| CPE 232 PRELIM EXAM | |
|---|---|
| Pastrana, Mark Laurenz S. | Start date: Sep 28, 2023 |
| CPE31S5 | Submitted: Oct 2, 2023 |
| CPE 232 - Managing Server Enterprise | Engr. Richard Roman |

Tools Needed:

1. Control Node (CN) - 1
2. Manage Node (MN) - 1 Ubuntu
3. Manage Node (MN) - 1 CentOS

Procedure:

Note: You are required to create a document report of the steps you will do for this exam. All screenshots should be labeled and explained properly.

1. **Create a repository in your GitHub account and label it as Surname_PrelimExam**

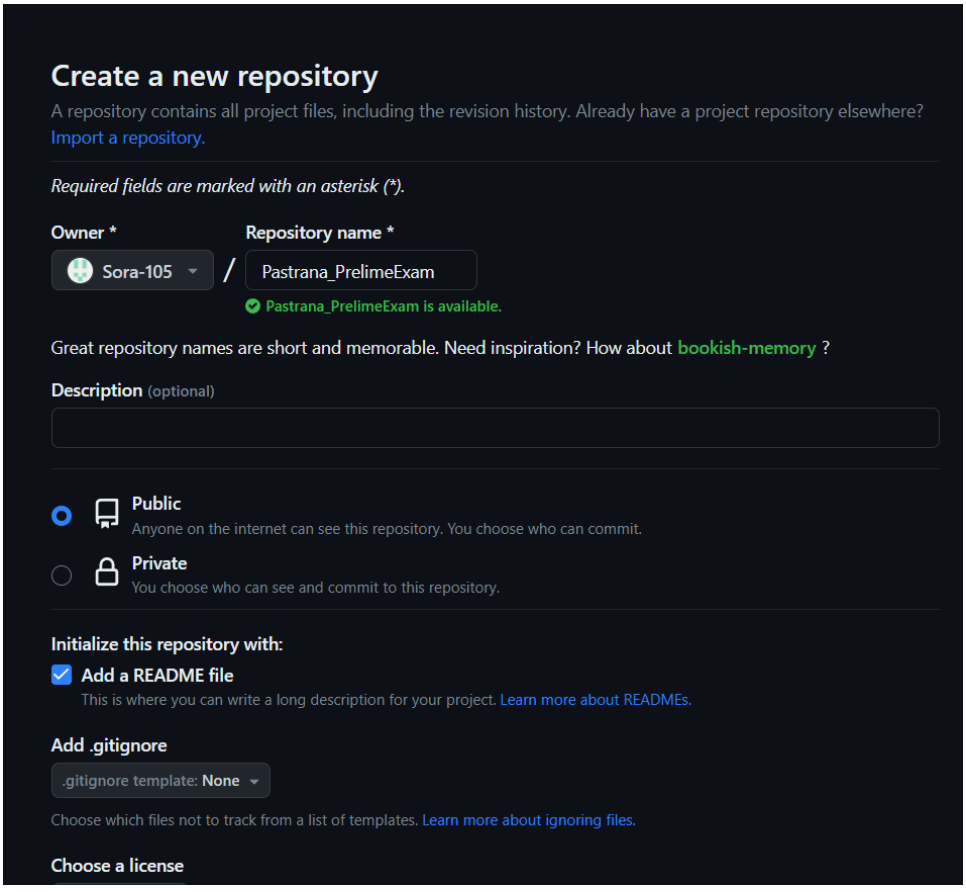


Figure 1.1: This is the new repository in my git account that I just created.

**2. Clone your new repository in your CN.**

```
pastrana@localmachina:~$ git clone git@github.com:Sora-105/Pastrana_PrelimeExam.git
Cloning into 'Pastrana_PrelimeExam'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
pastrana@localmachina:~$
```

Figure 2.1: I cloned my repository that I just created in git. When entering this command and pasting the ssh link that I copied in git, it will automatically create a new directory on my local machine.

