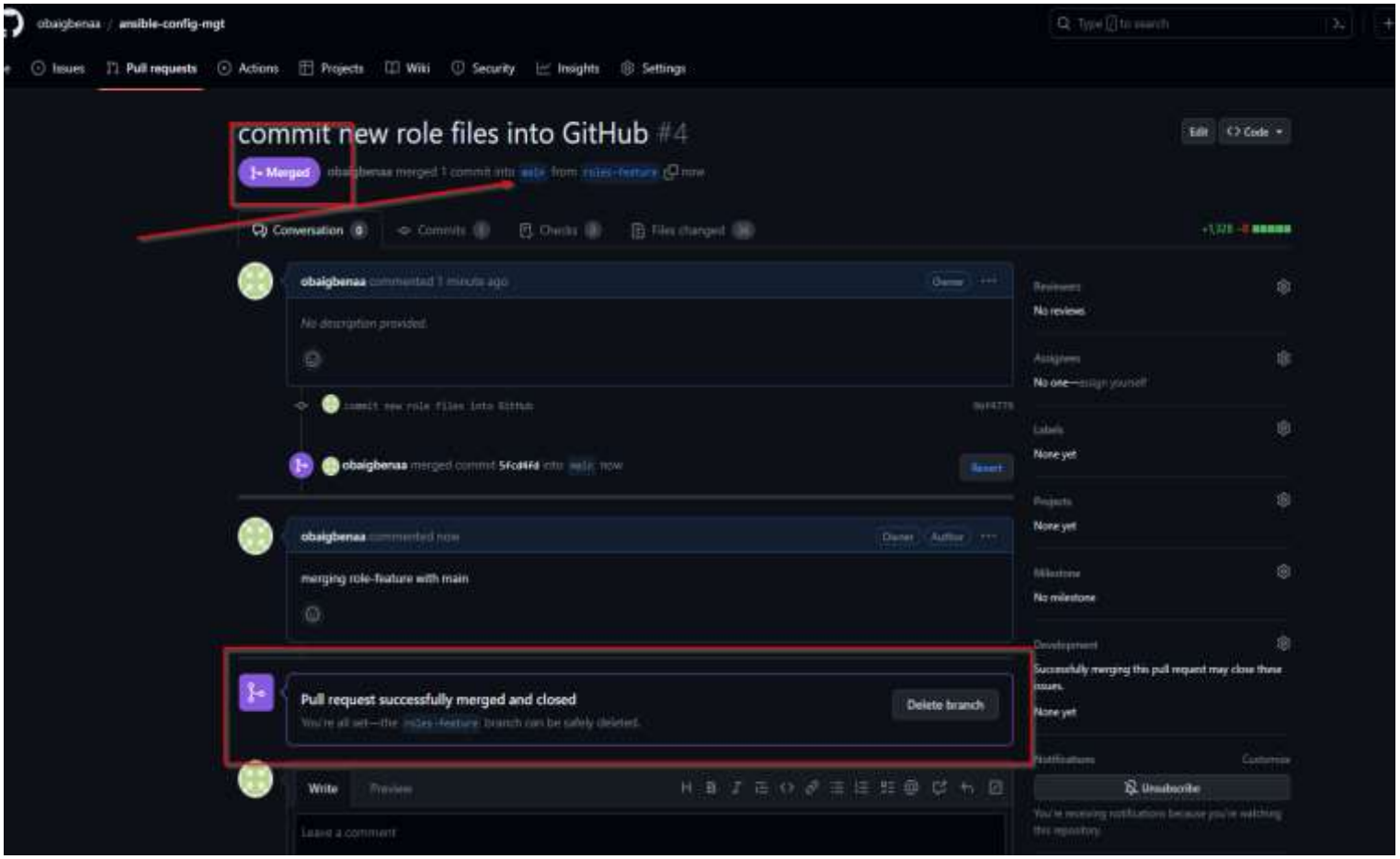

PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ git commit -m "commit new role files into GitHub"
[roles-feature 9bf4779] commit new role files into GitHub
36 files changed, 1328 insertions(+)
create mode 100644 roles/MySQL/.ansible-lint
create mode 100644 roles/MySQL/.github/FUNDING.yml
create mode 100644 roles/MySQL/.github/stale.yml
create mode 100644 roles/MySQL/.github/workflows/ci.yml
create mode 100644 roles/MySQL/.github/workflows/release.yml
create mode 100644 roles/MySQL/.gitignore
create mode 100644 roles/MySQL/.yamllint
create mode 100644 roles/MySQL/LICENSE
create mode 100644 roles/MySQL/README.md
create mode 100644 roles/MySQL/defaults/main.yml
create mode 100644 roles/MySQL/handlers/main.yml
create mode 100644 roles/MySQL/meta/.galaxy_install_info
create mode 100644 roles/MySQL/meta/main.yml
create mode 100644 roles/MySQL/molecule/default/converge.yml
create mode 100644 roles/MySQL/molecule/default/molecule.yml
create mode 100644 roles/MySQL/tasks/configure.yml
create mode 100644 roles/MySQL/tasks/databases.yml
create mode 100644 roles/MySQL/tasks/main.yml
create mode 100644 roles/MySQL/tasks/replication.yml
create mode 100644 roles/MySQL/tasks/secure-installation.yml
create mode 100644 roles/MySQL/tasks/setup-Archlinux.yml
create mode 100644 roles/MySQL/tasks/setup-Debian.yml
create mode 100644 roles/MySQL/tasks/setup-RedHat.yml
create mode 100644 roles/MySQL/tasks/users.yml
create mode 100644 roles/MySQL/tasks/variables.yml
create mode 100644 roles/MySQL/templates/my.cnf.j2
create mode 100644 roles/MySQL/templates/root-my.cnf.j2
create mode 100644 roles/MySQL/templates/user-my.cnf.j2
create mode 100644 roles/MySQL/vars/Archlinux.yml
create mode 100644 roles/MySQL/vars/Debian-10.yml
create mode 100644 roles/MySQL/vars/Debian-11.yml
create mode 100644 roles/MySQL/vars/Debian-12.yml
create mode 100644 roles/MySQL/vars/Debian.yml
create mode 100644 roles/MySQL/vars/RedHat-7.yml
create mode 100644 roles/MySQL/vars/RedHat-8.yml
create mode 100644 roles/MySQL/vars/RedHat-9.yml
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$

