Contents

[ACTIVITY 6: Creating a playbook: 1](#_Toc517775226)

[ACTIVITY 7:Using conditions and defining Handlers: 4](#_Toc517775227)

[ACTIVITY 8: Introduction to roles, creating site.yml 9](#_Toc517775228)

[ACTIVITY 9: Ansible Galaxy 13](#_Toc517775229)

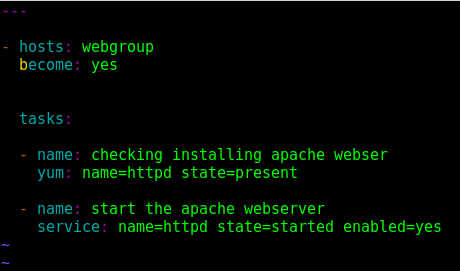
## **ACTIVITY 6:** Creating a playbook:

* + A playbook can have multiple plays.
  + Plays map hosts to tasks
  + A play can have multiple tasks
  + Playbook is a YAML file. A file with extension .yaml or .yml

1. Create a directory lab 5. Create a playbook inside lab5

(Faculty: to explain each section of the playbook )

vi myfirstplaybook.yml

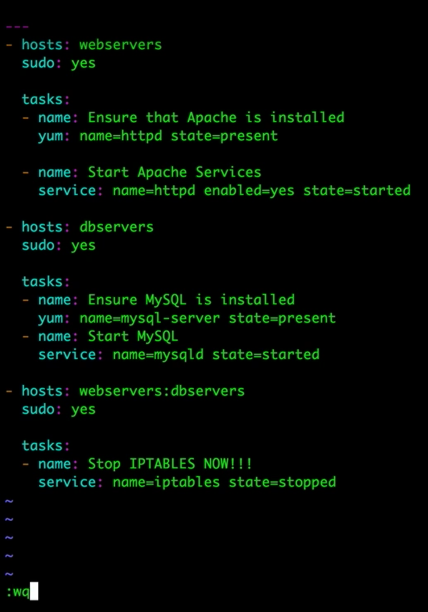


1. Extending the same to another group called “dbservers”/”dbgroup”

Try to append the code without referring to the section above.

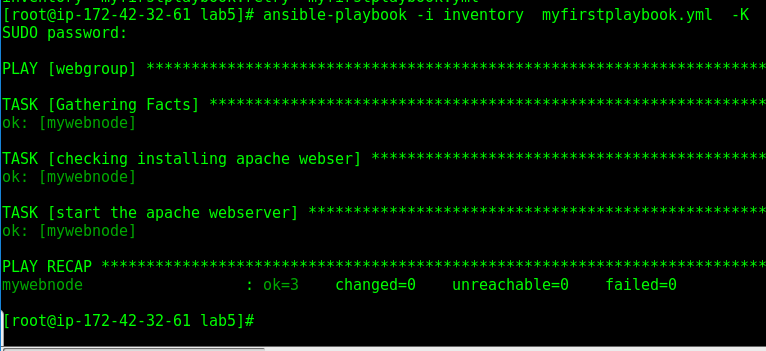
( though in your lab environment, you might have only one node, you can still practice writing the same in the playbook )

Once done you can verify whether you have written as below:

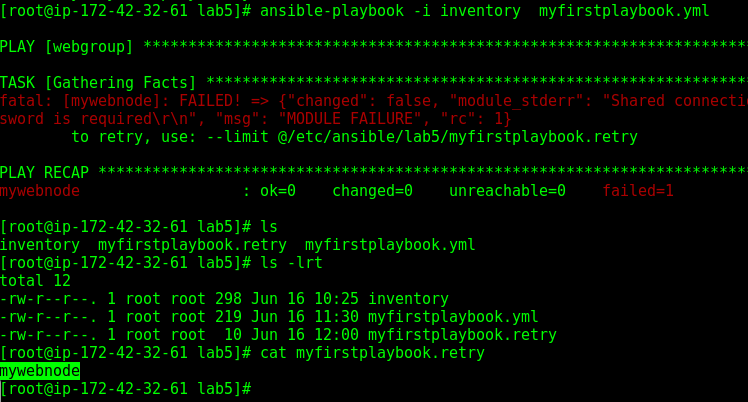


1. Comment out the extended group’s lines and execute the play for webgroup node:

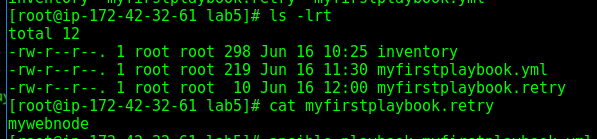
If successful the below will be displayed:

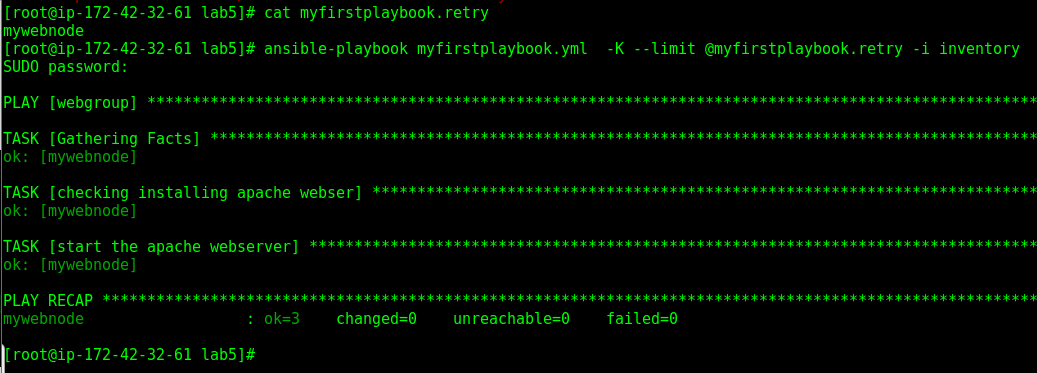


Incase of error, observe that a retry file will be created, listing the nodes where it failed to execute the tasks:



Then you can make the changes and execute only for the node group listed in the retry file. Instead of “all”





## ACTIVITY 7:Using conditions and defining Handlers:

mkdir lab6 in /etc/ansible

vi mysecondplaybook.yml

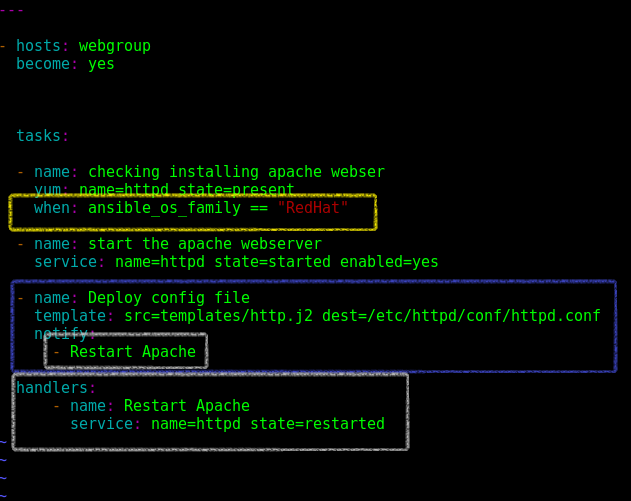
1. adding a when condition:

Note that Default task gathering\_facts is yes, hence everytime a playbook runs, it would gather all the facts from the hosts/nodes, thus checking the condition for one of the variable values, i.e ansible\_os\_family whether the value retrieved from hosts/nodes is “RedHat”.

If these facts are not used in your playbook, then the gathering\_facts can be set to no, as it takes a significant amount of time.

1. Adding handlers.

If a particular event happens, then execute the handler. In the below scenario if there is a change in httpd.conf file in the node, then the handler of restarting httpd service will be executed.



1. Adding variables and values and also a prompt at execution of playbook :

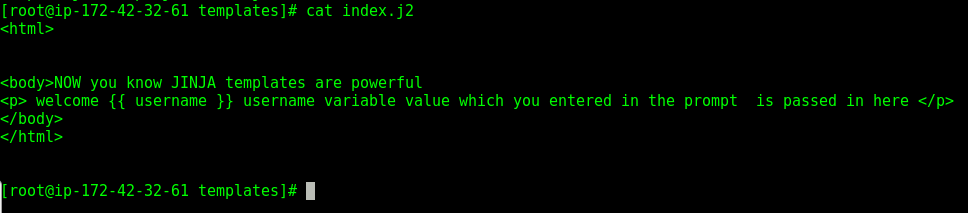
In the below example, http\_port, doc\_dir, doc\_root variables are defined with values.

the prompt should be written under “vars\_prompt”.

