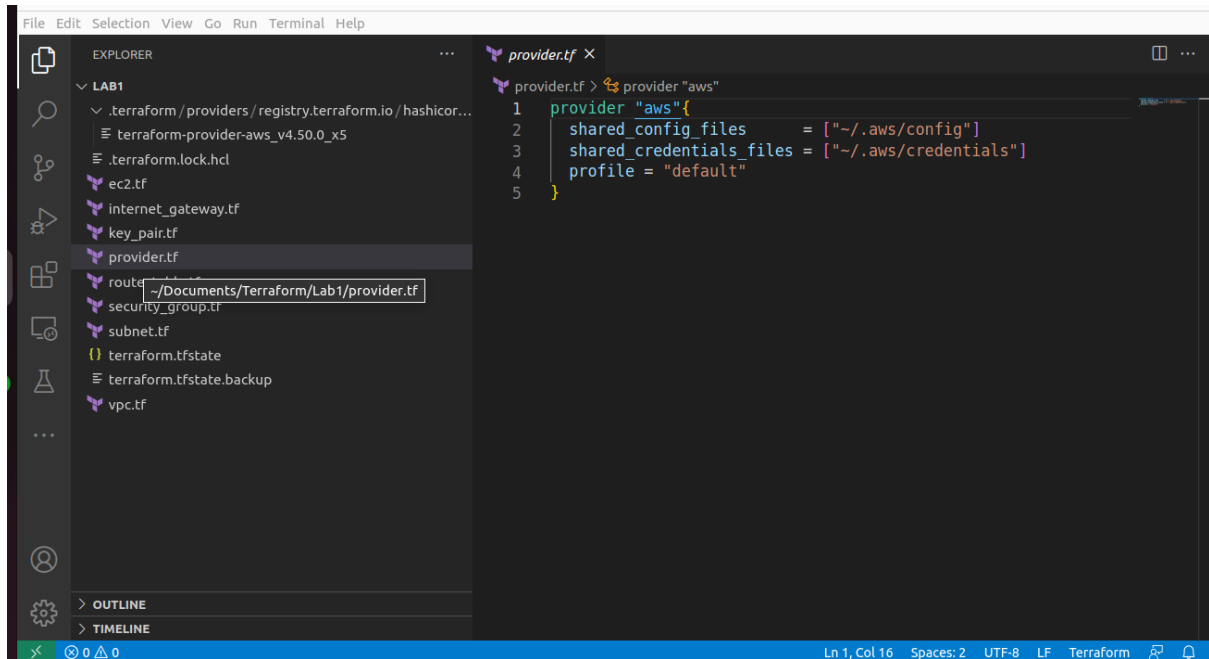


LAB 1



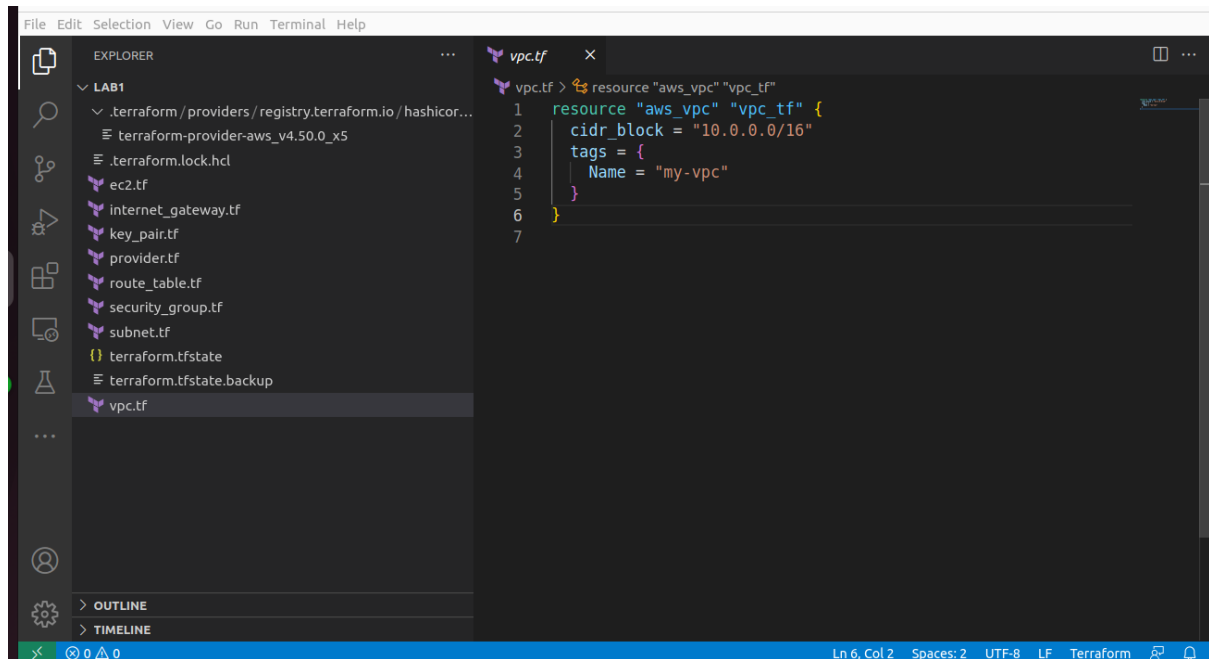
This screenshot shows the Visual Studio Code interface with a Terraform project. The Explorer sidebar on the left displays a directory structure for 'LAB1' containing various Terraform files, including 'provider.tf'. The 'provider.tf' file is selected and its content is shown in the main editor. The code defines the AWS provider with specific configuration files and a default profile.

```
File Edit Selection View Go Run Terminal Help

EXPLORER
  LAB1
    .terraform/providers/registry.terraform.io/hashicor...
    terraform-provider-aws_v4.50.0_x5
    .terraform.lock.hcl
    ec2.tf
    internet_gateway.tf
    key_pair.tf
    provider.tf
    route_table.tf
    security_group.tf
    subnet.tf
    terraform.tfstate
    terraform.tfstate.backup
    vpc.tf

  provider.tf
    provider "aws"
    1 provider "aws"{
    2   shared_config_files = ["~/.aws/config"]
    3   shared_credentials_files = ["~/.aws/credentials"]
    4   profile = "default"
    5 }
```

Ln 1, Col 16 Spaces: 2 UTF-8 LF Terraform



This screenshot shows the Visual Studio Code interface with the same Terraform project. The Explorer sidebar on the left shows the 'vpc.tf' file selected. The main editor displays the content of 'vpc.tf', which defines an AWS VPC resource with a specific CIDR block and a tag.

