Step 1-: Configure aws

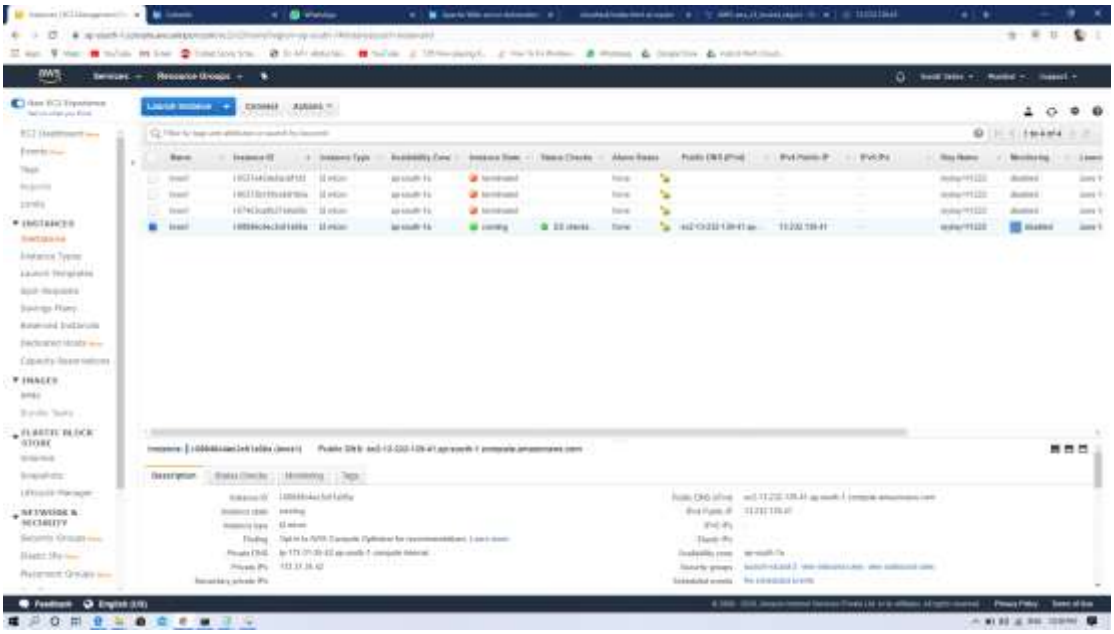
```
Microsoft Windows [Version 10.0.18363.900]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Dell>cd C:\Users\Dell\OneDrive\Desktop\tera\mytest

C:\Users\Dell\OneDrive\Desktop\tera\mytest>aws configure --profile sumit
AWS Access Key ID [****************H5MB]:
AWS Secret Access Key [****************AtHP]:
Default region name [ap-south-1]:
Default output format [json]:

C:\Users\Dell\OneDrive\Desktop\tera\mytest>terraform apply -auto-approve
```

Step 2-: Open aws profile

```
provider "aws" {
  region = "ap-south-1"
  profile = "sumit"
}
```

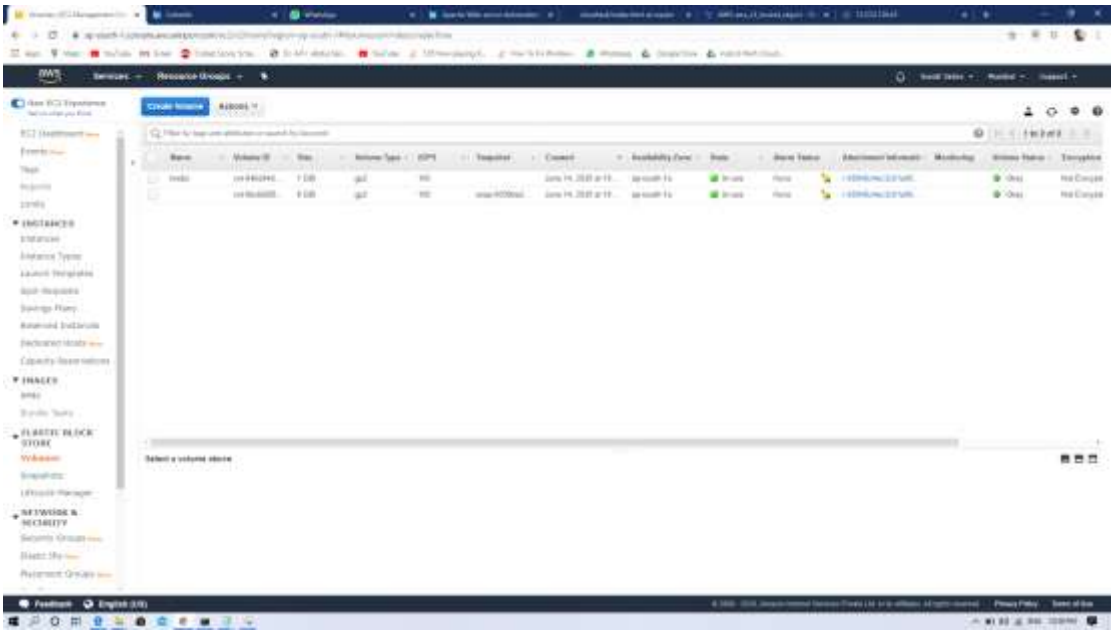
Step 3-: Create and connect to the instance

```
resource "aws_instance" "web" {
  ami           = "ami-0447a12f28fddb066"
  instance_type = "t2.micro"
  key_name = "mykey111222"
  security_groups = [ "launch-wizard-2" ]

  connection {
    type     = "ssh"
    user     = "ec2-user"
    private_key = file("C:/Users/Dell/Downloads/mykey111222.pem")
    host     = aws_instance.web.public_ip
  }

  provisioner "remote-exec" {
    inline = [
      "sudo yum install httpd  php git -y",
      "sudo systemctl restart httpd",
      "sudo systemctl enable httpd",
    ]
  }

  tags = {
    Name = "lwos1"
  }

}
```



