

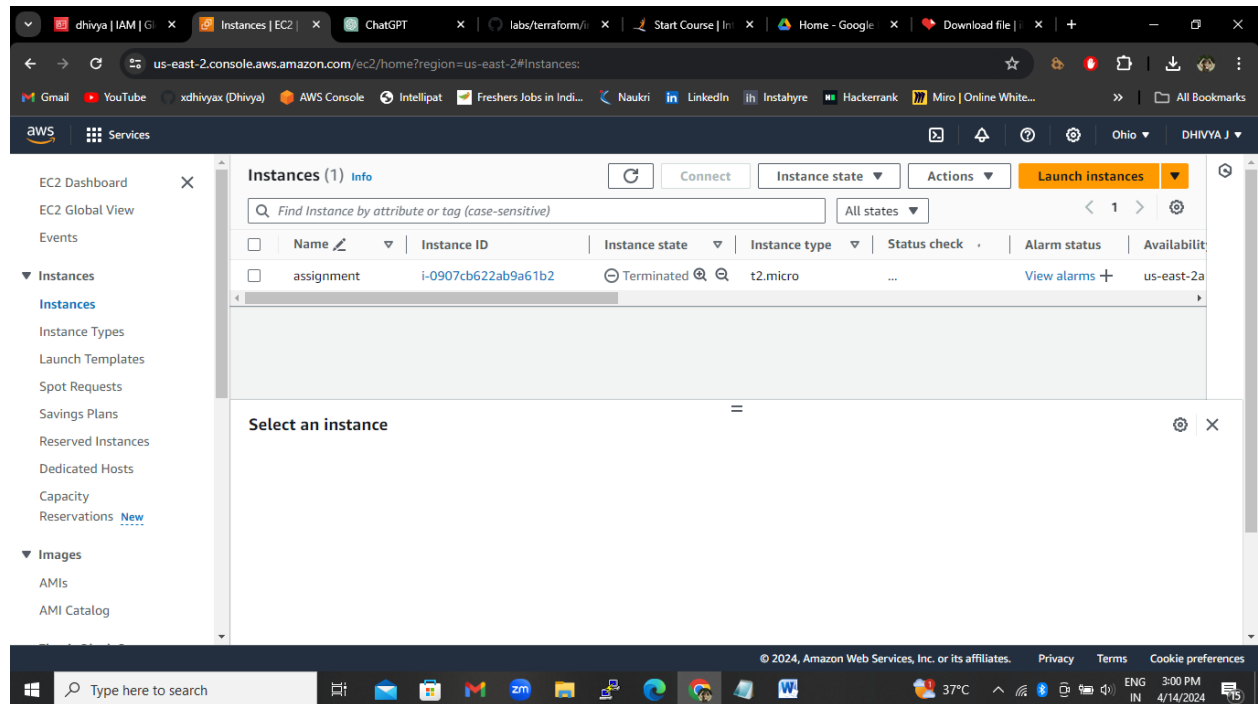
## Terraform Assignment - 2

You have been asked to:

- Destroy the previous deployment
- Create a new EC2 instance with an Elastic IP

```
ubuntu@ip-172-31-95-109: ~  
ubuntu@ip-172-31-95-109:~$ terraform destroy  
aws_instance.assignment: Refreshing state... [id=i-0907cb622ab9a61b2]  
  
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:  
- destroy  
  
Terraform will perform the following actions:  
  
# aws_instance.assignment will be destroyed  
- resource "aws_instance" "assignment" {  
  - ami                        = "ami-0b8b44ec9a8f90422" -> null  
  - arn                        = "arn:aws:ec2:us-east-2:917788167444:instance/i-0907cb622ab9a61b2" -> null  
  - associate_public_ip_address = true -> null  
  - availability_zone           = "us-east-2a" -> null  
  - cpu_core_count              = 1 -> null  
  - cpu_threads_per_core        = 1 -> null  
  - disable_api_stop            = false -> null  
  - disable_api_termination     = false -> null  
  - ebs_optimized               = false -> null  
  - get_password_data           = false -> null  
  - hibernation                 = false -> null  
  - id                          = "i-0907cb622ab9a61b2" -> null  
  - instance_initiated_shutdown_behavior = "stop" -> null  
  - instance_state              = "running" -> null  
  - instance_type               = "t2.micro" -> null  
  - ipv6_address_count          = 0 -> null  
  - ipv6_addresses              = [] -> null  
  - monitoring                  = false -> null  
}
```