In the below example user will be asked to enter input at the execution of playbook and the value will be stored under the variable “username”

1. Creating jinja2 files and passing variable values. In the below example, 2 jinja2 files are used: index.j2 and httpd.j2.

**Create a folder called “templates”** in parallel to your group\_vars or inventory file and create **two files named index.j2 and httpd.j2** with the touch command. The contents of the jinja files are as below:

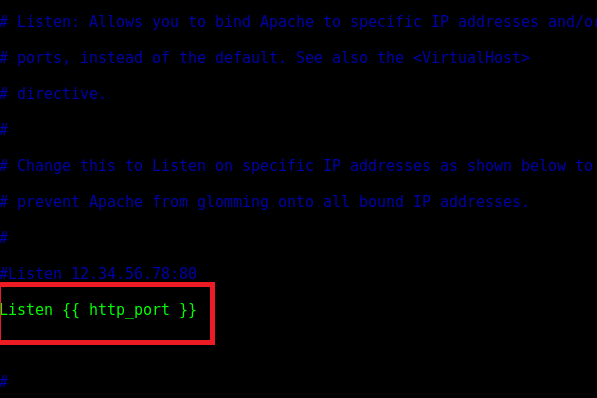


The httpd.j2 is as attached here. Transfer the content of the attached file into httpd.j2

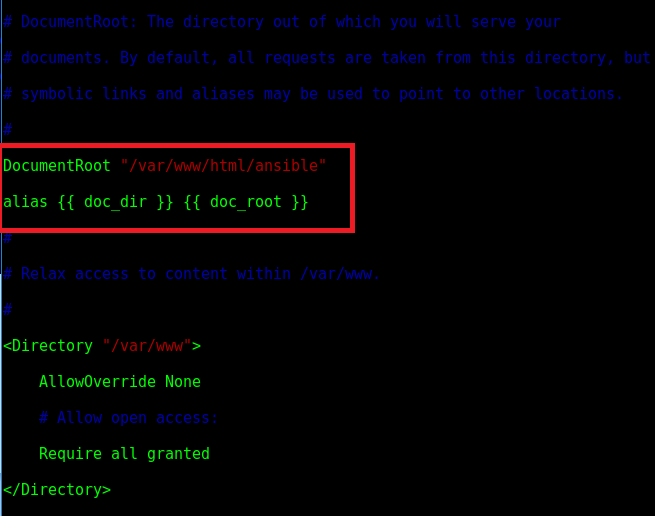


**Note** the highlighted variables below are to be modified. Open the file and make edits as highlighted below:

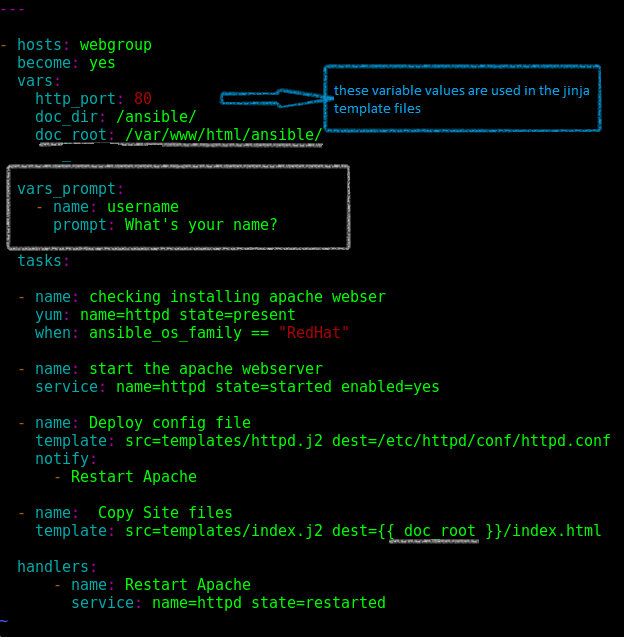
Here the http\_port value comes from the playbook variables defined in **the var section.**