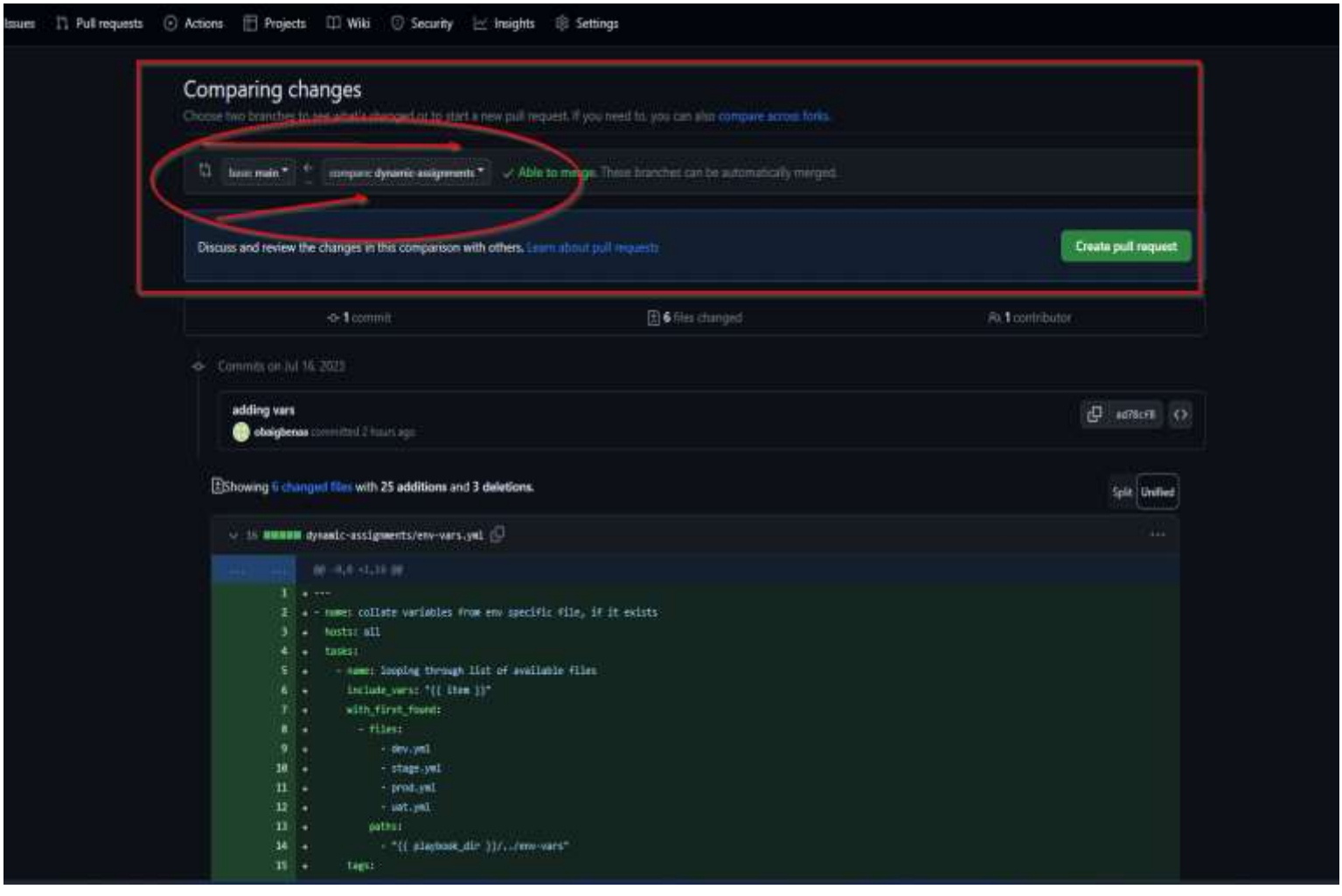

create mode 100644 roles/MySQL/vars/RedHat-9.yml
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$ git push origin roles-feature
Enumerating objects: 51, done.
Counting objects: 100% (51/51), done.
Compressing objects: 100% (44/44), done.
Writing objects: 100% (49/49), 17.42 KiB | 495.00 KiB/s, done.
Total 49 (delta 4), reused 0 (delta 0)
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
remote:
remote: Create a pull request for 'roles-feature' on GitHub by visiting:
remote: https://github.com/obaigbenaa/ansible-config-mgt/pull/new/roles-feature
remote:
To https://github.com/obaigbenaa/ansible-config-mgt.git
* [new branch] roles-feature -> roles-feature
ubuntu@ip-172-31-37-181:~/ansible-config-mgt$


ansible-config-mgt
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Label issues and pull requests for new contributors
Now, GitHub will help potential first-time contributors discover issues labeled with good first issue
Dismiss
roles-feature had recent pushes 1 minute ago
Compare & pull request
Filters
Labels 9
Milestones 0
New pull request
0 Open
3 Closed
Author
Label
Projects
Milestones
Reviews
Assignee
Sort
There aren't any open pull requests.
You could search all of GitHub or try an advanced search.
ProTip! Notify someone on an issue with a mention, like: @obaigbenaa