```
File Edit Selection View Go Run Terminal Help

EXPLORER
  LAB1
    .terraform/providers/registry.terraform.io/hashicor...
    terraform-provider-aws_v4.50.0_x5
    .terraform.lock.hcl
    ec2.tf
    internet_gateway.tf
    key_pair.tf
    provider.tf
    route_table.tf
    security_group.tf
    subnet.tf
    terraform.tfstate
    terraform.tfstate.backup
    vpc.tf

  vpc.tf
    resource "aws_vpc" "vpc_tf"
    1 resource "aws_vpc" "vpc_tf" {
    2   cidr_block = "10.0.0.0/16"
    3   tags = {
    4     Name = "my-vpc"
    5   }
    6 }
    7
```

Ln 6, Col 2 Spaces: 2 UTF-8 LF Terraform

File Edit Selection View Go Run Terminal Help

EXPLORER

LAB1

- .terraform/providers/registry.terraform.io/hashicor...
- terraform-provider-aws_v4.50.0_x5
- .terraform.lock.hcl
- ec2.tf
- internet_gateway.tf
- key_pair.tf
- provider.tf
- route_table.tf
- security_group.tf
- subnet.tf
- terraform.tfstate
- terraform.tfstate.backup
- vpc.tf

~/Documents/Terraform/Lab1/route_table.tf

OUTLINE

TIMELINE

internet_gateway.tf

```
1 resource "aws_internet_gateway" "gw_tf" {
2   vpc_id = "vpc-097ef18d99b03a966"
3
4   tags = {
5     Name = "my_internet_gateway"
6   }
7 }
```