**3. In your CN, create an inventory file and ansible.cfg files.**

**inventory:**

```
pastrana@localmachina: ~/Pastrana_PrelimeExam
 GNU nano 6.2                                                inventory *
[servers]
#server1
192.168.56.107 ansible_python_interpreter=/usr/bin/python3
#centos
192.168.56.109 ansible_python_interpreter=/usr/bin/python3
```

Figure 3.1: Issuing the command *sudo nano "{{ filename }}"* will create a new nano file in your directory. In this case, I just copied the contents of the previous activity.

**ansible.cfg:**

```
pastrana@localmachina: ~/Pastrana_PrelimeExam
 GNU nano 6.2                                                ansible.cfg
[defaults]

inventory = inventory
host_key_checking = False
deprecation_warning = False

remote_user = pastrana
private_key_file = ~/.ssh/
```

Figure 3.2: Issuing the command *sudo nano "{{ filename }}"* will create a new nano file in your directory. In this case, I just copied the contents of the previous activity.

```
pastrana@localmachina:~/Pastrana_PrelimeExam$ ansible all -m ping
192.168.56.107 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
192.168.56.109 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
```

Figure 3.3: I pingped both IP addresses to see if they were really reachable.

4. **Create an Ansible playbook that does the following with an input of a config.yaml file for both Manage Nodes**

    - **Installs the latest python3 and pip3**


Figure 4.1: In this new YAML file, I'm asked to install Python 3 and Python 3. That's why I entered those commands in lines. We have to be very precise with the indention and spaces because there will be an error if we're lacking or have excess space.

```
pastrana@localmachina:~/Pastrana_PrelimeExam$ ansible-playbook --ask-become-pass config.yaml
BECOME password:

PLAY [all] ***********************************************************************************************************

TASK [Gathering Facts] ***********************************************************************************************
ok: [192.168.56.107]
ok: [192.168.56.109]

TASK [Installs latest python3 and pip3] ******************************************************************************
ok: [192.168.56.107]
changed: [192.168.56.109]

PLAY RECAP ***********************************************************************************************************
192.168.56.107             : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
192.168.56.109             : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Figure 4.2: Issuing this command will automatically run the YAML file and do the task that I enter in it. In this scenario, it successfully installed Python 3 and pip3 on those servers and local machines.

- **use pip3 as default pip**

```
pastrana@localmachina: ~/Pastrana_PrelimeExam
  GNU nano 6.2                                                inventory
[servers]
#server1
192.168.56.107 ansible_python_interpreter=/usr/bin/python3 pip_package=pip3
192.168.56.107 apache_package=apache2 php_package=libapache2-mod-php

#centos
192.168.56.109 ansible_python_interpreter=/usr/bin/python3 pip_package=pip3
192.168.56.109 apache_package=httpd php_package=php
```

Figure 4.3: I set the pip3 as default.

- **use python3 as default python**

```
pastrana@localmachina: ~/Pastrana_PrelimeExam
  GNU nano 6.2                                                inventory
[servers]
#server1
192.168.56.107 ansible_python_interpreter=/usr/bin/python3 pip_package=pip3
192.168.56.107 apache_package=apache2 php_package=libapache2-mod-php

#centos
192.168.56.109 ansible_python_interpreter=/usr/bin/python3 pip_package=pip3
192.168.56.109 apache_package=httpd php_package=php
```

Figure 4.4: I set python3 as default python.

- **Install Java open-jdk**