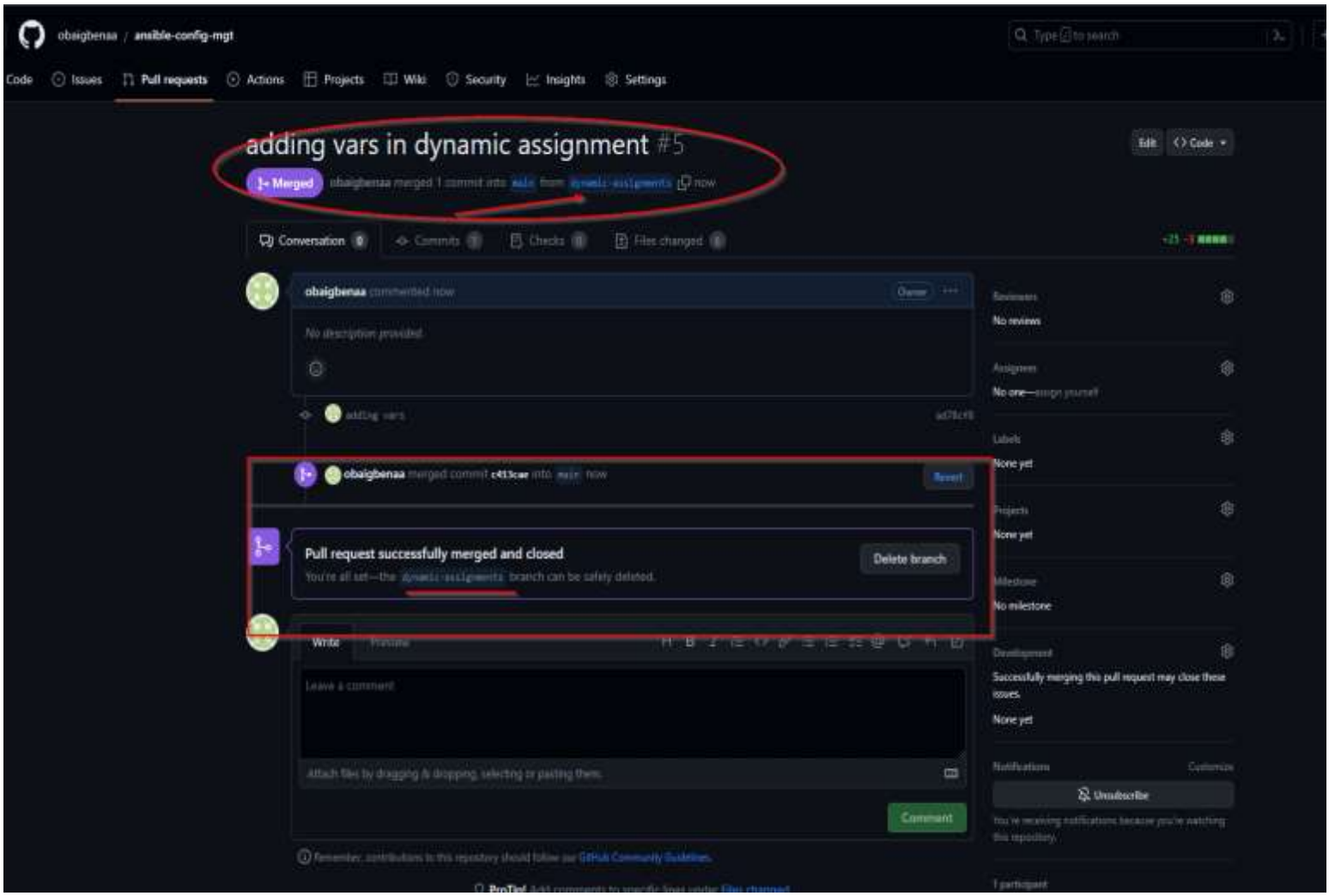

Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Open a pull request
base: main
compare: roles-feature
Able to merge. These branches can be automatically merged.
commit new role files into GitHub
Write
Preview
Leave a comment
Attach files by dragging & dropping, selecting or pasting them.
Create pull request
Reviewers
No reviews
Assignees
No one—assign yourself
Labels
None yet
Projects
None yet
No milestone
Development
Use Closing keywords in the description to automatically close issues
Helpful resources
GitHub Community Guidelines
1 commit
36 files changed
1 contributor
Commits on Jul 16, 2023


commit new role files into GitHub #4
Open
obaigbenaa wants to merge 1 commit into main from roles-feature
Conversation 0
Commits 1
Checks 11
Files changed 14
obaigbenaa commented now
Owner
No description provided.
commit new role files into GitHub
Add more commits by pushing to the roles-feature branch on obaigbenaa/ansible-config-mgt.
Require approval from specific reviewers before merging
Branch protection rules ensure specific people approve pull requests before they're merged.
Add rule
Continuous integration has not been set up
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.
This branch has no conflicts with the base branch
Merging can be performed automatically.
Merge pull request
You can also open this in GitHub Desktop or view command line instructions.
Write
Preview
Leave a comment
Attach files by dragging & dropping, selecting or pasting them.
Close pull request
Comment
Reviewers
No reviews
Still in progress? Convert to draft
Assignees
No one—assign yourself
Labels
None yet
Projects
None yet
Milestone
No milestone
Development
Successfully merging this pull request may close these issues.
None yet
Notifications
Customize
Unsubscribe
You're receiving notifications because you're watching this repository.
1 participant
Lock conversation

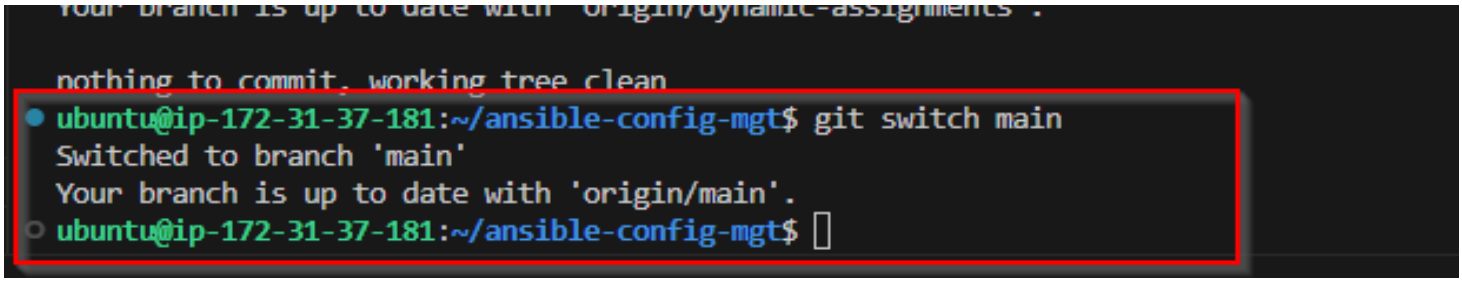

obaigbenaa / ansible-config-mgt
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
commit new role files into GitHub #4
Merged
Edit
Code
Conversation
Commits
Checks
Files changed
obaigbenaa commented 1 minute ago
No description provided.
commit new role files into GitHub
obaigbenaa merged commit 5fc4ff4 into main now
Revert
obaigbenaa commented now
merging role-feature with main
Pull request successfully merged and closed
You're all set—the role-feature branch can be safely deleted.
Delete branch
Write
Preview
Leave a comment
Reviewers
No reviews
Assignees
No one—assign yourself
Labels
None yet
Projects
None yet
Milestone
No milestone
Development
Successfully merging this pull request may close these issues.
None yet
Notifications
Customize
Unsubscribe
You're receiving notifications because you're watching this repository.


Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks.
base: main
compare: dynamic-assignments
Able to merge. These branches can be automatically merged.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request
1 commit
6 files changed
1 contributor
Commits on Jul 16, 2023
adding vars
obaigbenaa committed 2 hours ago
ad78cf8
Showing 6 changed files with 25 additions and 3 deletions.
Split
Unified
15 dynamic-assignments/env-vars.yml
@@ -0,0 +1,15 @@

- name: collate variables from env specific file, if it exists
hosts: all
tasks:
- name: looping through list of available files
include_vars: "{{ item }}"
with_first_found:
- files:
- env.yml
- stage.yml
- prod.yml
- uat.yml
paths:
- "{{ playbook_dir }}/../env-vars"
tags:



$ git switch main



LOAD BALANCER ROLES

We want to be able to choose which Load Balancer to use, Nginx or Apache, so we need to have two roles respectively:

1. Nginx
2. Apache

With your experience on Ansible so far you can:

- Decide if you want to develop your own roles, or find available ones from the community.

#I find it faster to develop roles from the community.

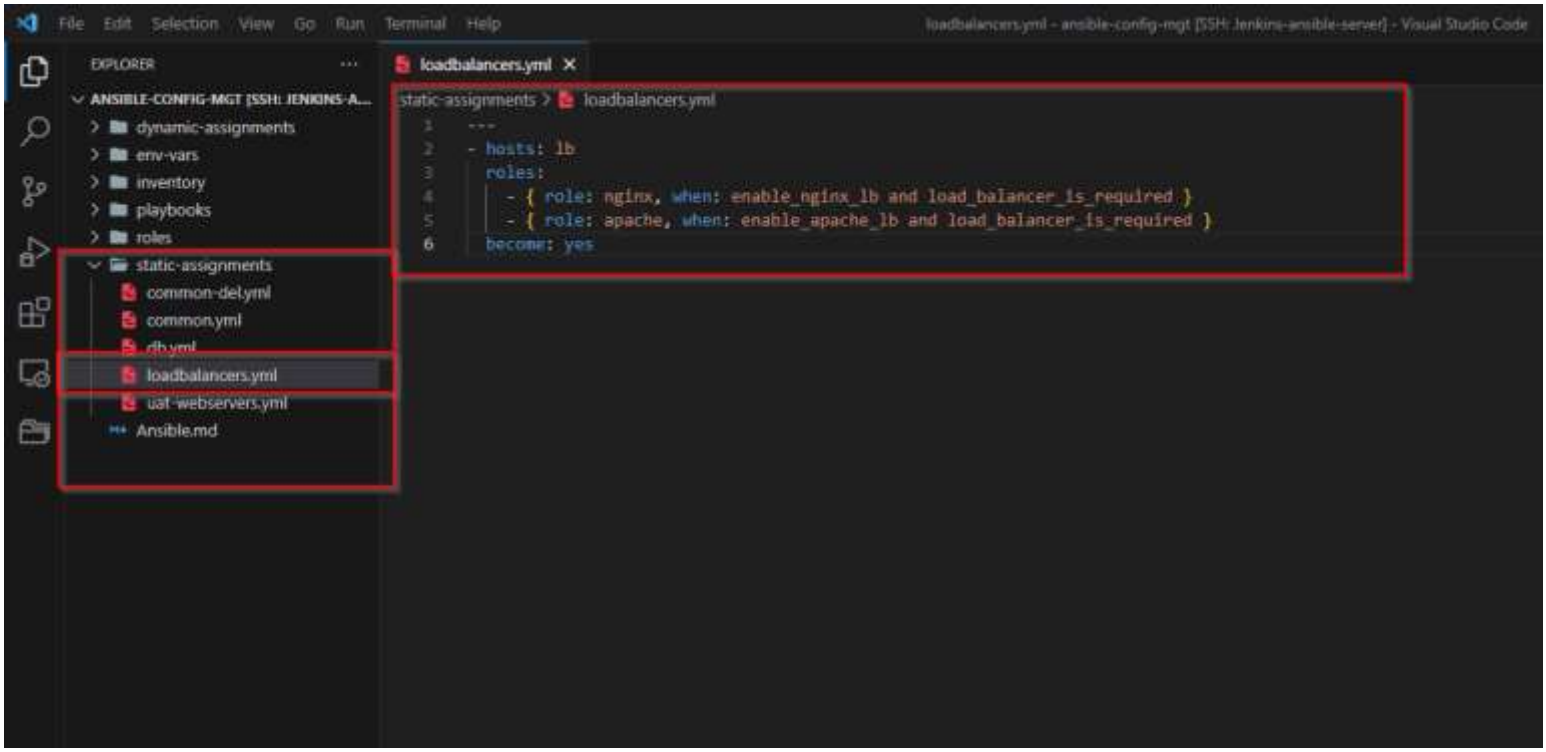
$ ansible-galaxy install geerlingguy.nginx

$ ansible-galaxy install geerlingguy.apache

$ mv geerlingguy.nginx/ nginx

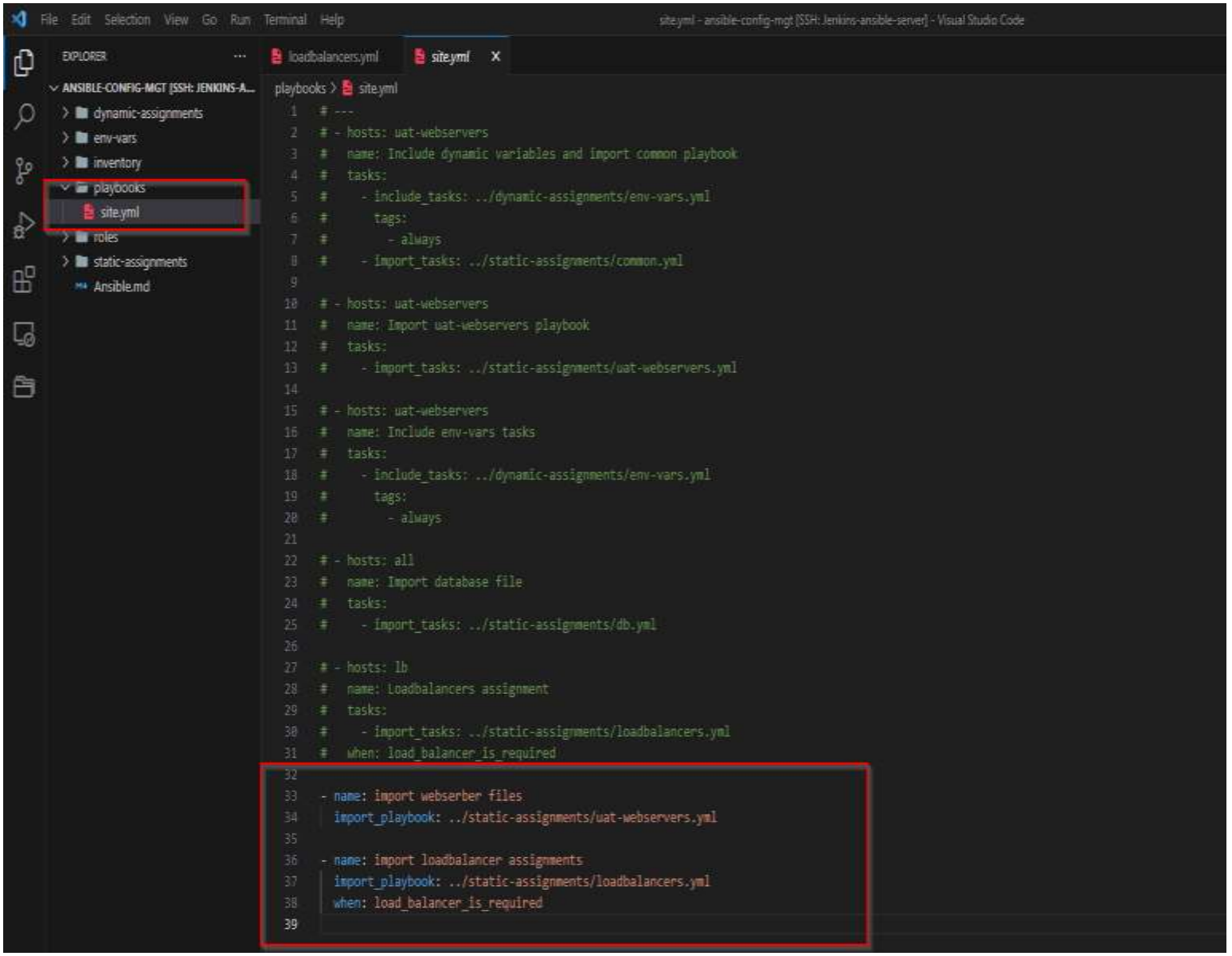
$ mv geerlingguy.apache/ apache

- Update both static-assignment and site.yml files to refer the roles



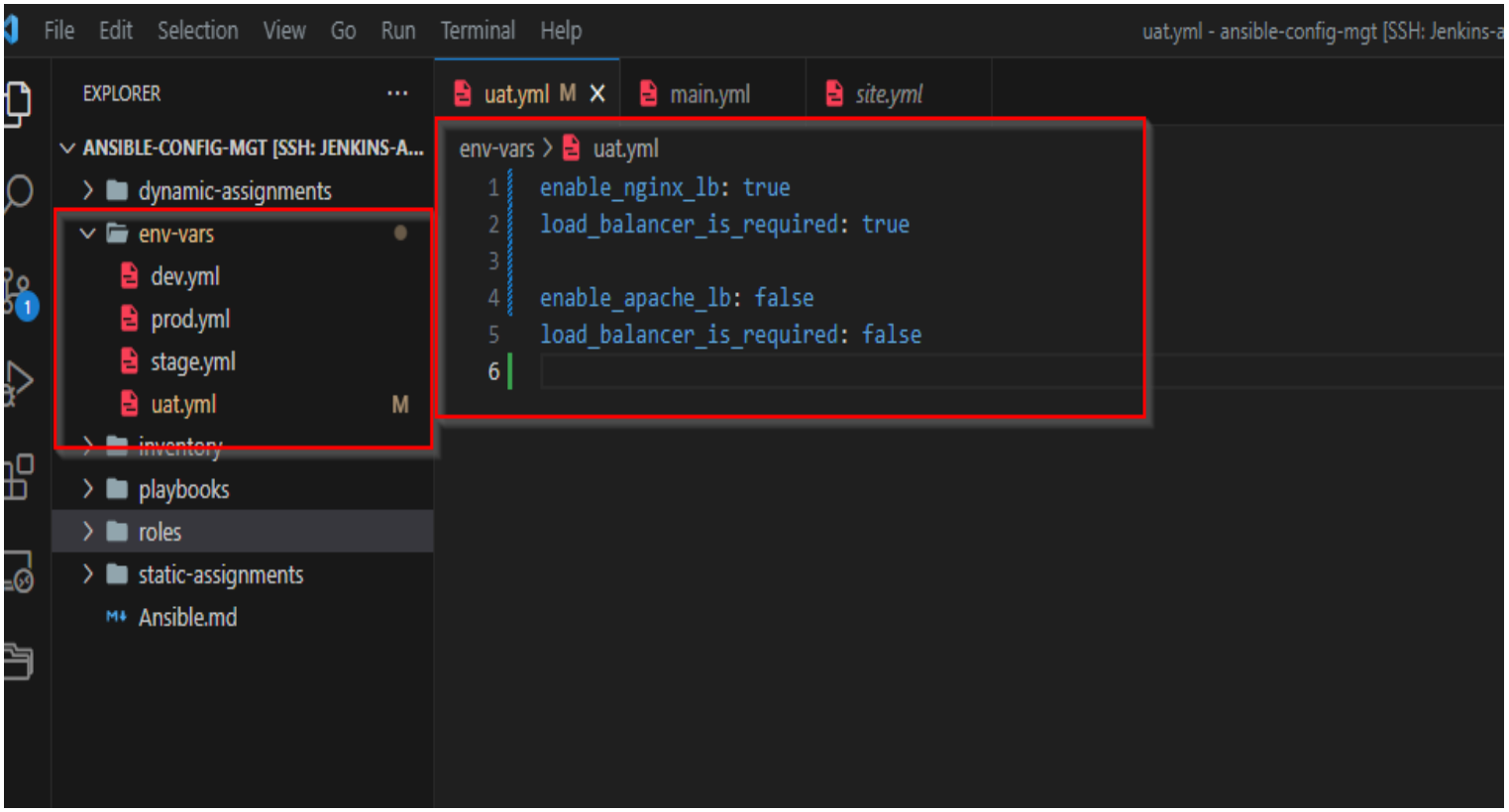
Important Hints:

- Since we cannot use both **Nginx** and **Apache** load balancer, we need to add a condition to enable either one – this is where we can make use of variables.
- Declare a variable in ***defaults/main.yml*** file inside the Nginx and Apache roles. Name each variables **enable_nginx_lb** and **enable_apache_lb** respectively.
- Set both values to false like this **enable_nginx_lb: false** and **enable_apache_lb: false**.
- Declare another variable in both roles **load_balancer_is_required** and set its value to **false** as well.

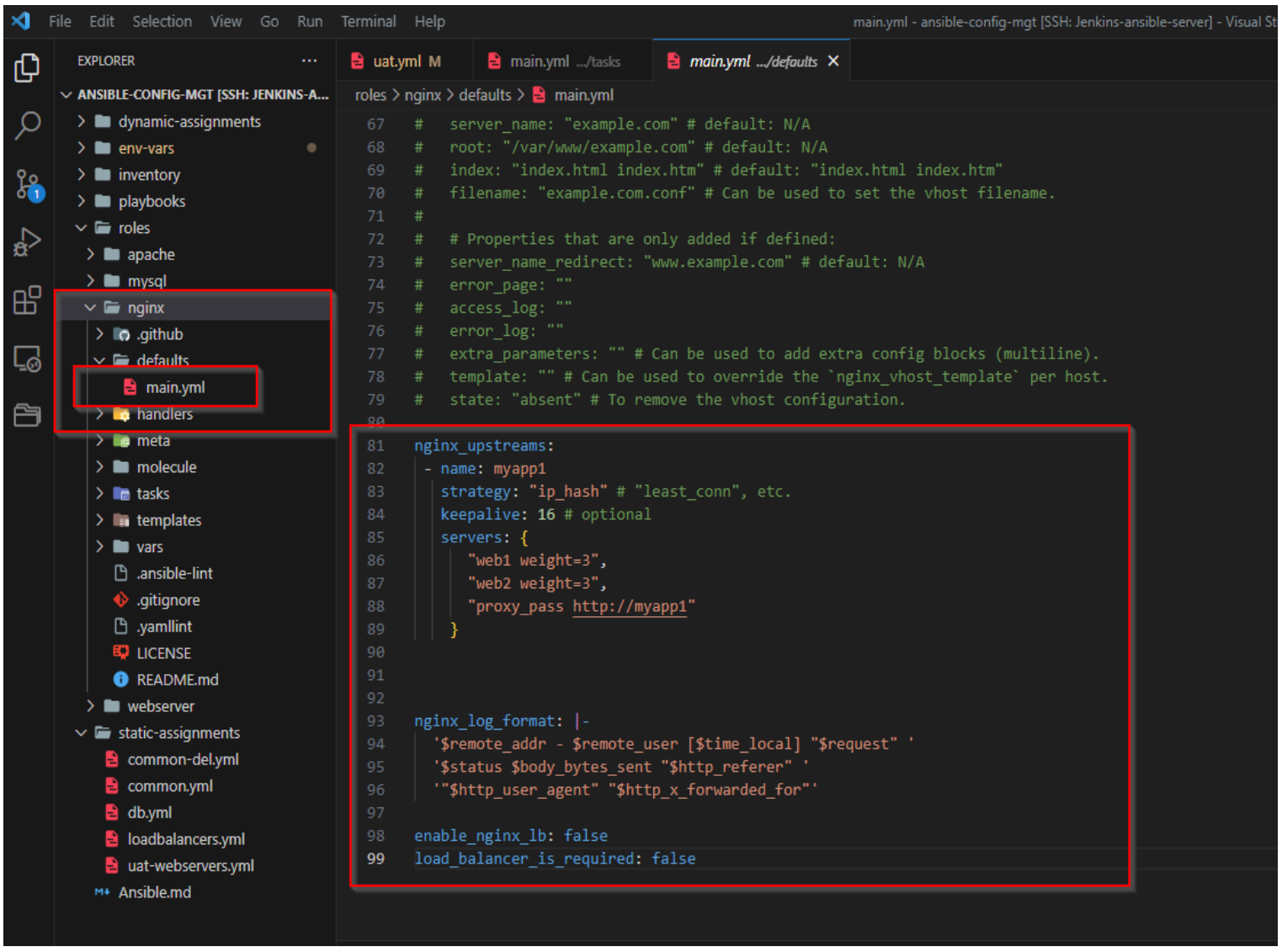


Note the content of the file that has been commented out.

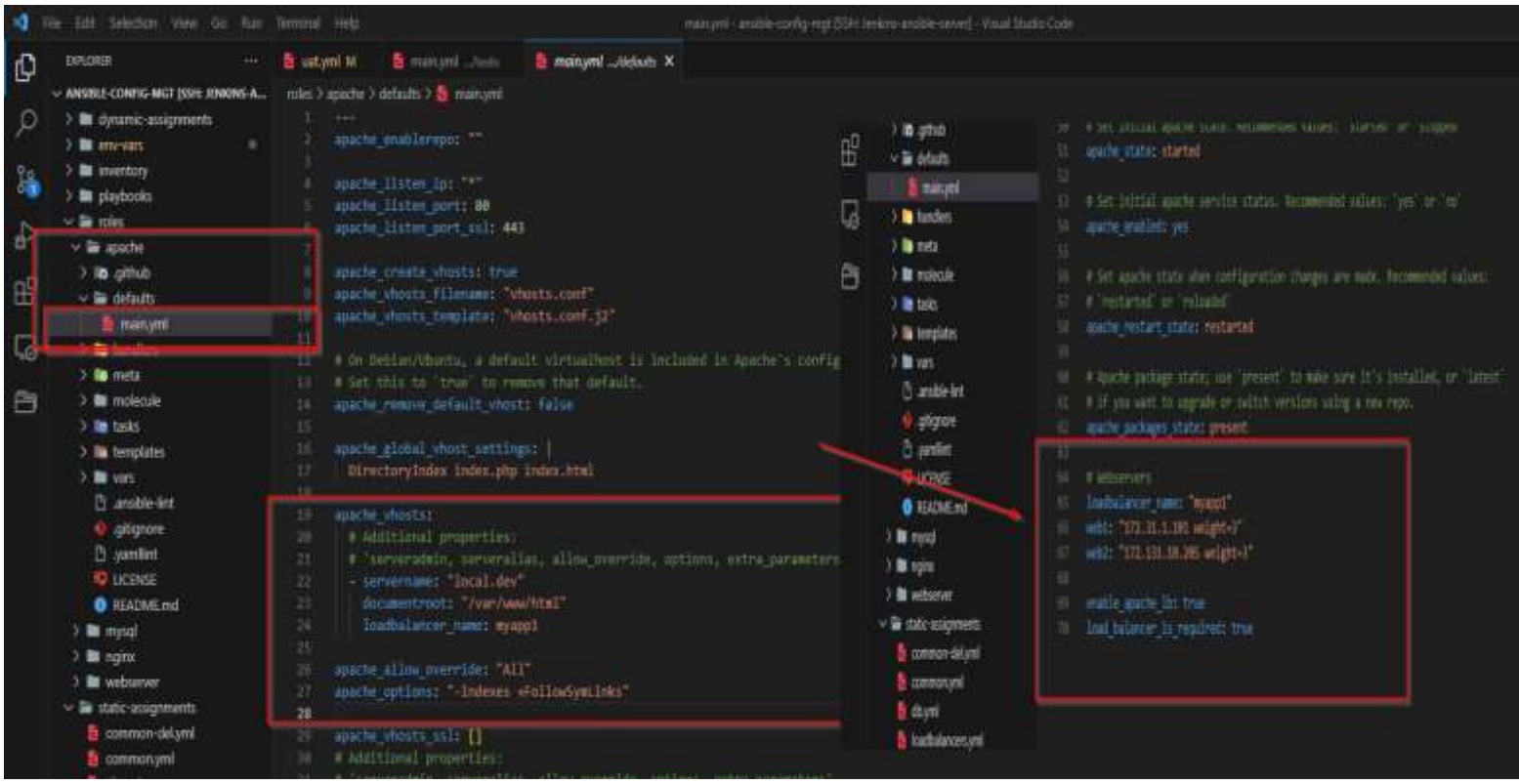
We can make use of ***env-vars\uat.yml*** file to define which loadbalancer to use in UAT environment by setting respective environmental variable to **true**.