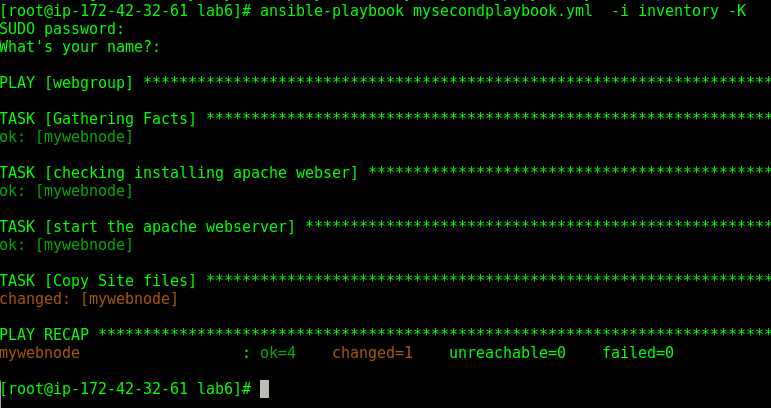
Here the doc\_root and the doc\_dir , again comes from the variables defined in the playbook.



1. The playbook is written as below:

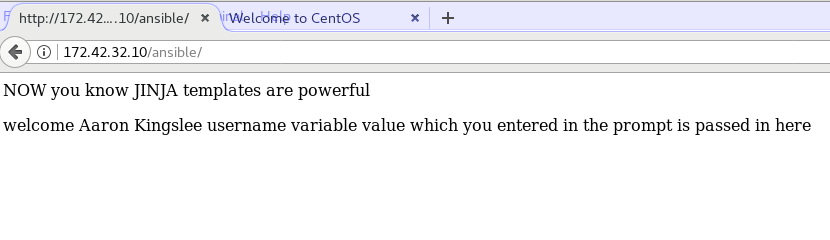


1. Execute the playbook as below and observe the results, ***before that make sure the ansible directory is created in the /var/www/html folder in the node***.:



At the prompt input “Aaron Kingslee”

1. At successful, execution , go to node and verify the files http.conf and index.html created /changed in the respective locations.
2. Also access the IP/ansible and observe the page displaying the below:

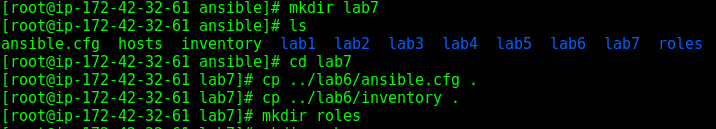


## ACTIVITY 8: Introduction to roles, creating site.yml

Default roles directory exists in /etc/ansible.

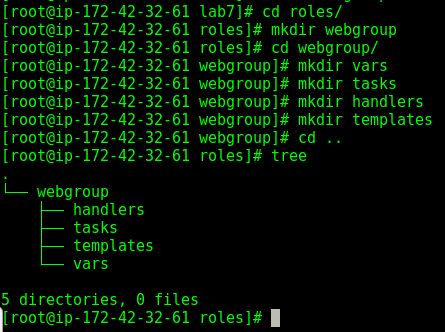
Create Roles directory in lab7

Inside the roles directory in lab7, create directories as mentioned below:

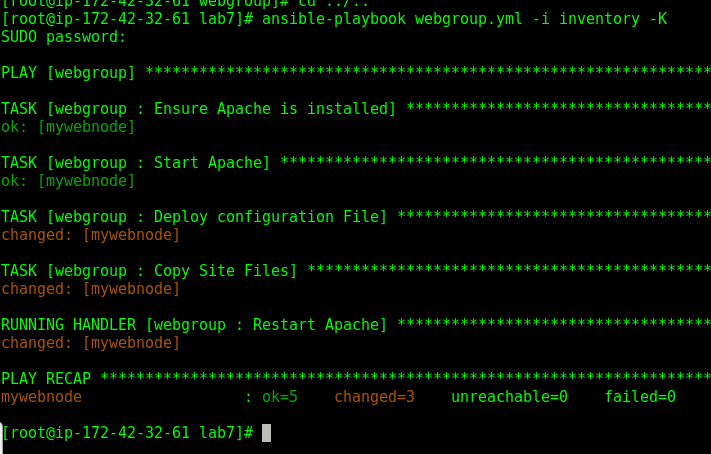


Create a webgroup directory inside the roles and

Inside webgroup create - vars, tasks, handlers, templates directories.

