**DEVOPS PROJECT**

Problem Statement - Create an Autoscaling group using Terraform on AWS. The instances in the group should have Nginx installed on them. Parameters in the ASG can be set at your discretion and should be mentioned in the solution documentation.

Any variables in the script should be placed in a separate variables.tf file. The output of the script should be the DNS of the associated load balancer.

1. **Solution Architecture with Template**

ALB

Public Internet

VPC

IGW

AZ 1

AZ 2

Public Subnet1

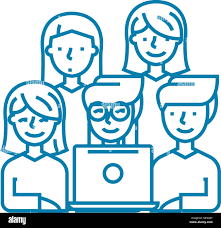
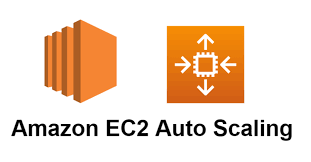
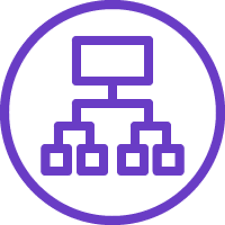
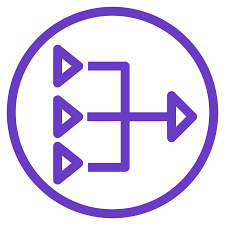
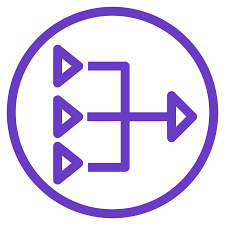
NGW 1