```
- name: Installs Java open-jdk in Ubuntu
  apt:
    name:
    - openjdk-17-jdk
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: Installs Java open-jdk in CentOS
  yum:
    name:
    - java-11-openjdk
    state: latest
    update_cache: yes
  when: ansible_distribution == "CentOS"
```

Figure 4.5: In the same YAML file I just typed this syntax that will install Java in both Ubuntu and CentOS. I created 2 modules since there is a different command for CentOS.

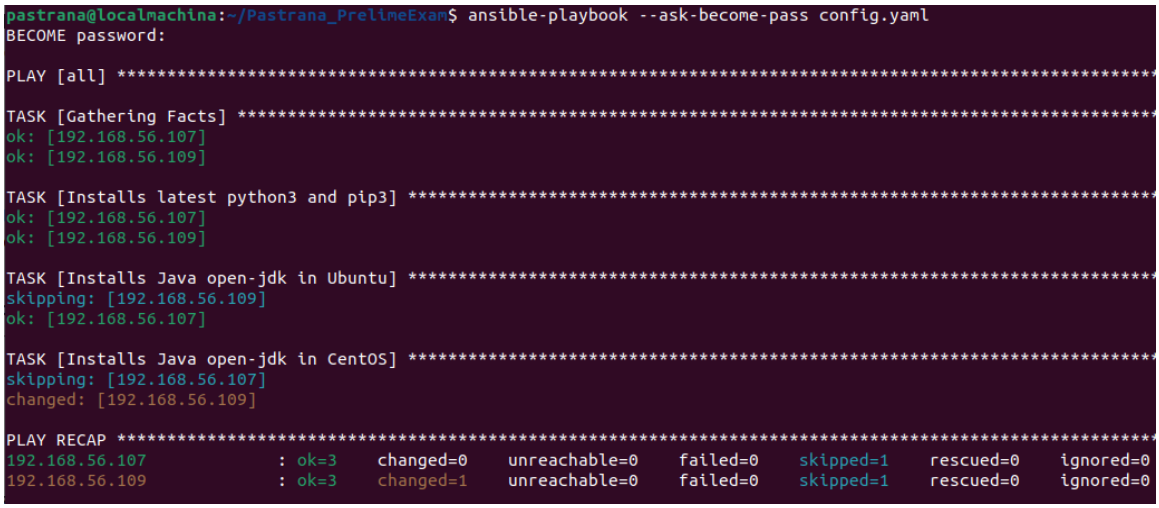



Figure 4.6: I ran again the command that will automatically install in and it was successfully done.

- **Create Motd containing the text defined by a variable defined in config.yaml file and if there is no variable input the default motd is "Ansible Managed node by (your user name)"**

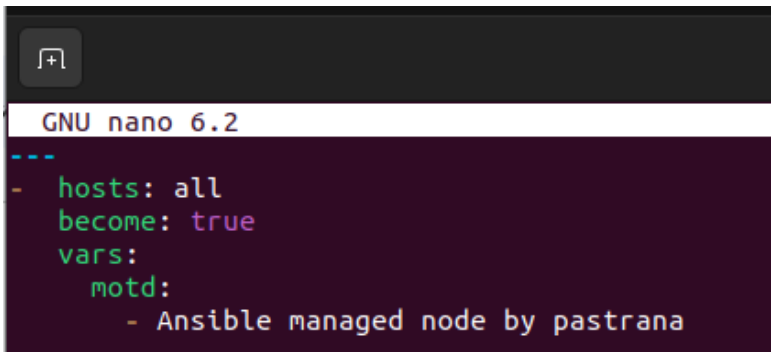



Figure 4.7: Now I'm asked to make a "message of the day" to be executed once I run the playbook command.


Figure 4.8: This is the module for the MOTD

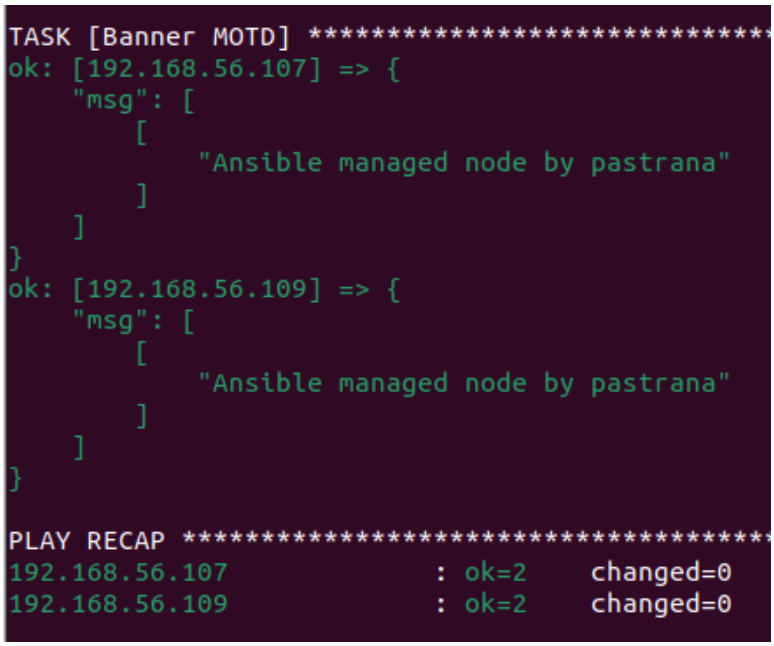



Figure 4.9: It was successfully displayed.

- **Create a user with a variable defined in config.yaml**