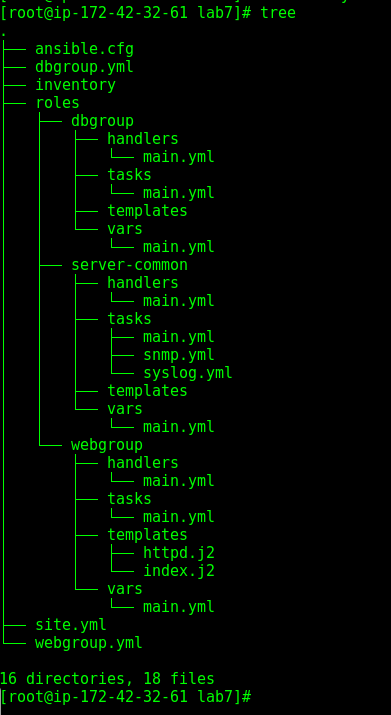


Copy the same playbook as in lab6 into lab7 as webgroup.yml and can execute the playbook once and check whether it executes fine.



Just to practice the various directories, extend the roles to the “dbgroup”, this time try to create the folders without referring to the above and check your memory.

Once done verify as below:



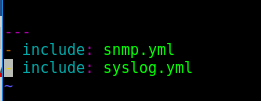
Creating site.yml file which will inturn call all the ymls :



Below is the content of site.yml fine. Not that what is mentioned inside server-common will be executed for both dbgroup and webgroup. Also webgroup.yml and dbgroup.yml is written seperately will do their respective tasks on their respective hosts/nodes.