Private Subnet1

Public Subnet2

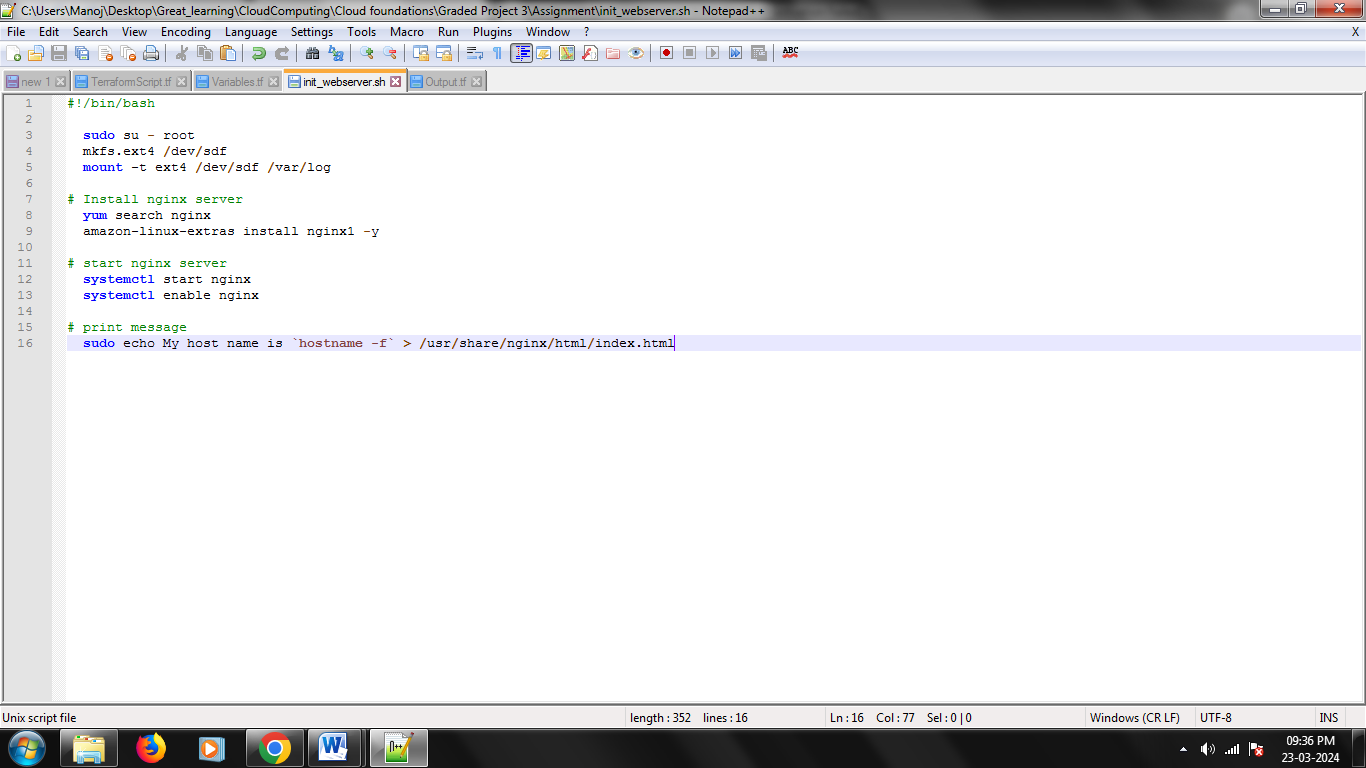
NGW 2

Private Subnet2

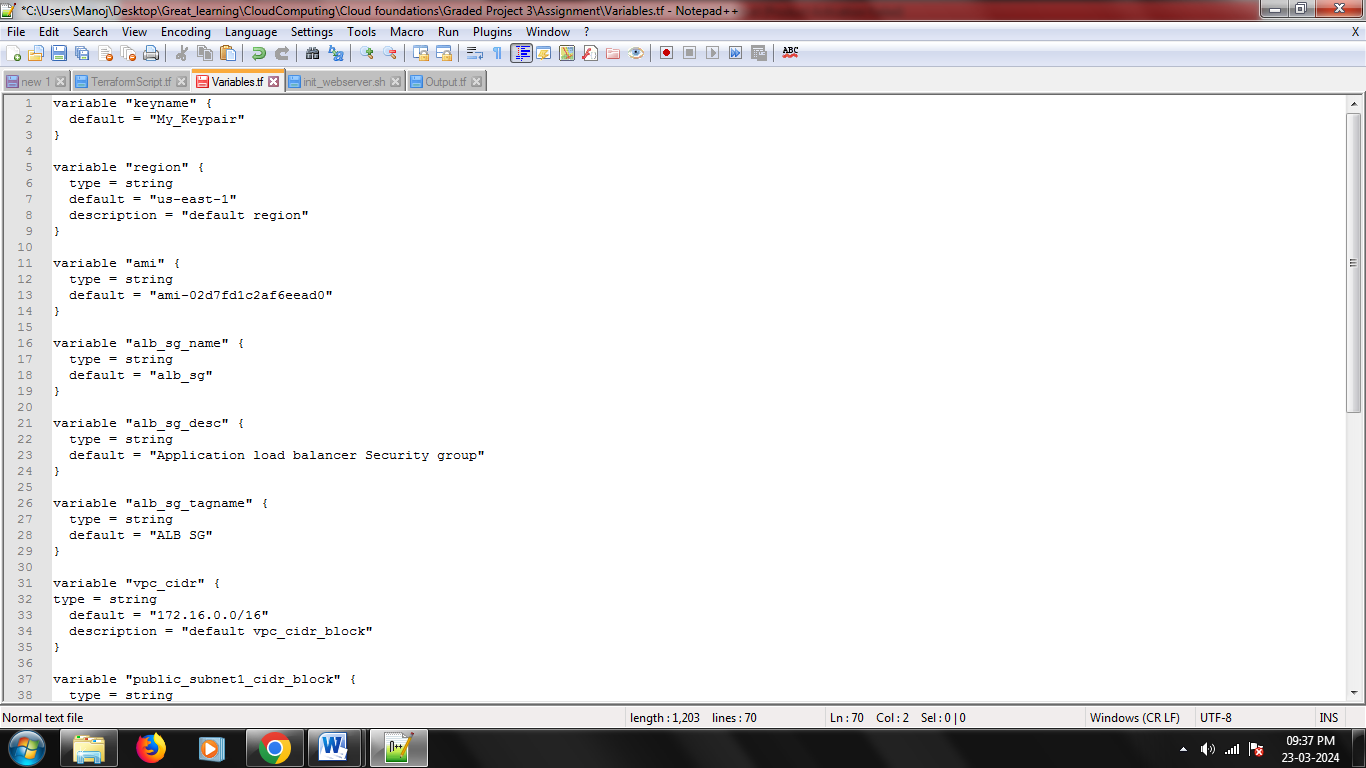


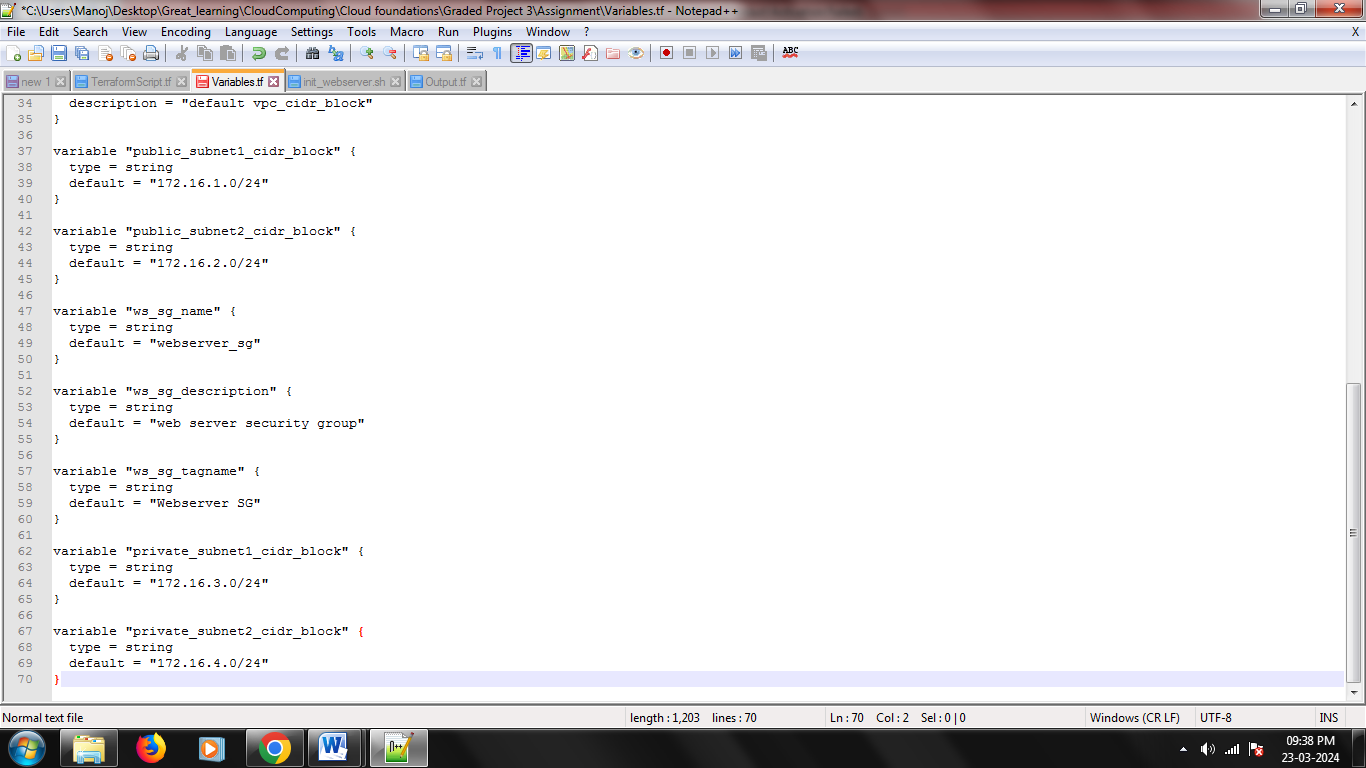
* AWS CLI profile: Create a AWS CLI profile which can be used in TerraformScript.tf file
* Security Groups:
  1. Security group for the application load balancer alb\_sg permitting inbound HTTP traffic from any source and unrestricted outbound traffic.
  2. Security group for the web servers webserver\_sg allowing inbound HTTP and SSH traffic from any source and unrestricted outbound traffic.
* Virtual Private Cloud: Named my-vpc, this VPC has a specified CIDR block determined by the var.vpc\_cidr variable.
* Subnets:
  1. Two public subnets named public\_subnet1 and public\_subnet2 situated in different availability zones named us-east-1a and us-east-1b, tailored for public-facing resources. Their CIDR blocks are defined by variables named public\_subnet1\_cidr\_block & public\_subnet2\_cidr\_block.
  2. Two private subnets named private\_subnet1 and private\_subnet2 named placed in separate availability zones, designed for private resources. Their CIDR blocks are specified by variables private\_subnet1\_cidr\_block & private\_subnet3\_cidr\_block.
* Internet Gateway: internet gateway my-igw, this gateway facilitates internet access for the public subnets.
* NAT Gateways:
  1. Two NAT gateways ngw-1 & ngw-2 deployed in public subnets, serving to grant outbound internet access for private subnets.
* Route Tables:
  1. A public route table public\_subnet1\_rt connected to the public subnets, featuring a route to the internet gateway.
  2. Two private route tables’ private\_subnet1\_rt and private\_subnet2\_rt associated with the private subnets, each containing a route to a corresponding NAT gateway.
* Launch Configuration: Named webserver-launch-config, this configuration defines the characteristics of web server instances such as AMI ID, instance type, key pair, block devices, and user data script.
* Auto Scaling Group: Named my-ASG, this group manages web server instances with specific capacity settings and launches instances across both private subnets, with a desired capacity of 2, a minimum of 1, and a maximum of 4 instances.
* Application Load Balancer: Named my-ALB, this ALB includes listeners on port 80 to evenly distribute traffic among web servers within the Auto Scaling Group.
* Target Group: Named my-TG, this target group encompasses web server instances launched by the Auto Scaling Group and defines a health check mechanism to monitor their operational health.