```
vars_promt:
    - name: username
      prompt: Input username
      private: false

    - name: uid
      prompt: Input ID
      private: false
```

Figure 4.10: This is the variable that will prompt the user.

```
- name: Create a User
  ansible.builtin.user:
    name: "{{ username }}"
    comment: New User
    uid: "{{ uid }}"
    createhome: yes
    home: /home/"{{ username }}"
    shell: /bin/bash
```

Figure 4.11: This is the module that will create a new user.

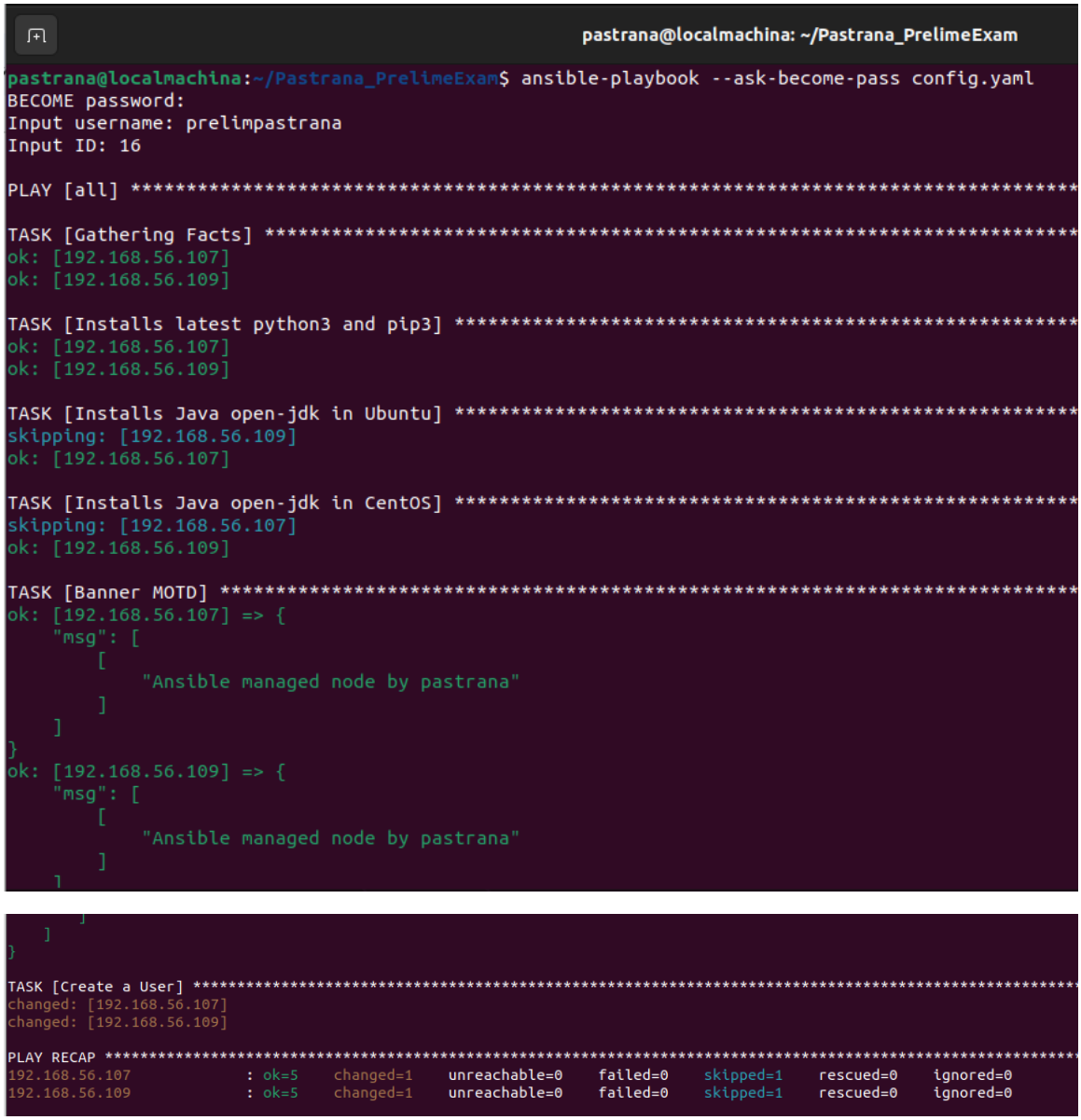



Figure 4.12: Once again, I execute the command and all of the modules that are in the YAML file successfully complete their tasks.

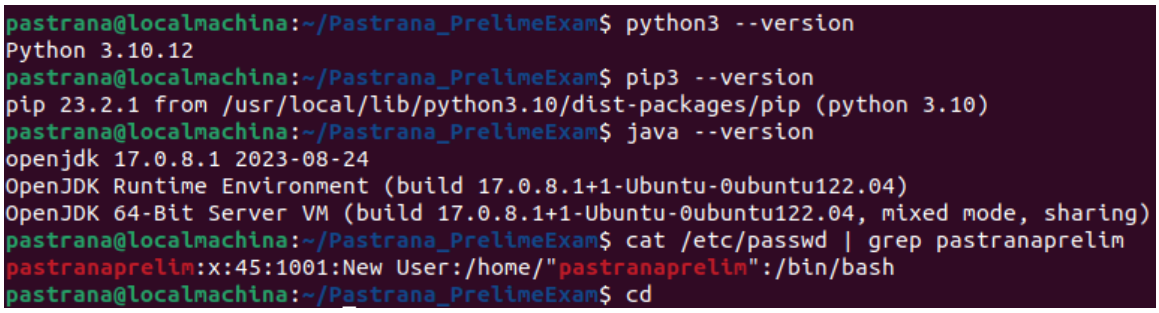


Figure 4.13: Verifying those installations in Ubuntu.

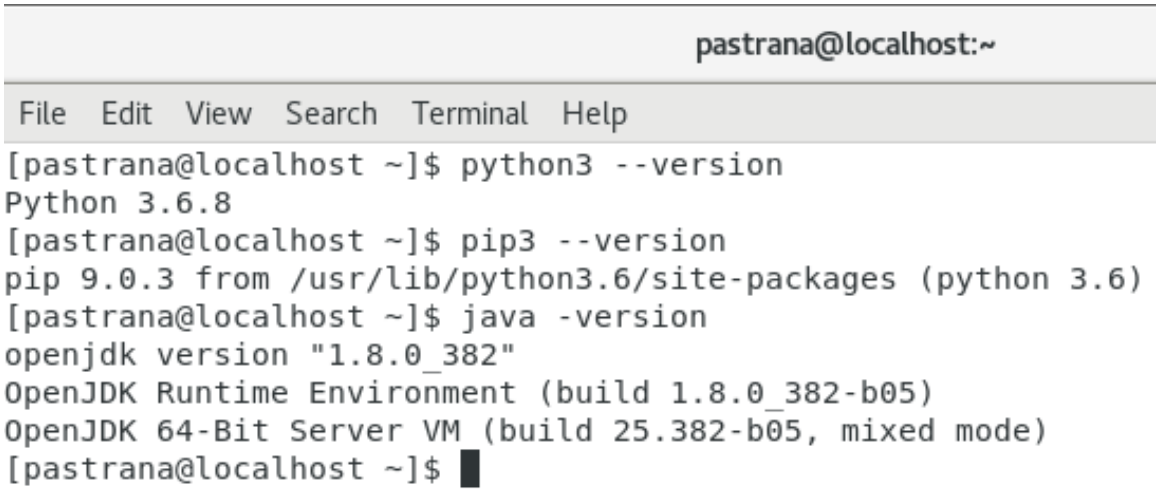



Figure 4.14: Verifying those installations inCentOS.

**Config.yaml file:**

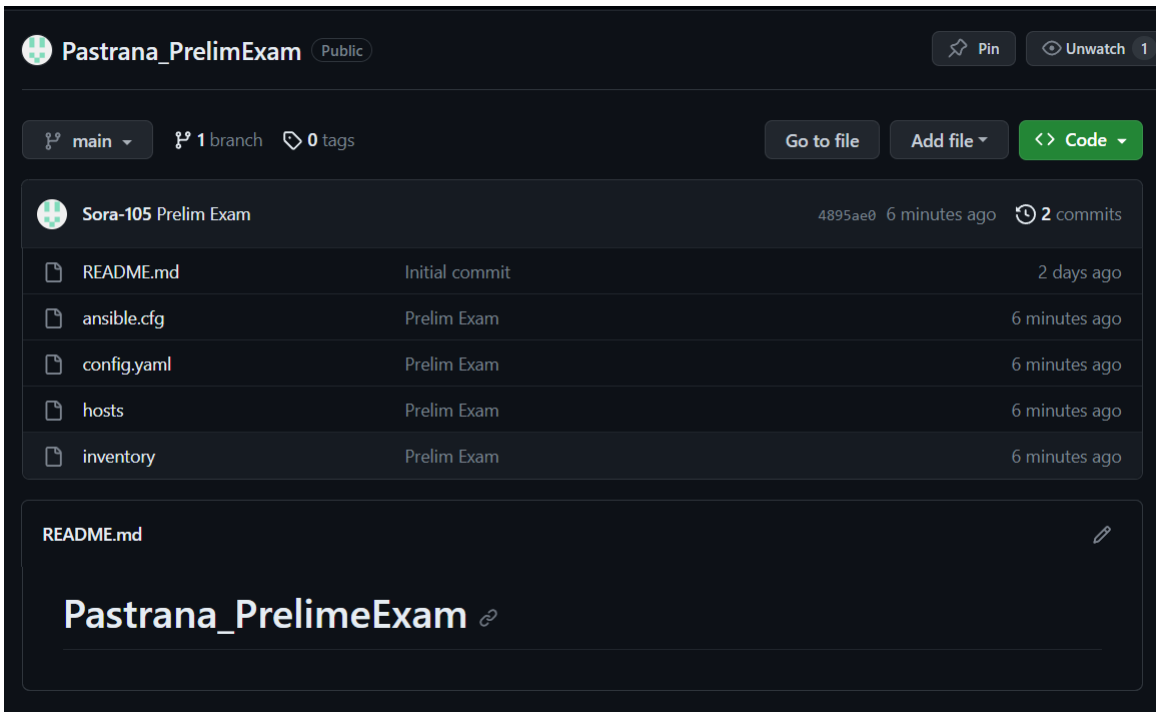

**5. PUSH and COMMIT your PrelimExam in your GitHub repo**

```
pastrana@localmachina:~/Pastrana_PrelimeExam$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ansible.cfg
        config.yaml
        hosts
        inventory

nothing added to commit but untracked files present (use "git add" to track)
pastrana@localmachina:~/Pastrana_PrelimeExam$ git add *
pastrana@localmachina:~/Pastrana_PrelimeExam$ git commit -m "Prelim Exam"
[main 4895ae0] Prelim Exam
 4 files changed, 73 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 config.yaml
 create mode 100644 hosts
 create mode 100644 inventory
pastrana@localmachina:~/Pastrana_PrelimeExam$ git push origin
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 1.10 KiB | 1.10 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
remote: This repository moved. Please use the new location:
remote:   git@github.com:Sora-105/Pastrana_PrelimExam.git
To github.com:Sora-105/Pastrana_PrelimeExam.git
   26a8357..4895ae0  main -> main
```

Figure 4.15: Now I commit all the things I did in the repository.




4.16: Verification in the git site.

**6. Repository link:** *https://github.com/Sora-105/Pastrana_PrelimeExam.git*

Conclusion: In this examination, I learned and also gained more knowledge on how to make a ansible-playbook. I think this enhanced my skills in this course. I hope to learn more in the midterm.