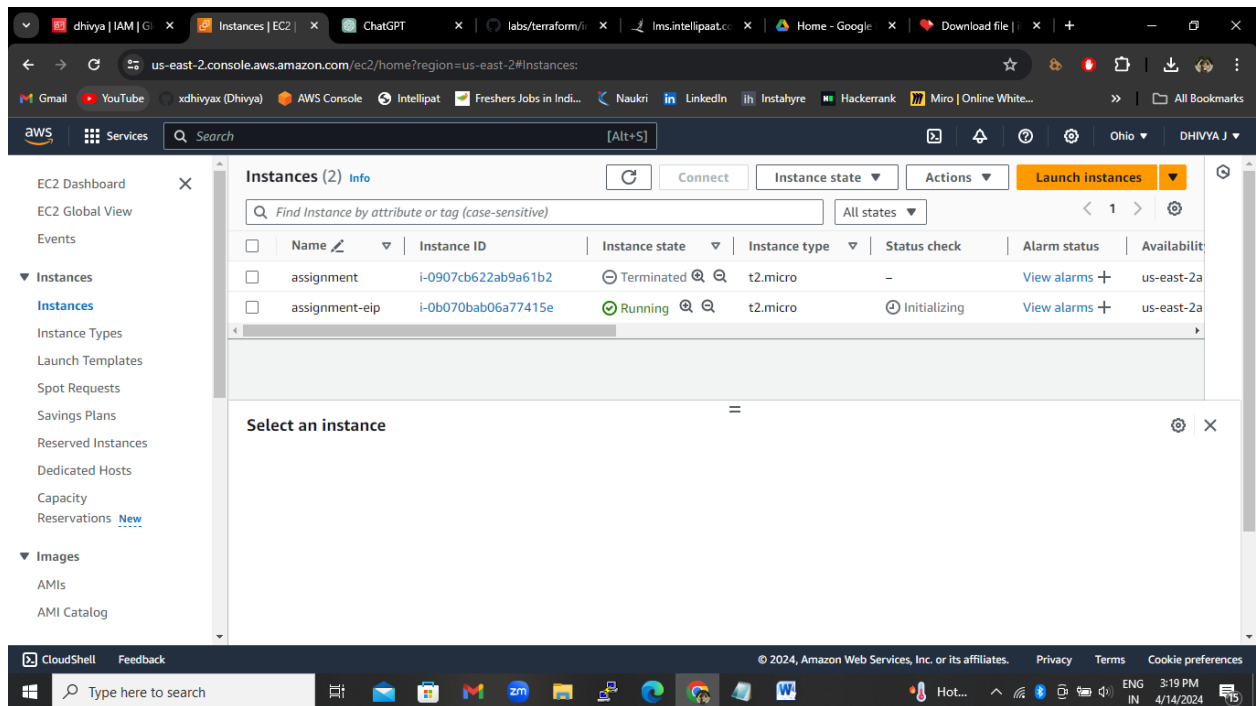
```
ubuntu@ip-172-31-95-109: ~  
- device_name          = "/dev/sda1" -> null  
- encrypted            = false -> null  
- iops                 = 100 -> null  
- tags                 = {} -> null  
- tags_all             = {} -> null  
- throughput          = 0 -> null  
- volume_id           = "vol-041524740959f5503" -> null  
- volume_size         = 8 -> null  
- volume_type         = "gp2" -> null  
  # (1 unchanged attribute hidden)  
}  
}  
  
Plan: 0 to add, 0 to change, 1 to destroy.  
  
Do you really want to destroy all resources?  
Terraform will destroy all your managed infrastructure, as shown above.  
There is no undo. Only 'yes' will be accepted to confirm.  
  
Enter a value: yes  
  
aws_instance.assignment: Destroying... [id=i-0907cb622ab9a61b2]  
aws_instance.assignment: Still destroying... [id=i-0907cb622ab9a61b2, 10s elapsed]  
aws_instance.assignment: Still destroying... [id=i-0907cb622ab9a61b2, 20s elapsed]  
aws_instance.assignment: Still destroying... [id=i-0907cb622ab9a61b2, 30s elapsed]  
aws_instance.assignment: Still destroying... [id=i-0907cb622ab9a61b2, 40s elapsed]  
aws_instance.assignment: Destruction complete after 40s  
  
Destroy complete! Resources: 1 destroyed.  
ubuntu@ip-172-31-95-109:~$
```



```
ubuntu@ip-172-31-95-109: ~  
ubuntu@ip-172-31-95-109:~$ terraform init  
  
Initializing the backend...  
  
Initializing provider plugins...  
- Reusing previous version of hashicorp/aws from the dependency lock file  
- Using previously-installed hashicorp/aws v5.45.0  
  
Terraform has been successfully initialized!  
  
You may now begin working with Terraform. Try running "terraform plan" to see  
any changes that are required for your infrastructure. All Terraform commands  
should now work.  
  
If you ever set or change modules or backend configuration for Terraform,  
rerun this command to reinitialize your working directory. If you forget, other  
commands will detect it and remind you to do so if necessary.  
ubuntu@ip-172-31-95-109:~$
```

```
ubuntu@ip-172-31-95-109: ~  
ubuntu@ip-172-31-95-109:~$ vi ec2.tf  
ubuntu@ip-172-31-95-109:~$ terraform apply  
  
Terraform used the selected providers to generate the following execution plan. Resource actions are  
indicated with the following symbols:  
+ create  
  
Terraform will perform the following actions:  
  
# aws_eip.example will be created  
+ resource "aws_eip" "example" {  
  + allocation_id      = (known after apply)  
  + association_id     = (known after apply)  
  + carrier_ip         = (known after apply)  
  + customer_owned_ip  = (known after apply)  
  + domain             = (known after apply)  
  + id                 = (known after apply)  
  + instance           = (known after apply)  
  + network_border_group = (known after apply)  
  + network_interface  = (known after apply)  
  + private_dns        = (known after apply)  
  + private_ip         = (known after apply)  
  + public_dns         = (known after apply)  
  + public_ip          = (known after apply)  
  + public_ipv4_pool   = (known after apply)  
  + tags               = {  
    + "Name" = "example-eip"  
  }  
  + tags_all           = {  
    + "Name" = "example-eip"  
  }  
}
```

```
ubuntu@ip-172-31-95-109: ~  
+ "Name" = "assignment-eip"  
+ tags all = {  
+ "Name" = "assignment-eip"  
+ }  
+ tenancy = (known after apply)  
+ user_data = (known after apply)  
+ user_data_base64 = (known after apply)  
+ user_data_replace_on_change = false  
+ vpc_security_group_ids = (known after apply)  
}  
  
Plan: 2 to add, 0 to change, 0 to destroy.  
  
Do you want to perform these actions?  
Terraform will perform the actions described above.  
Only 'yes' will be accepted to approve.  
  
Enter a value: yes  
  
aws_instance.assignment: Creating...  
aws_instance.assignment: Still creating... [10s elapsed]  
aws_instance.assignment: Still creating... [20s elapsed]  
aws_instance.assignment: Still creating... [30s elapsed]  
aws_instance.assignment: Creation complete after 32s [id=i-0b070bab06a77415e]  
aws_eip.example: Creating...  
aws_eip.example: Creation complete after 2s [id=eipalloc-0c83e9d96875cbbb5]  
  
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.  
ubuntu@ip-172-31-95-109:~$
```



us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#Addresses:

Elastic IP addresses (1)

Find resources by attribute or tag

<input type="checkbox"/>	Name	Allocated IPv4 address	Type	Allocation ID	Reverse
<input type="checkbox"/>	example-eip	3.128.68.76	Public IP	eipalloc-0c83e9d96875cbbb5	-

View IP address usage and recommendations to release unused IPs with [Public IP insights](#).

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

37°C 3:19 PM ENG IN 4/14/2024

us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#Instances:

Instances (1/1) Info

Find Instance by attribute or tag (case-sensitive) Running

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
<input checked="" type="checkbox"/>	assignment-eip	i-0b070bab06a77415e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-2a

Instance: i-0b070bab06a77415e (assignment-eip)

Details Status and alarms New Monitoring Security Networking Storage Tags

Instance summary Info

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0b070bab06a77415e (assignment-eip)	3.128.68.76 (example-eip) <a href="#">open address</a>	172.31.6.114
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-3-128-68-76.us-east-2.compute.amazonaws.com <a href="#">open address</a>

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

37°C 3:20 PM ENG IN 4/14/2024

ubuntu@ip-172-31-95-109: ~

```
ubuntu@ip-172-31-95-109:~$ cat ec2.tf
resource "aws_instance" "assignment" {
  ami           = "ami-0b8b44ec9a8f90422"
  instance_type = "t2.micro"
  security_groups = ["default"]

  tags = {
    Name = "assignment-eip"
  }
}

resource "aws_eip" "example" {
  instance = aws_instance.assignment.id
  tags = {
    Name = "example-eip"
  }
}
ubuntu@ip-172-31-95-109:~$
```