1. **Shell script used to install Nginx**

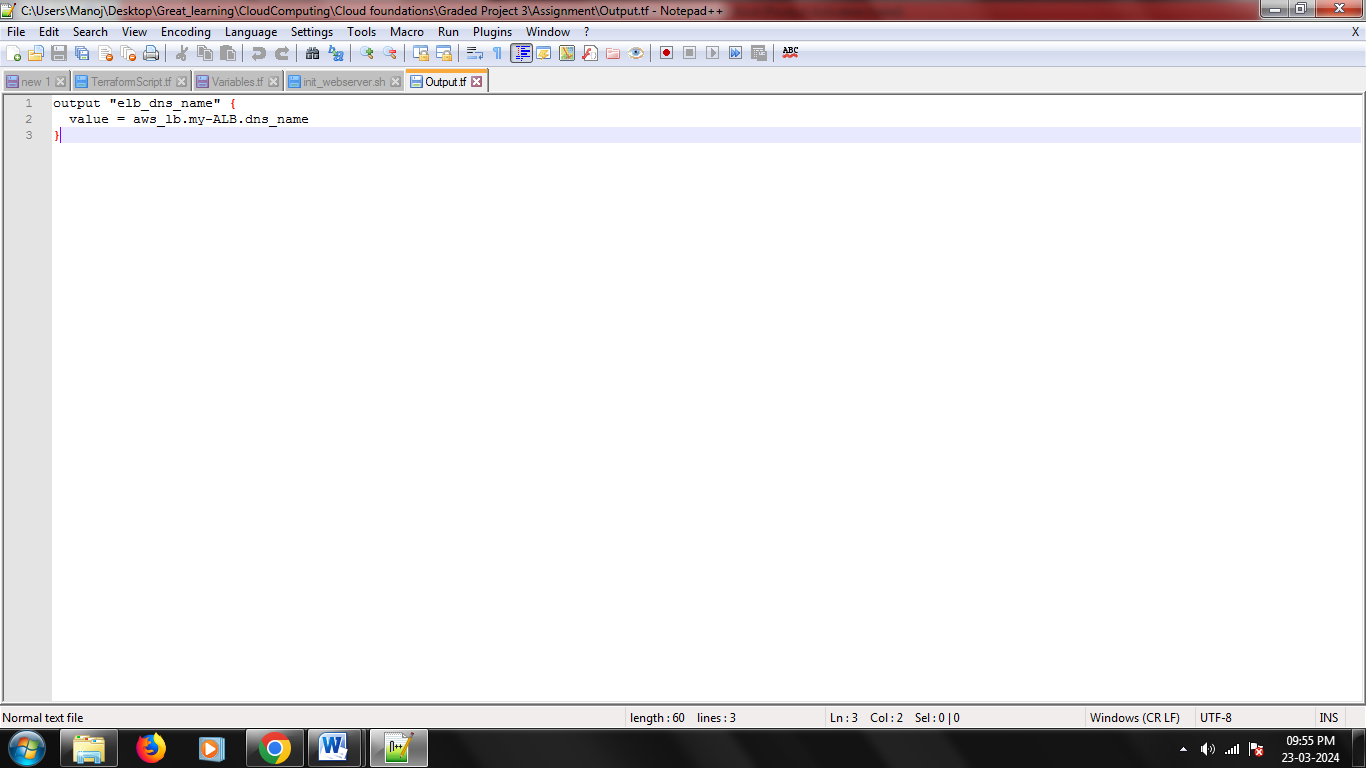