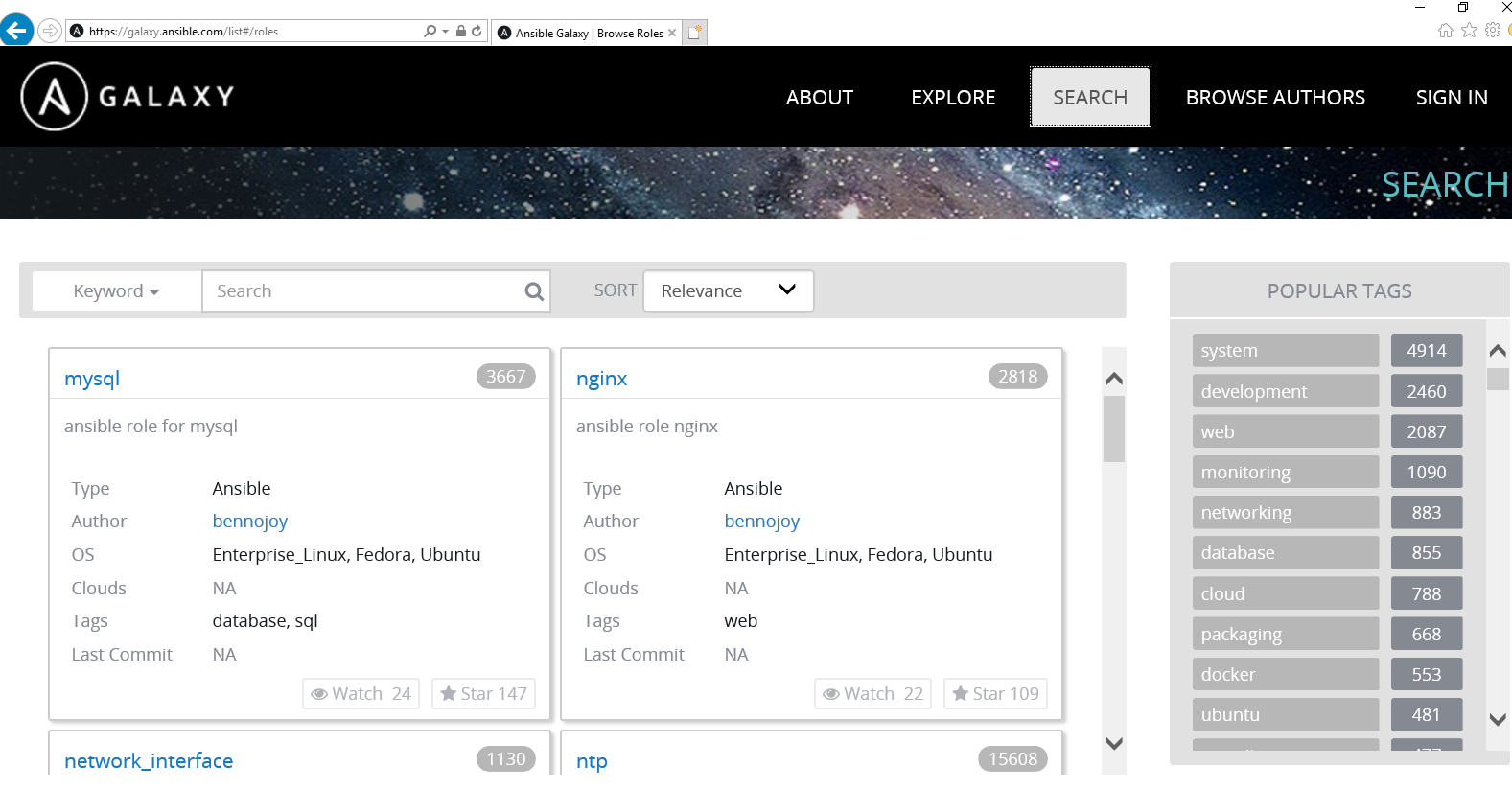




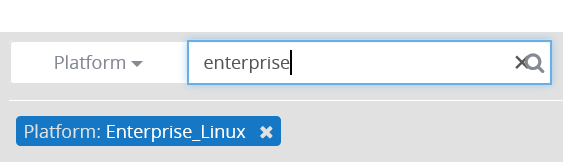
Above is just an example of included files. Though the above cannot be executed, as we have only one node, this activity was to practice to create site.yml

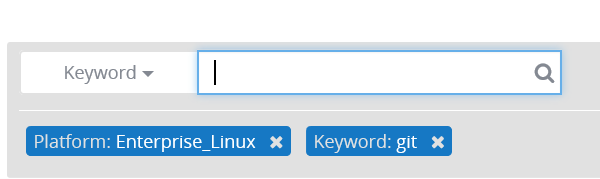
## ACTIVITY 9: Ansible Galaxy

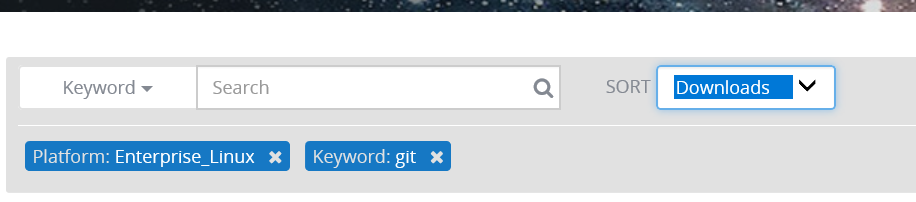
Go to Galaxy.ansible.com:

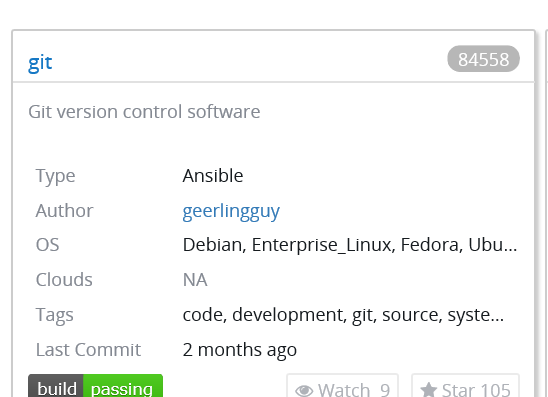


Set the filter: for the required platform and keyboard:

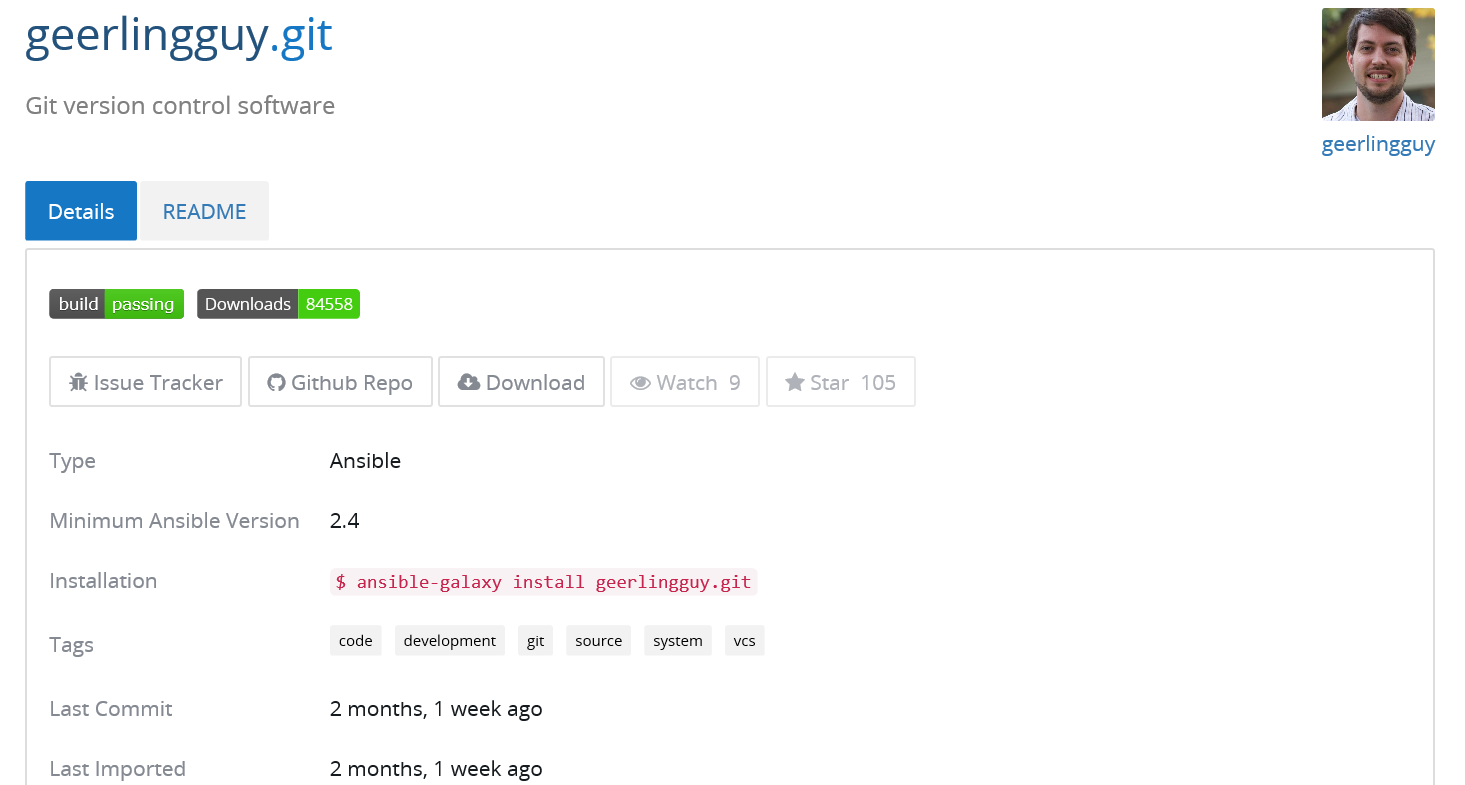


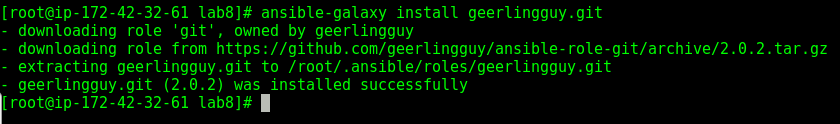






The search result for the above filter is as below:





Create a git.yml file:



