**Lesson 08 Lesson End Project**

**Creating an Apache Server on Ubuntu Using Ansible Roles**

**Project agenda:** To create an Apache server on Ubuntu using Ansible Roles

**Description:** Ansible Roles are a structured way of grouping tasks, handlers, vars, and other properties. They increase reusability. For this project, we will use two Ubuntu machines. The first one will be your Ansible controller and the second one will be your target machine for Apache installation.

**Tools required:** Ansible

**Prerequisites:** You must have Ansible Installed in your system and ensure you can connect to your target machine from your controller through Ansible.

**Expected deliverables:**

Creating Ansible Role

Writing configuration files

Running the Playbook

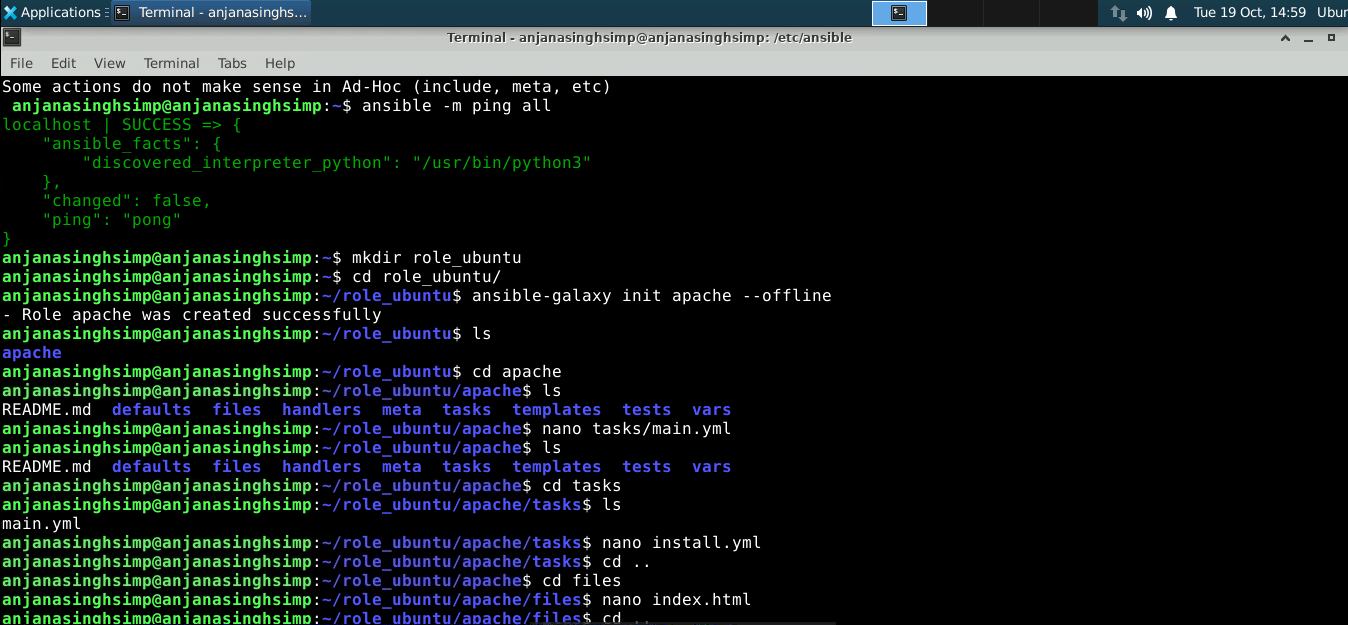
**Steps to be followed:**

1. Checking the connectivity of the target machine from the controller through Ansible
2. Creating a Role under the Role folder
3. Configuring the main components of Role
4. Using the Apache role with site.yml
5. Checking if the YAML files are well-formatted
6. Running the Playbook
7. Verifying the Apache server

**Step 1: Checking the connectivity of the target machine from the controller through Ansible**

* 1. Use the below command to check the connectivity of the target machine from the controller through Ansible:

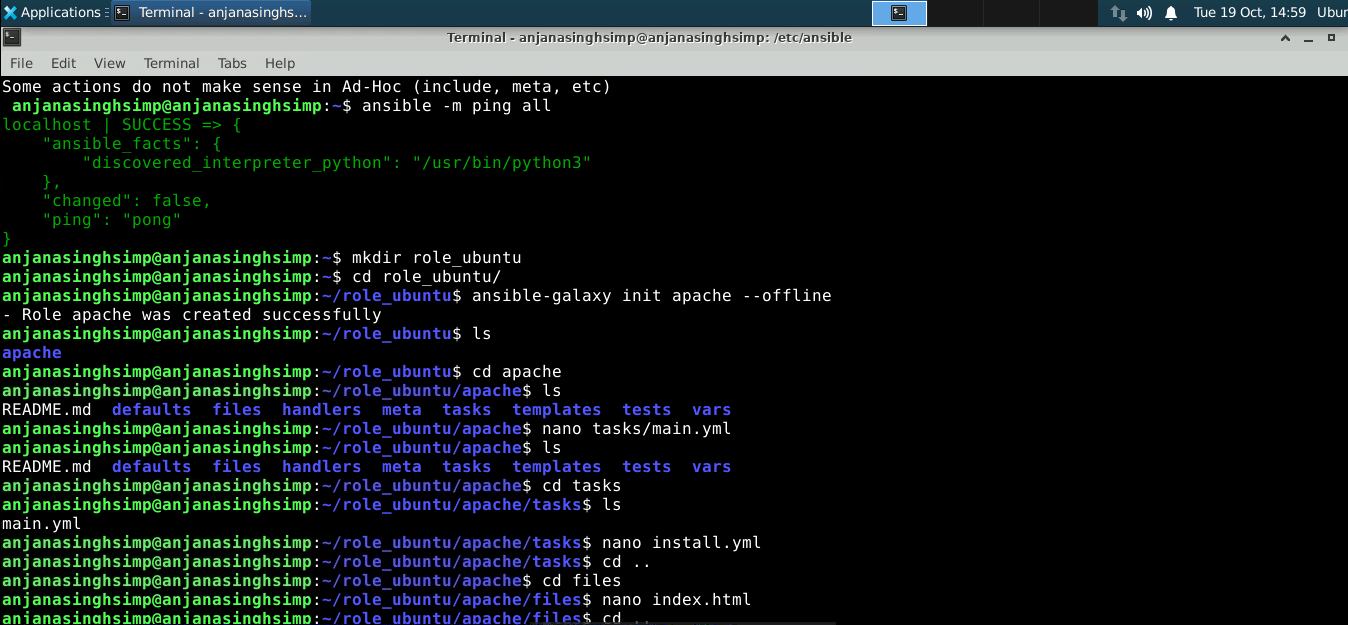
***ansible all -m ping***



**Step 2: Creating a Role under the Role folder**

2.1 In your **/etc/ansible**, there should be a roles folder. Go into the folder and issue the following command:

***ansible-galaxy init apache --offline***



**Step 3: Configuring the main components of Role**

* 1. Here are the main components we will use in this project:

**tasks/main.yml**: It is the starting point of the role tasks. You can use main.yml to point to other task files.

**handlers/main.yml**: It contains the handlers.

**files**: You can keep the files and resources that you want to deploy here.

Let’s start with **tasks/main.yml,** paste the following code inside:

***sudo nano /etc/ansible/roles/tasks main.yml***

**Code:**

---

# tasks file for apache

- include\_tasks: install.yml

- include\_tasks: configure.yml

- include\_tasks: service.yml

We are dividing the tasks into smaller portions and pointing to other YAML files. So, we need to create those files.

* 1. Inside **/etc/ansible/roles/apache/tasks**, let’s create **install.yml** with the following code:

***sudo nano install.yml***

**Code:**

---

# installing apache2

- name: installing apache2 server

apt:

name: apache2

state: present

**Note**: It is installing apache2 on the Apache server. It’s using apt because our target machine is running Ubuntu.

3.3 Create an **index.html** in the **/etc/ansible/roles/apache/files/** folder with the following code:

***sudo nano /etc/ansible/roles/apache/files/ index.html***

**Code:**

<head>

<title>LinuxHint Demo</title>

</head>

<body>

<h1>

Welcome to Earth!