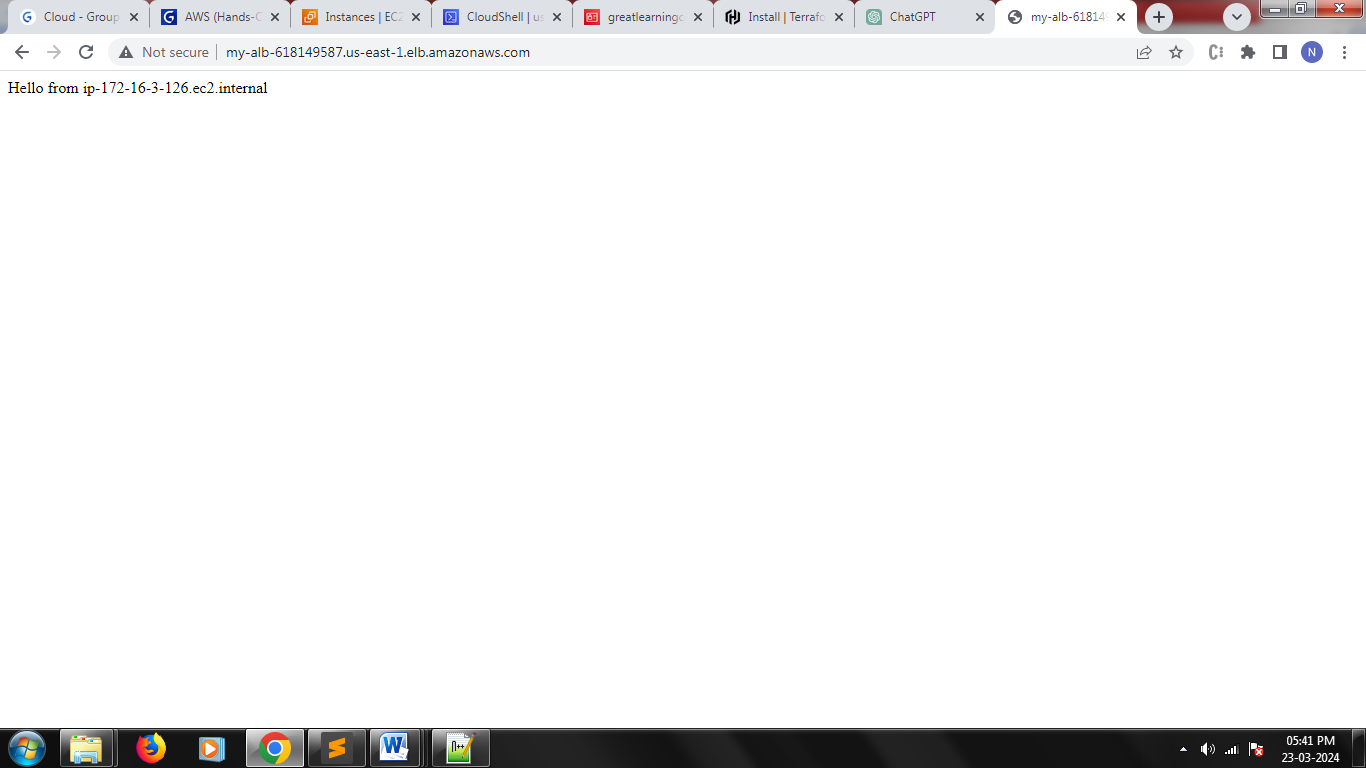
1. **Separate variables file**

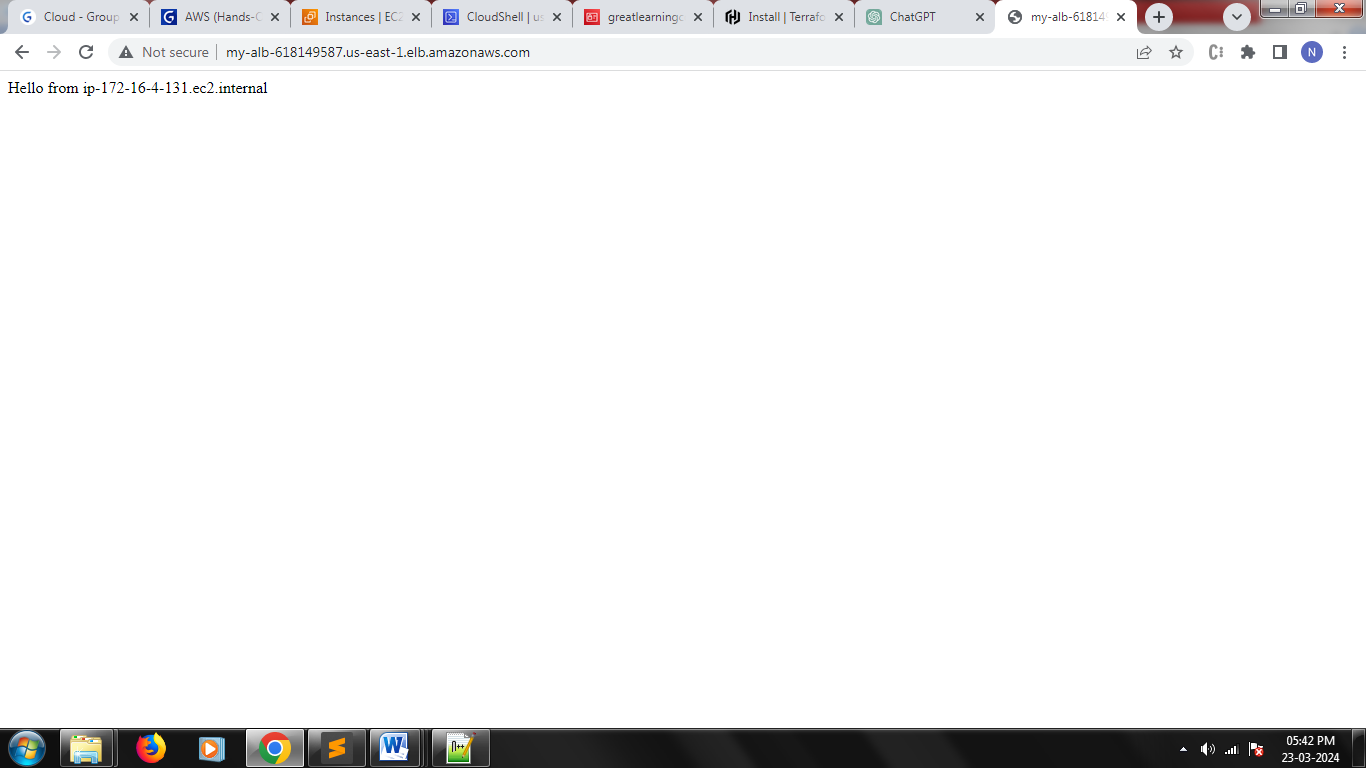




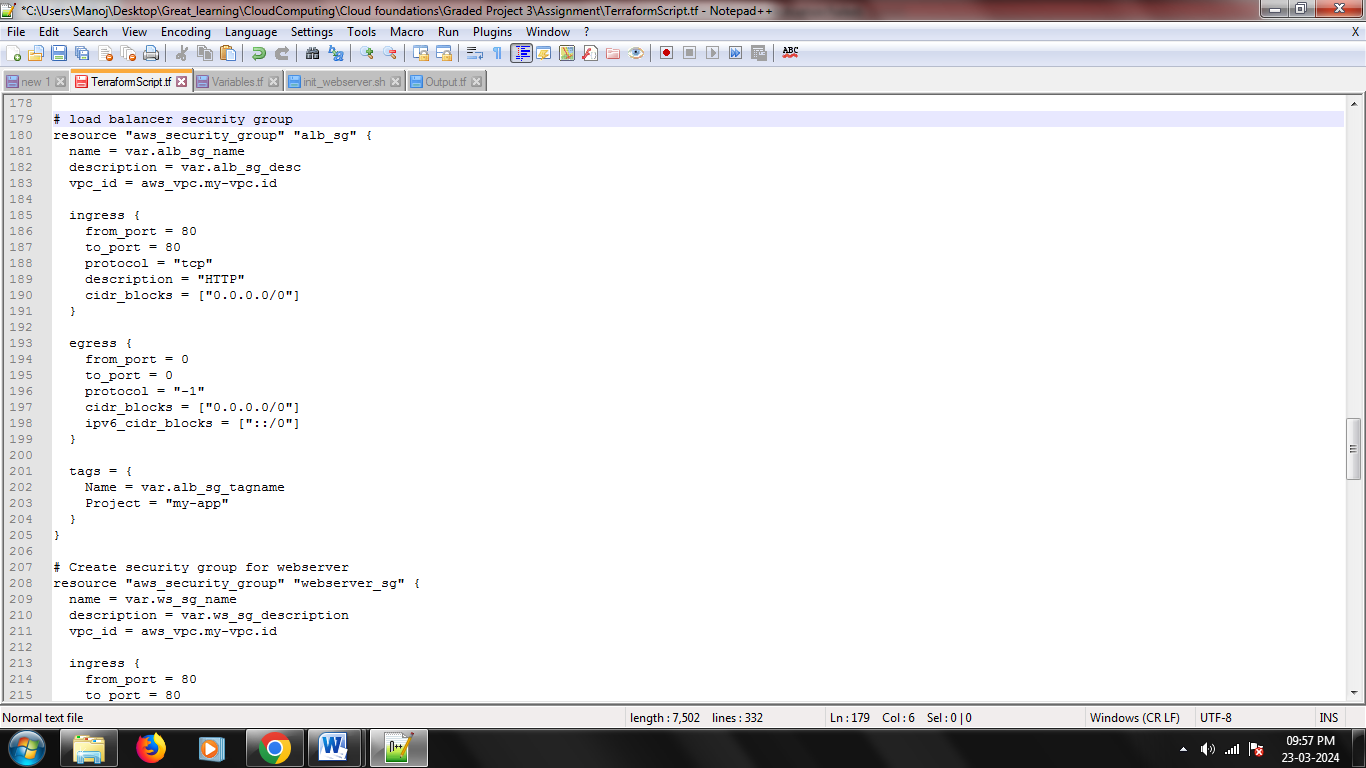
1. **Output file showing DNS of load balancer**

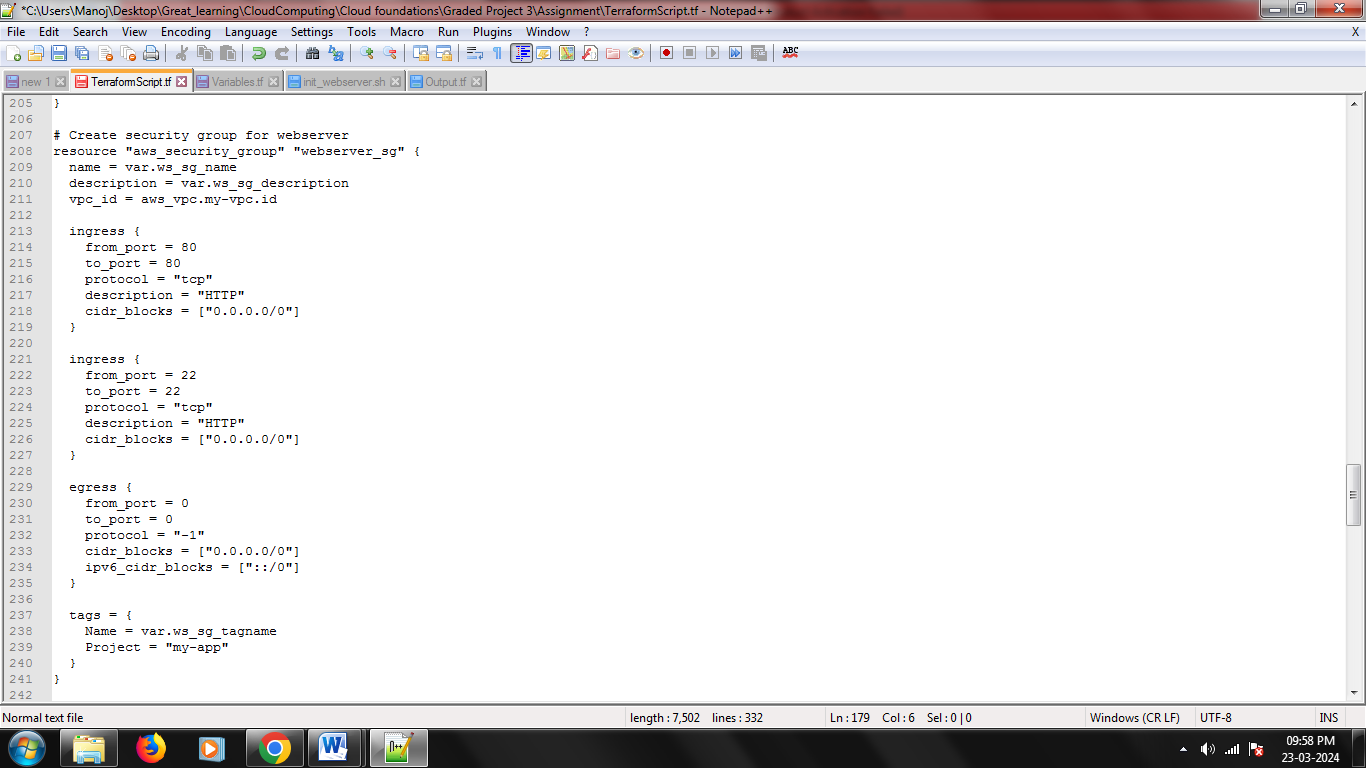




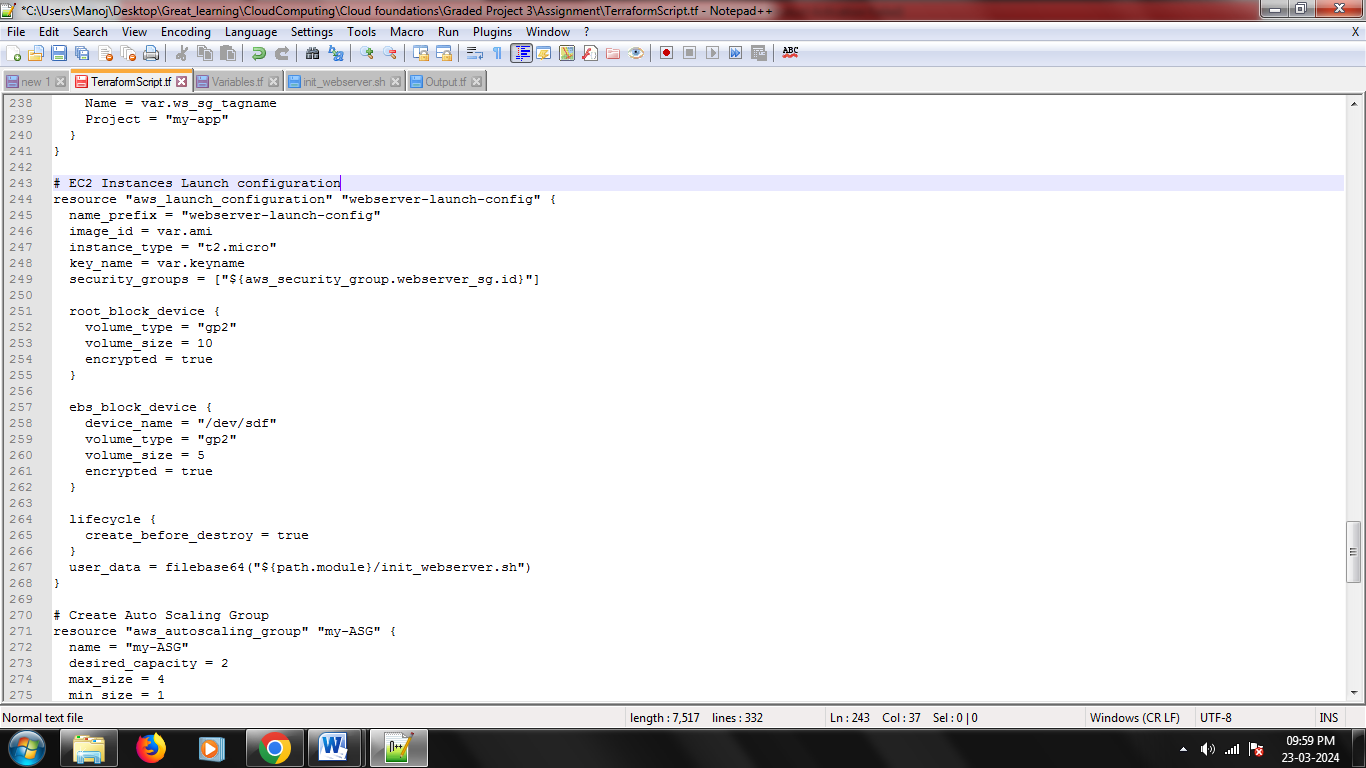


1. **Terraform script with given components**