Ln 2, Col 35 Spaces: 2 UTF-8 LF Terraform

File Edit Selection View Go Run Terminal Help

EXPLORER

LAB1

- .terraform/providers/registry.terraform.io/hashicor...
- terraform-provider-aws_v4.50.0_x5
- .terraform.lock.hcl
- ec2.tf
- internet_gateway.tf
- key_pair.tf
- provider.tf
- route_table.tf
- security_group.tf
- subnet.tf
- terraform.tfstate
- terraform.tfstate.backup
- vpc.tf

OUTLINE

TIMELINE

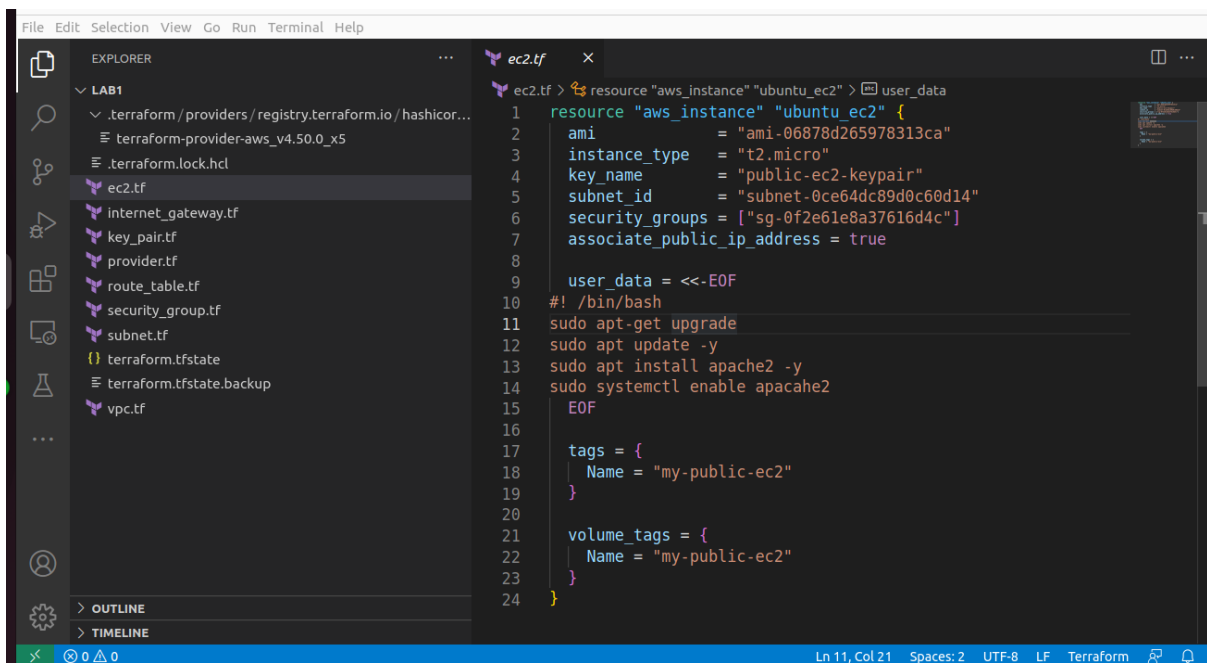
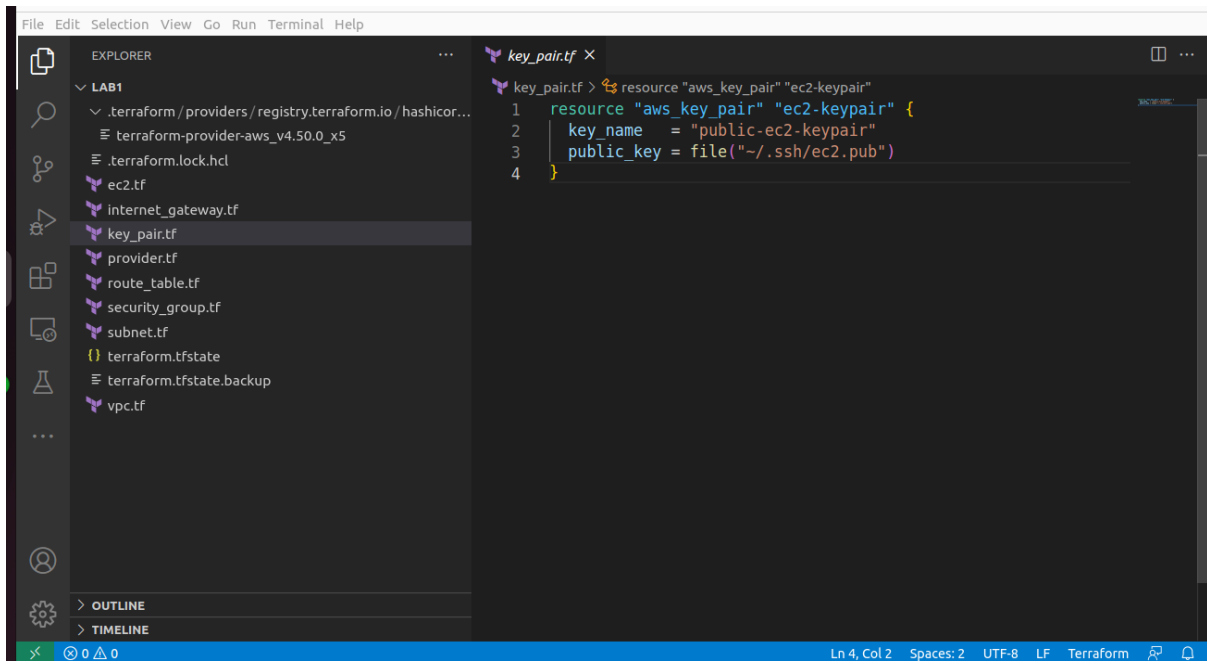
subnet.tf

```
1 resource "aws_subnet" "public_subnet_tf" {
2   vpc_id = "vpc-097ef18d99b03a966"
3   cidr_block = "10.0.0.0/24"
4
5   tags = {
6     Name = "my-public-subnet"
7   }
8 }
```