Step 4-:Create and attach volume

```
resource "aws_ebs_volume" "ebs2" {
  availability_zone = aws_instance.web.availability_zone
  size              = 1
  tags = {
    Name = "lwebs"
  }
}


resource "aws_volume_attachment" "ebs_att" {
  device_name = "/dev/sdh"
  volume_id   = aws_ebs_volume.ebs2.id
  instance_id = aws_instance.web.id
  force_detach = true
}
```



Step 5-: Connect and mount the volume

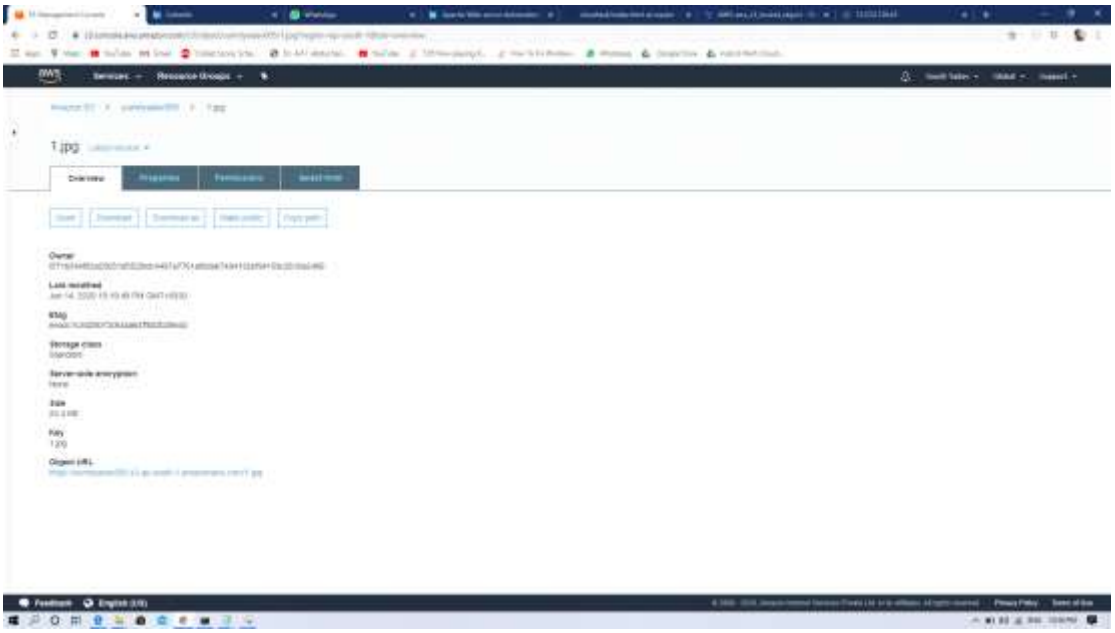
```
resource "null_resource" "nullremote3"  {

depends_on = [
    aws_volume_attachment.ebs_att,
  ]


  connection {
    type     = "ssh"
    user     = "ec2-user"
    private_key = file("C:/Users/Dell/Downloads/mykey111222.pem")
    host     = aws_instance.web.public_ip
  }

provisioner "remote-exec" {
    inline = [
      "sudo mkfs.ext4  /dev/xvdh",
      "sudo mount  /dev/xvdh  /var/www/html",
      "sudo rm -rf /var/www/html/*",
      "sudo git clone https://github.com/Sssyadav123/cloudtask.git  /var/www/html/"
    ]
  }
}
```

Step 6-: Create the bucket

```
resource "aws_s3_bucket" "sumitbucket" {
        bucket = "sumityadav005"
        acl    = "private"

        tags = {
          Name        = "sumityadav005"
        }
      }
```