</h1>

<br/><br/><br/>

<p>

</body>

</html>

* 1. Now let’s navigate back to the **/etc/ansible/roles/apache/tasks** folder and create **configure.yml** with the following code:

***sudo nano /etc/ansible/roles/tasks configure.yml***

**Code:**

---

# Configuring apache2

- name: create the webpage index.html

copy: src=index.html dest=/var/www/html/index.html

notify: restart apache service

* 1. The **notify** command requires a handler. So, we go into **/etc/ansible/roles/apache/handlers/main.yml** and enter the following code:

**Code:**

---

# resarting server

- name: restart apache service

service: name=apache2 state=restarted

This code will restart the Apache server.

* 1. Navigate back to the **/etc/ansible/roles/apache/tasks/** folder and create the **service.yml** file with the following code:

***cd /etc/ansible/roles/apache/tasks/***

***sudo nano service.yml***

**Code:**

---

# tasks file for apache

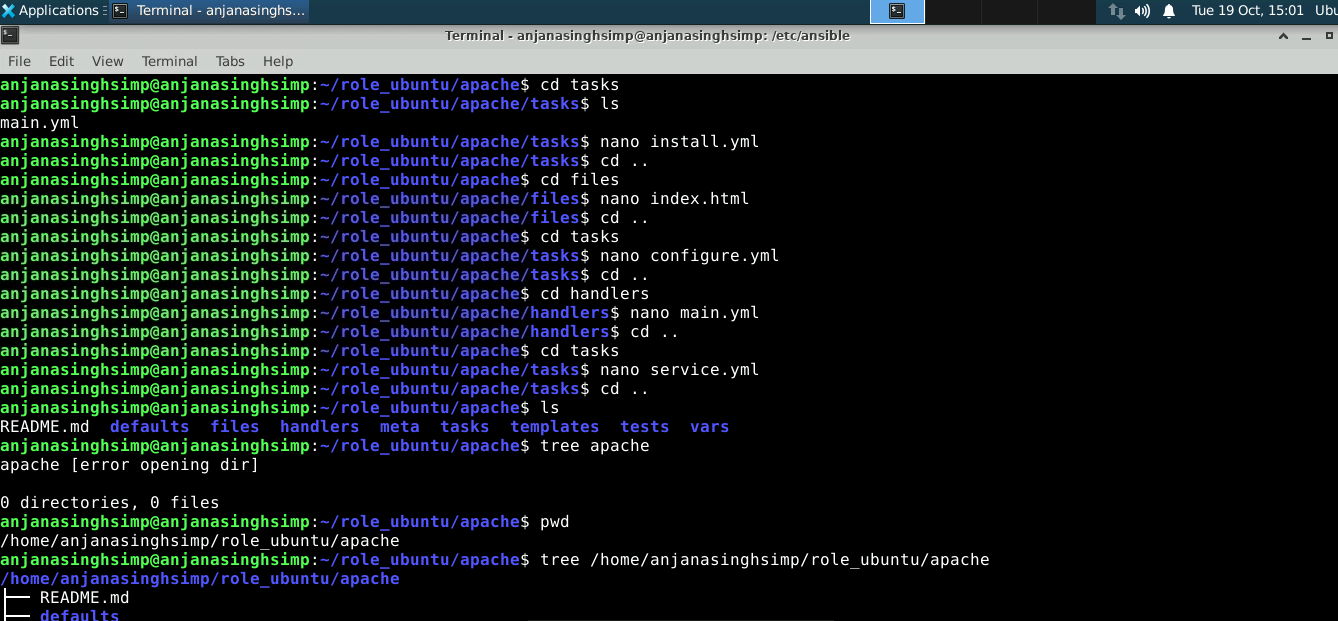
- name: start apache2 server

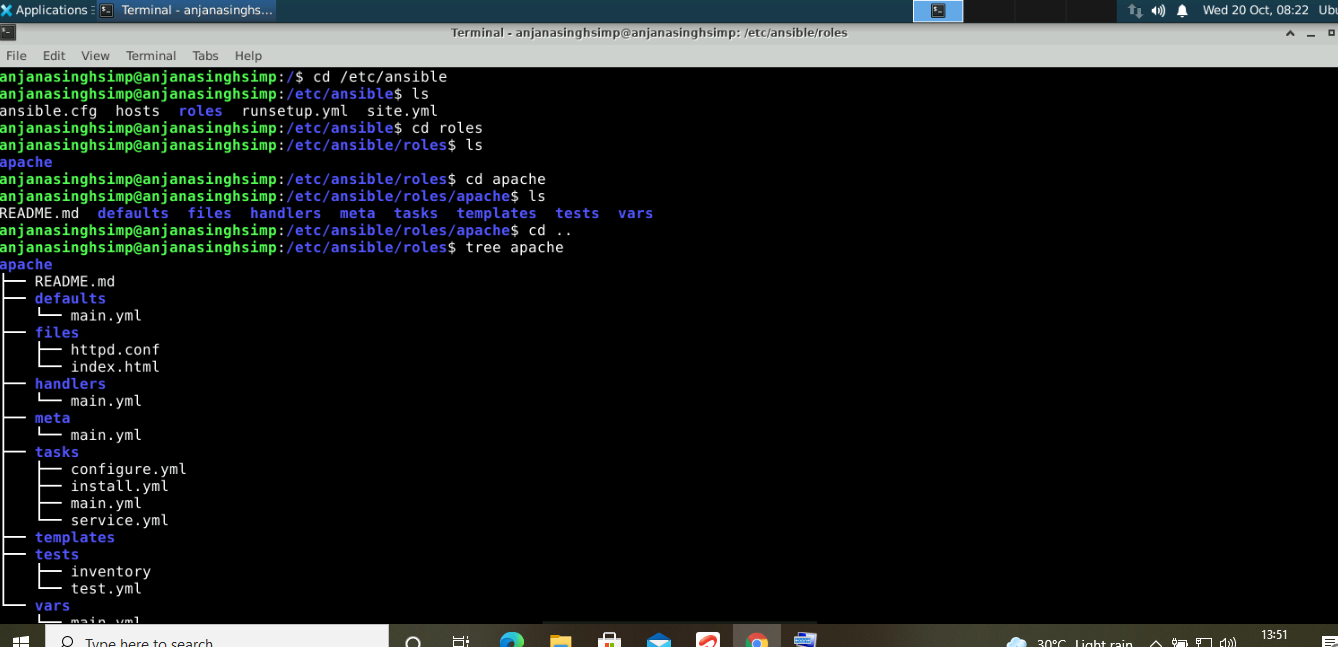
service: name=apache2 state=started

This will start the Apache server.

* 1. We are done with defining the apache role. Our apache folder inside **/etc/ansible/roles** can be viewed using the below command:

**tree /etc/ansible/roles/apache**



****

**Step 4: Using the Apache role with site.yml**

4.1 Now in the folder **/etc/ansible** define the following **site.yml**:

**sudo nano /etct/ansible site.yml**

**Code:**

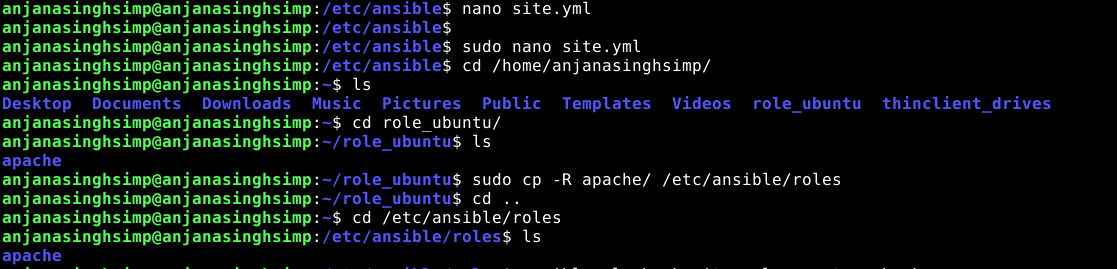
---

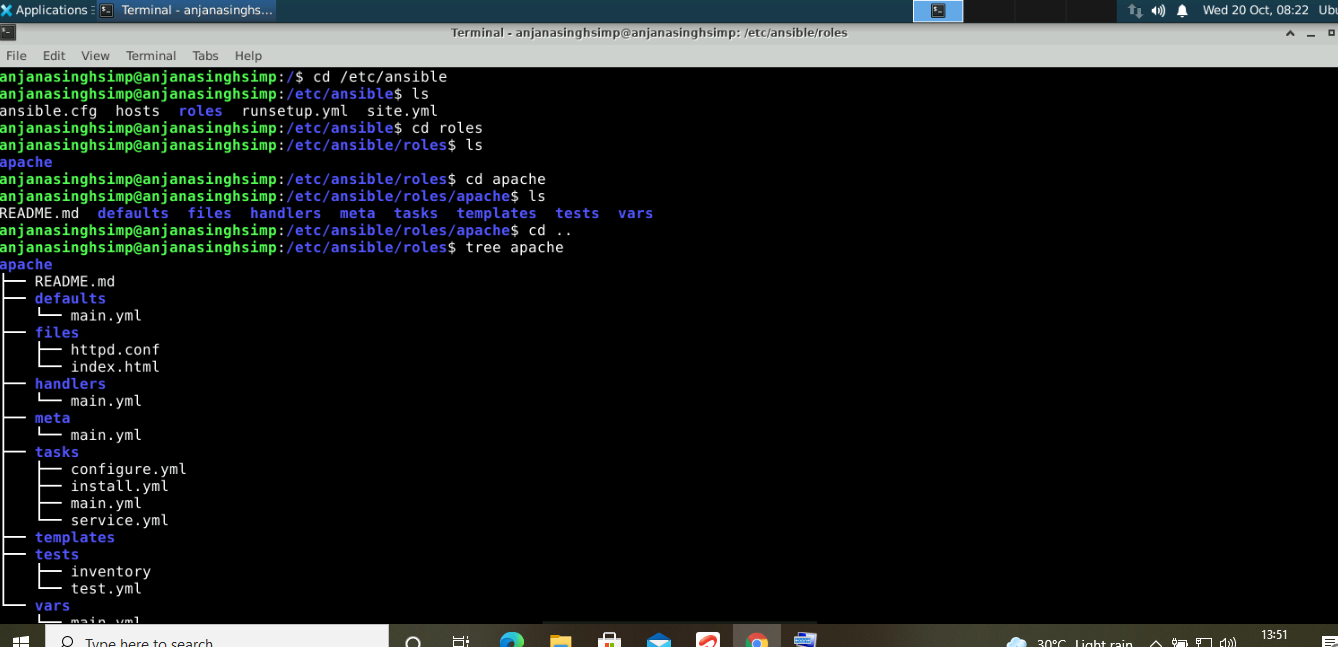
- hosts: myserver1

become: true

roles:

- apache

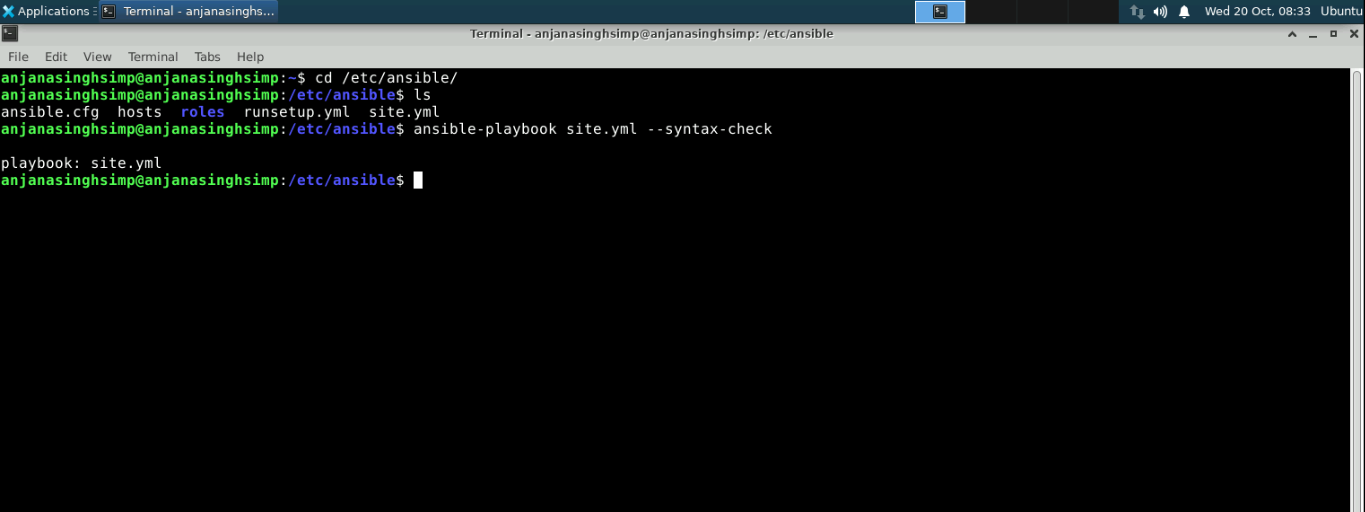




**Step 5: Checking if the YAML files are well-formatted**

5.1 We can check if our YAML files are well-formatted using the following command:

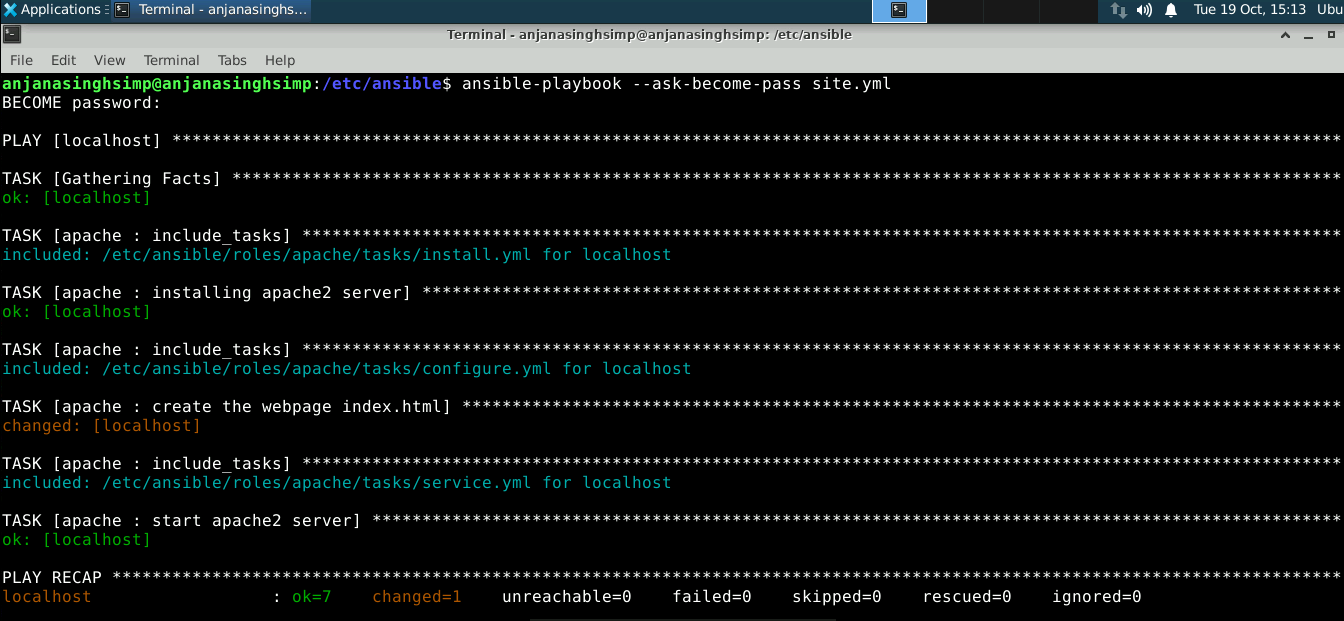
**ansible-playbook site.yml --syntax-check**

****

**Step 6: Running the Playbook**

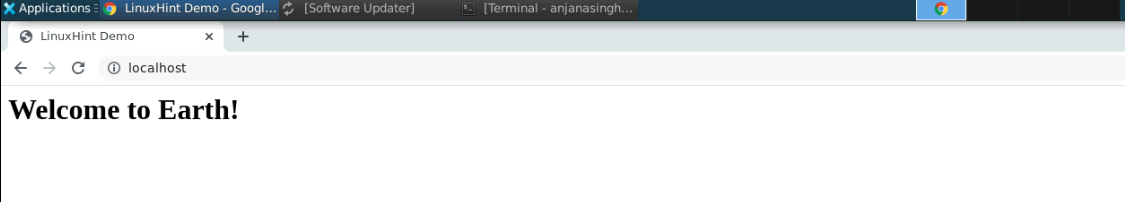
6.1 Run the following command:

***ansible-playbook --ask-become-pass site.yml***

****

**Step 7: Verifying the Apache server**

7.1 If you have port 80 open on your target server, then you should be able to go to **http://localhost** and see something like this:

****