Ln 6, Col 29 Spaces: 2 UTF-8 LF Terraform

```
1 resource "aws_route_table" "route_table_tf" {
2   vpc_id = "vpc-097ef18d99b03a966"
3
4   route {
5     cidr_block = "0.0.0.0/0"
6     gateway_id = "igw-0c54f8036fb82821d"
7   }
8
9   # route {
10  #   ipv6_cidr_block = ":::/0"
11  #   egress_only_gateway_id = aws_egress_only_internet_gateway_id
12  # }
13
14   tags = {
15     Name = "my_route_table"
16   }
17 }
18
19 resource "aws_route_table_association" "rt_association" {
20   subnet_id = "subnet-0ce64dc89d0c60d14"
21   route_table_id = "rtb-0332c292b815ac73d"
22 }
```

```
1 resource "aws_security_group" "sg" {
2   name = "allow_ssh_http"
3   description = "Allow ssh http inbound traffic"
4   vpc_id = "vpc-097ef18d99b03a966"
5
6
7   ingress {
8     description = "SSH from VPC"
9     from_port = 22
10    to_port = 22
11    protocol = "tcp"
12    cidr_blocks = ["0.0.0.0/0"]
13    ipv6_cidr_blocks = [":::/0"]
14  }
15
16   ingress {
17     description = "HTTP from VPC"
18     from_port = 80
19     to_port = 80
20     protocol = "tcp"
21     cidr_blocks = ["0.0.0.0/0"]
22     ipv6_cidr_blocks = [":::/0"]
23   }
24
25   egress {
26     from_port = 0
```



← → ↻ ⚠ Not secure | 34.227.18.19

Google

Gmail

Outlook

Facebook

YouTube

Google utilities


GeeksforGeeks

Codeforces

Teams

LinkedIn

» | Other bookmarks



Apache2 Default Page

Ubuntu

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in** [/usr/share/doc/apache2/README.Debian.gz](#). Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf  
|-- mods-enabled  
|   |-- *.load  
|   |-- *.conf  
|-- conf-enabled  
|   |-- *.conf  
|-- sites-enabled  
|   |-- *.conf
```