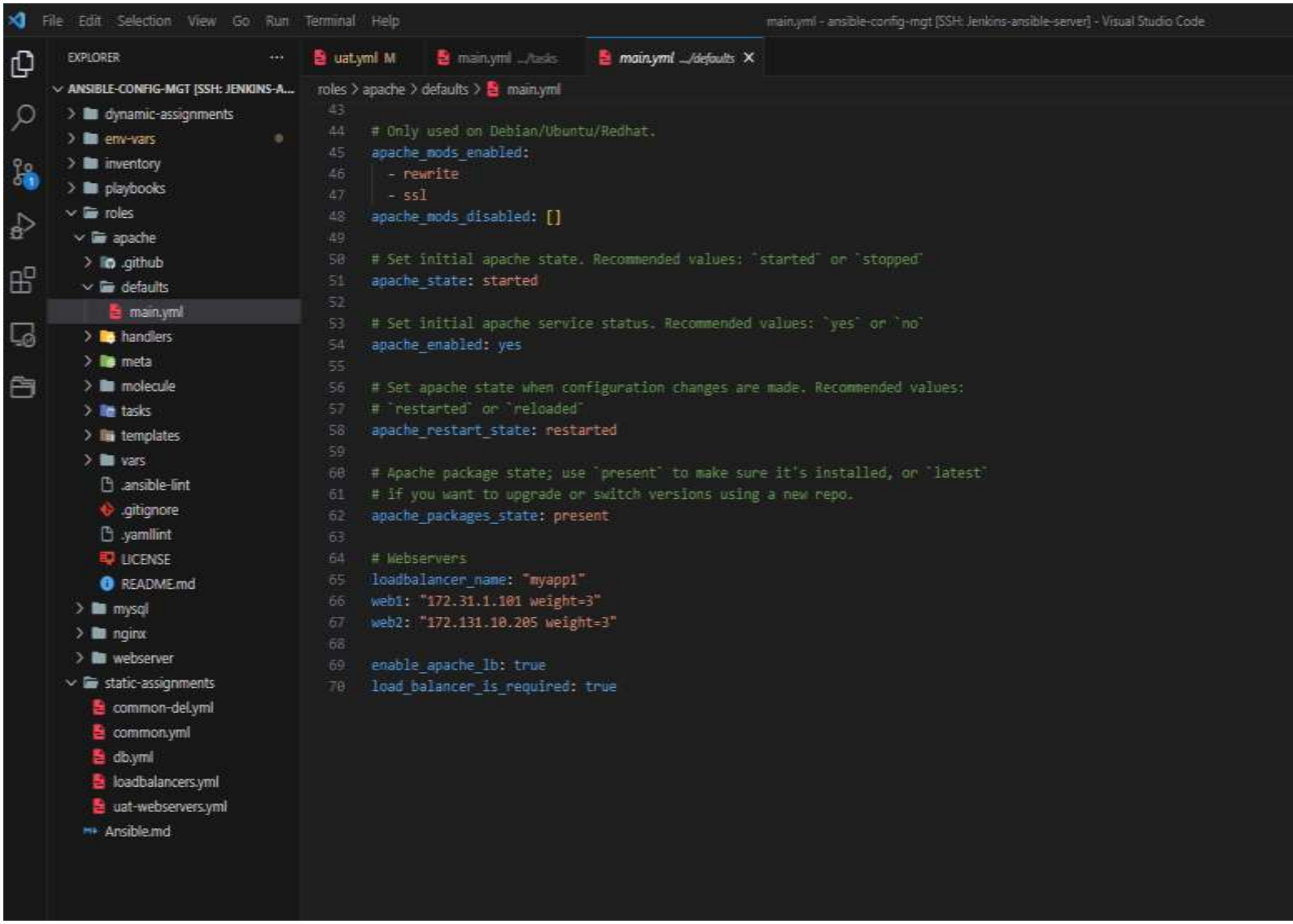


Configure nginx loadbalancer ***defaults/main.yml***

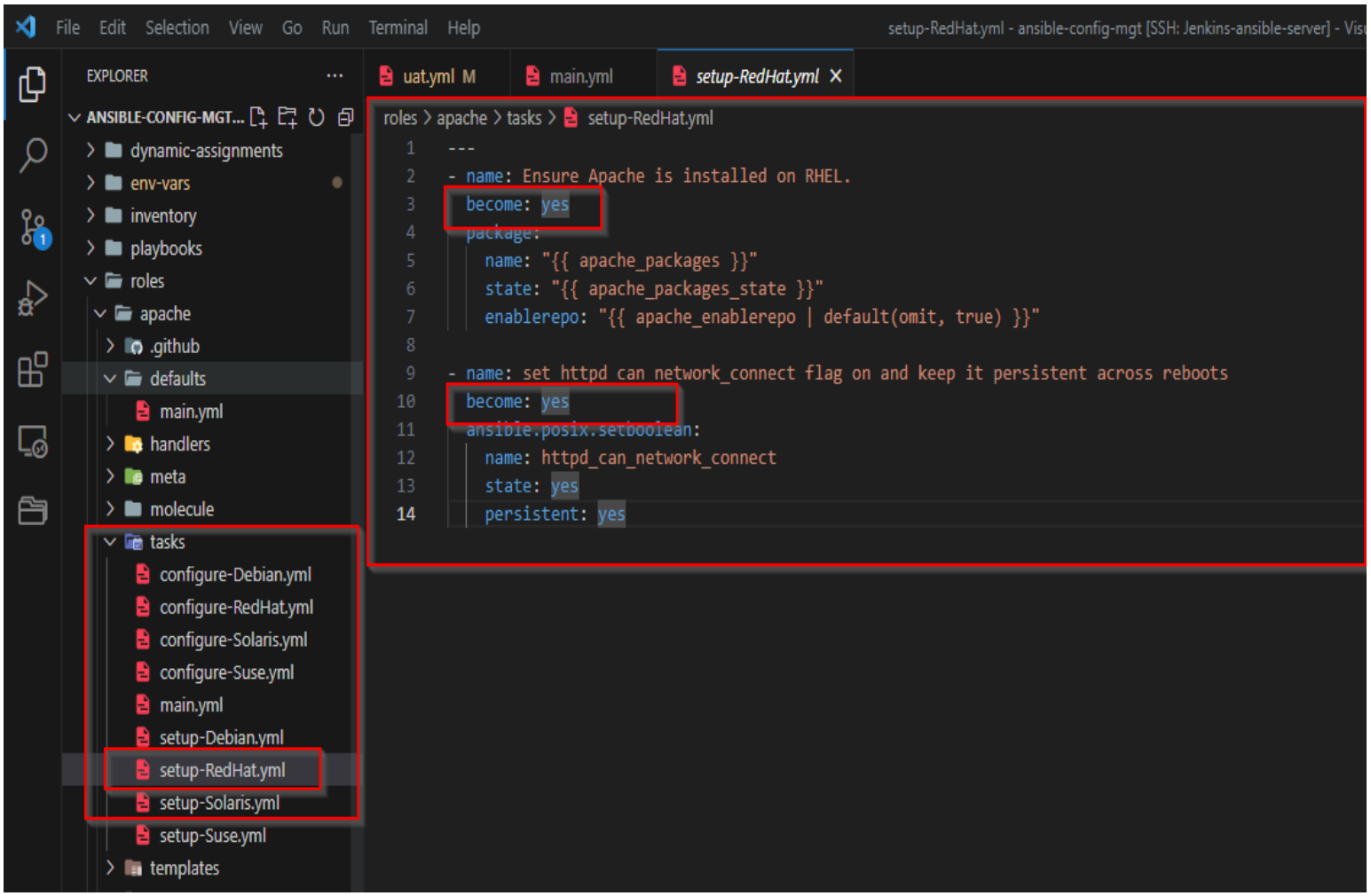


Configure apache loadbalancer ***defaults/main.yml***

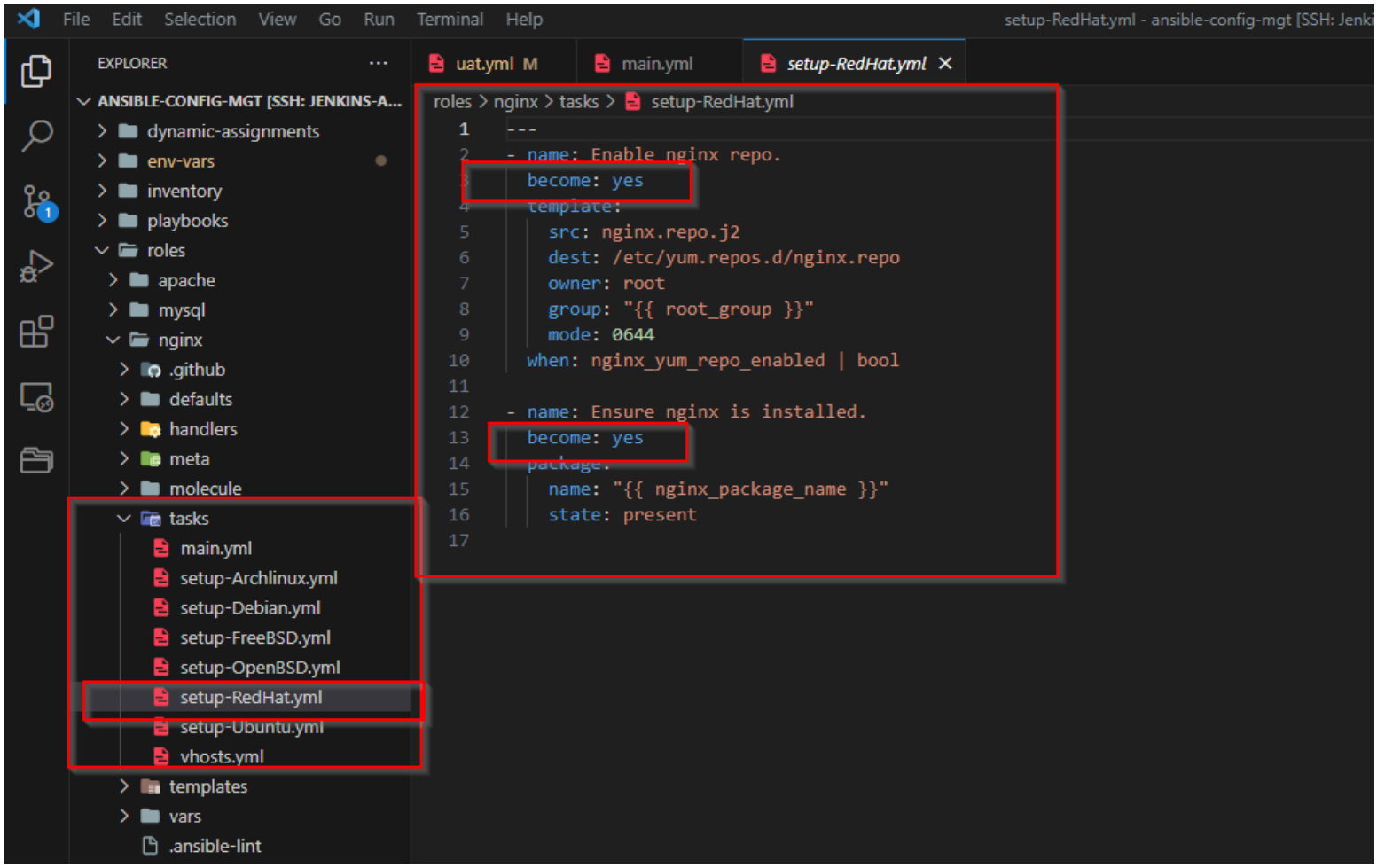




Enable super user on apache on ***roles/apache/tasks/setup-RedHat.yml***

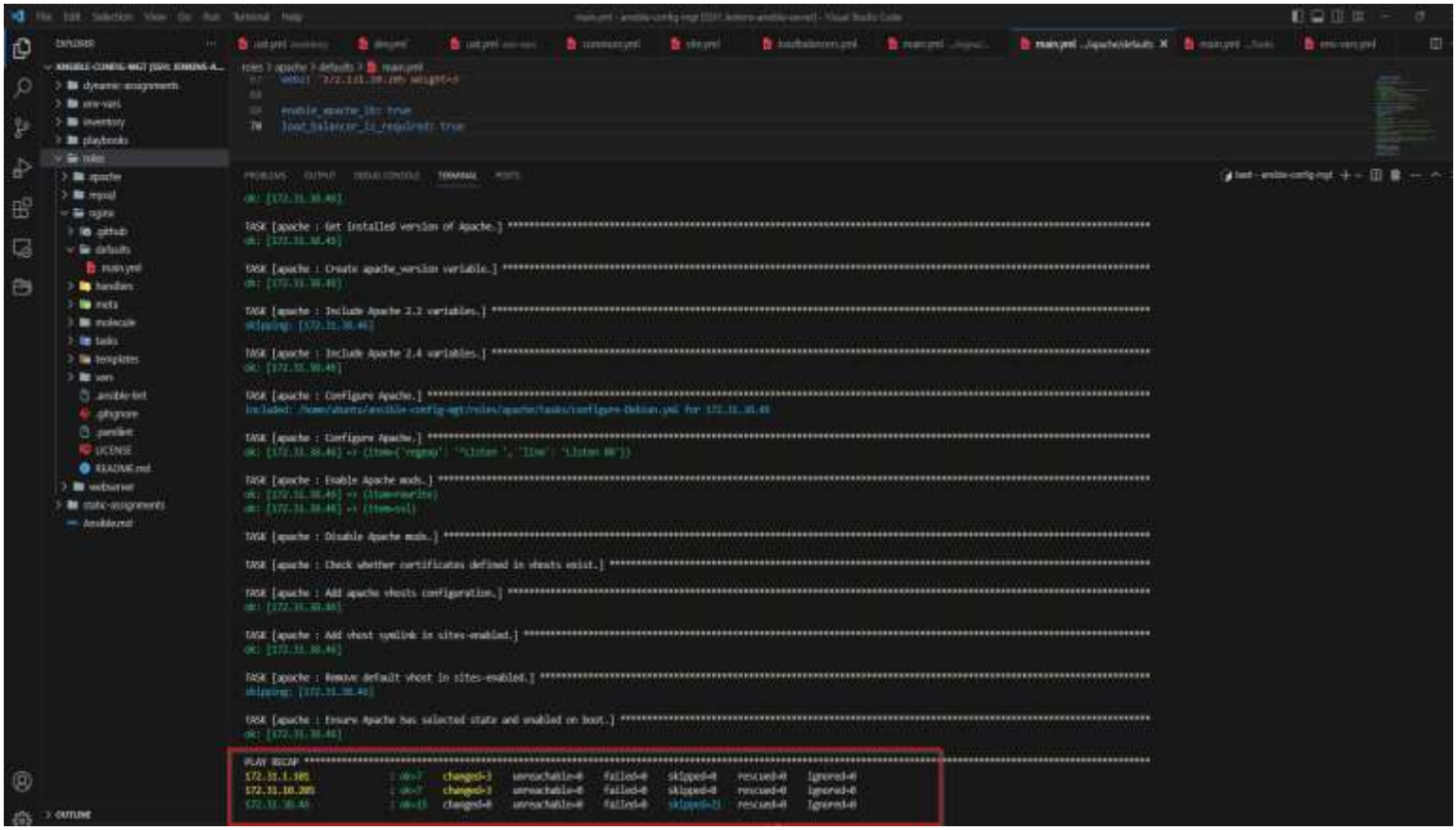


Enable super user on nginx on ***roles/nginx/tasks/setup-RedHat.yml***



Run Ansible against the **uat** and load balancer environment.

$ ansible-playbook -i inventory/uat.yml playbooks/site.yml



Congratulations!!!

We have learned and practiced how to use Ansible configuration management tool to prepare UAT environment for Tooling web solution.