ASSIGNMENT ON TERRAFORM

1. Create a VPC with one public subnet, one private subnet. Create variable files to set CIDR range, vpc name, subnet names, subnet CIDR range etc. Create an EC2 instance on the public subnet of the above VPC, and install an apache webserver on the EC2 instance. Create the above infrastructure using terraform scripts.

Sol:- Steps to create apache on ec2 instance

Step1: Create directory in the following extension file

```
create-ec2-instance/
    - providers.tf
    - main.tf
    - variables.tf
    - install.sh
    - output.tf
    - .gitignore
```

Step2: Creating a Git repository with these files. If you do so, you should start with t this **.gitignore** content:

```
# Compiled files
*.tfstate
*.tfstate.backup

# Module directory
.terraform/

# Sensitive Files
/variables.tf
```

Step3: Install instance **userdata** a shell script **install.sh** which contains commands to install an A
Apache Server:

xample of install.sh:

```
#!/bin/sh
apt-get update
apt-get install -y apache2
service start apache2
chkonfig apache2 on
echo "<html><h1>Welcome to Aapache Web Server</h2></html>" > /var/www/html/i
```

lutput.tf

Step4: navigate the directory and run the following: terraform init

Step5: run the apply command and see the ec2 instance deploying: terraform apply

Step6: When you're done experimenting with Terraform, it's a good idea to remove all the r
resources you created so AWS doesn't charge you for them.

Command: terraform destroy

terraform assignment.zip