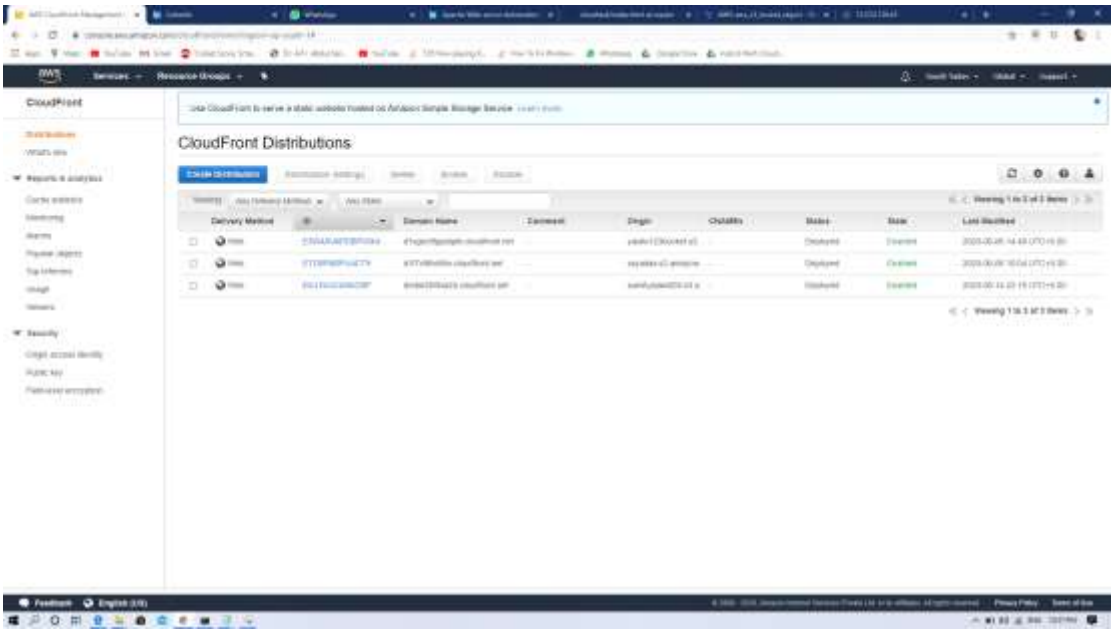
Step 7-: Upload the image on bucket

```
       locals {
          s3_origin_id = "myS3Origin"
        }
resource "aws_s3_bucket_object" "object" {
          bucket = aws_s3_bucket.sumitbucket.id
          key    = "1.jpg"
          source = "C:/Users/Dell/OneDrive/Desktop/tera/mytest/1.jpg"
          acl    = "public-read"
        }
```

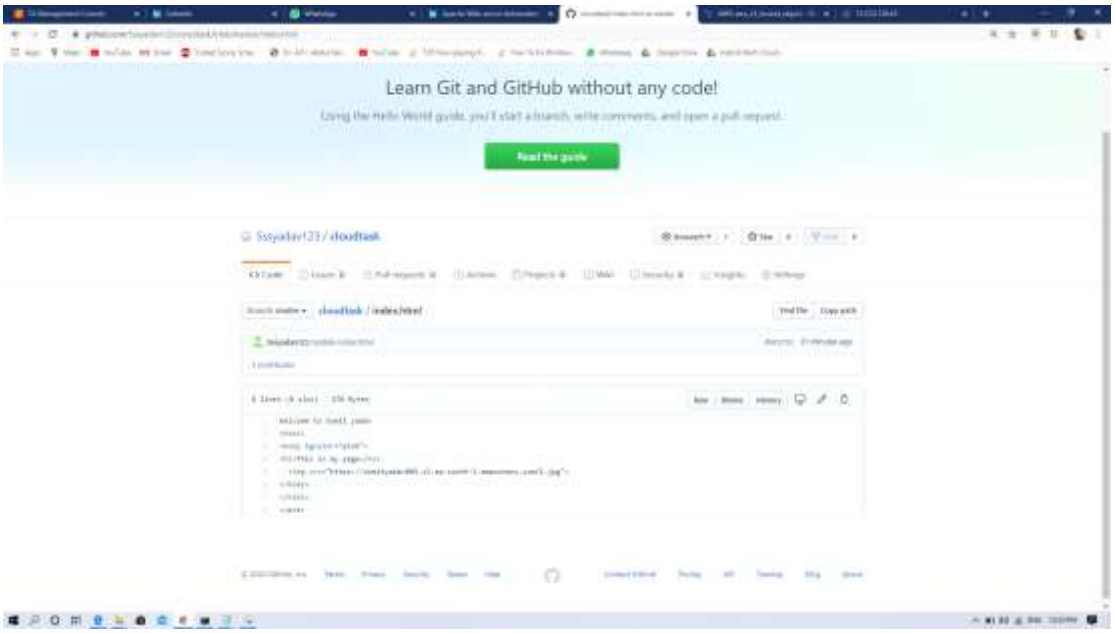


Step 8-: Create the cloud front

```
        locals {
           s3_origin_id = "myS3Origin"
         }
resource "aws_s3_bucket_object" "object" {
           bucket = aws_s3_bucket.sumitbucket.id
           key    = "1.jpg"
           source = "C:/Users/Dell/OneDrive/Desktop/tera/mytest/1.jpg"
           acl    = "public-read"
         }
```



Step 8-: Create the file in the github



Step 9-: Final Step

```
output "myos_ip" {
  value = aws_instance.web.public_ip
}


resource "null_resource" "nulllocal2"  {
        provisioner "local-exec" {
            command = "echo  ${aws_instance.web.public_ip} > publicip.txt"
        }
}
```