   1. **Security groups**

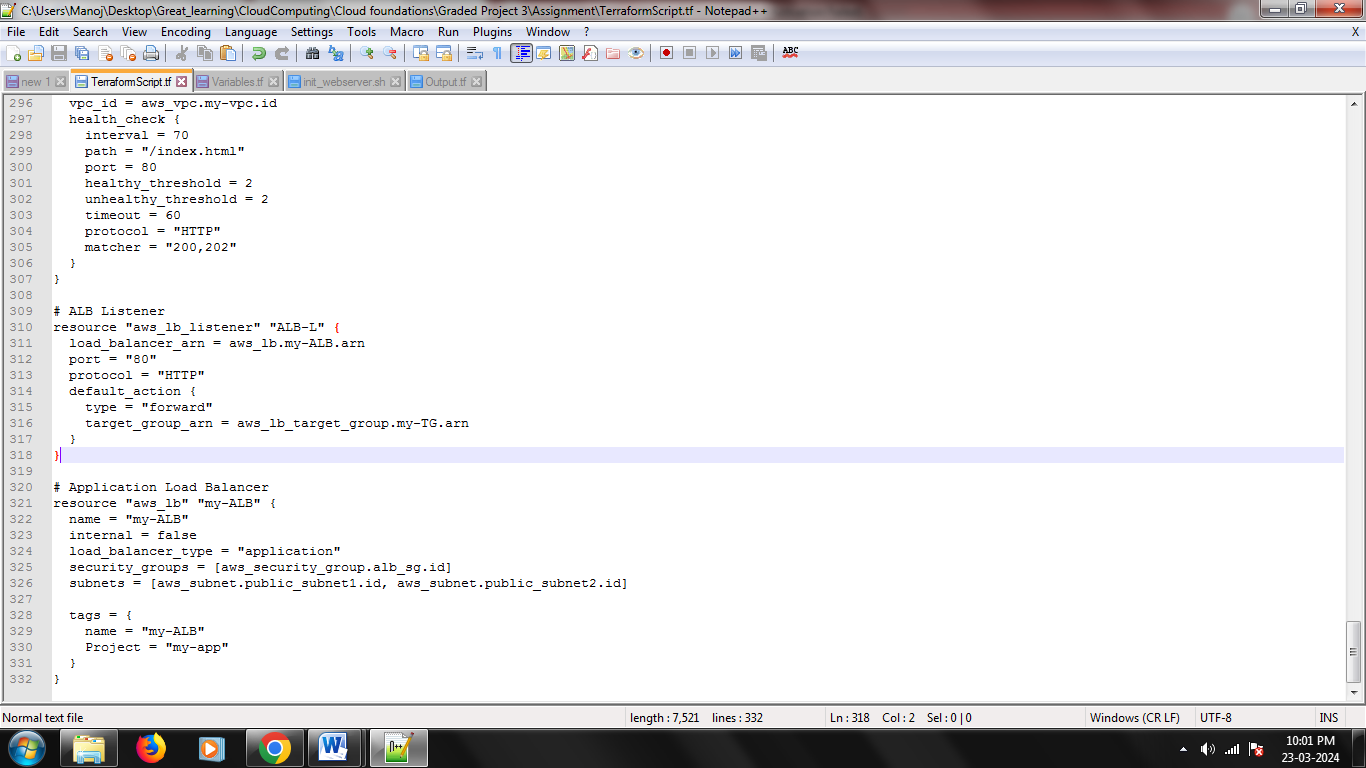




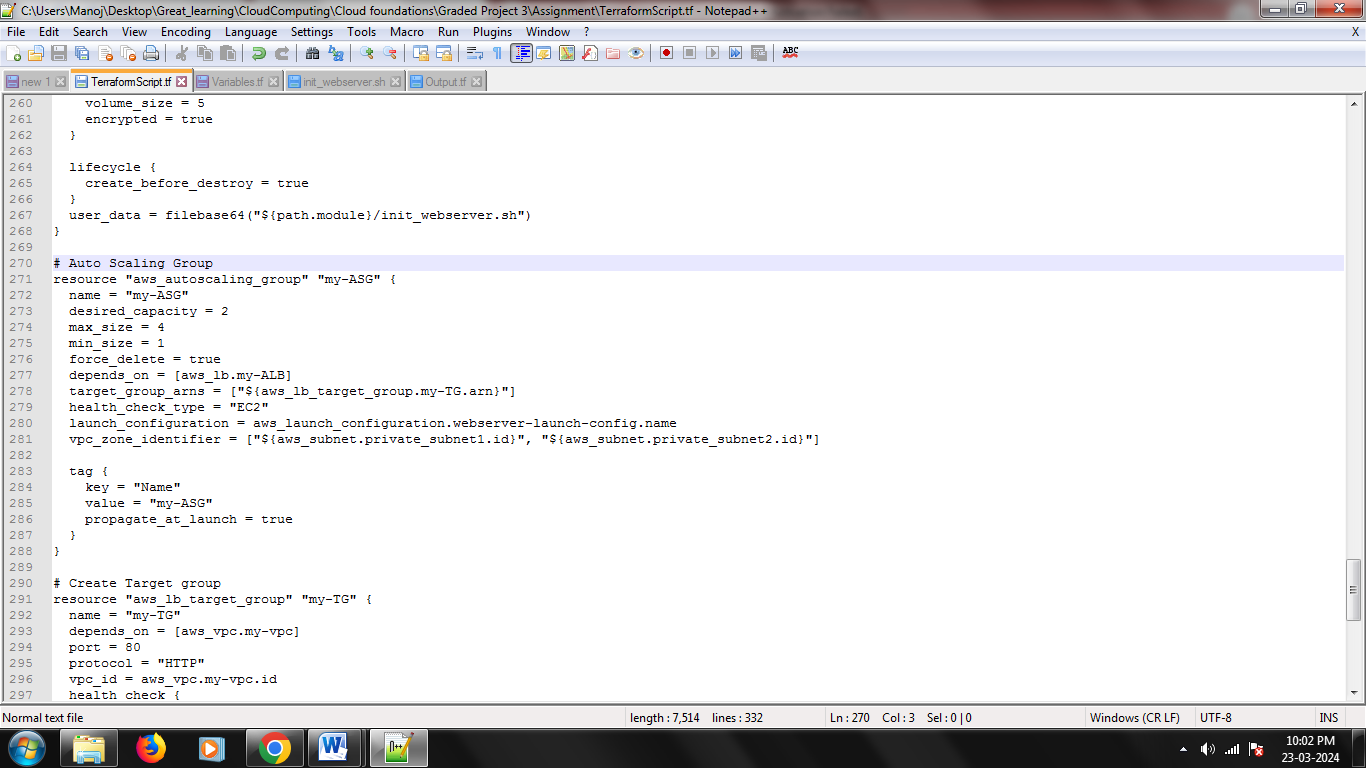
* 1. **EC2 instances**



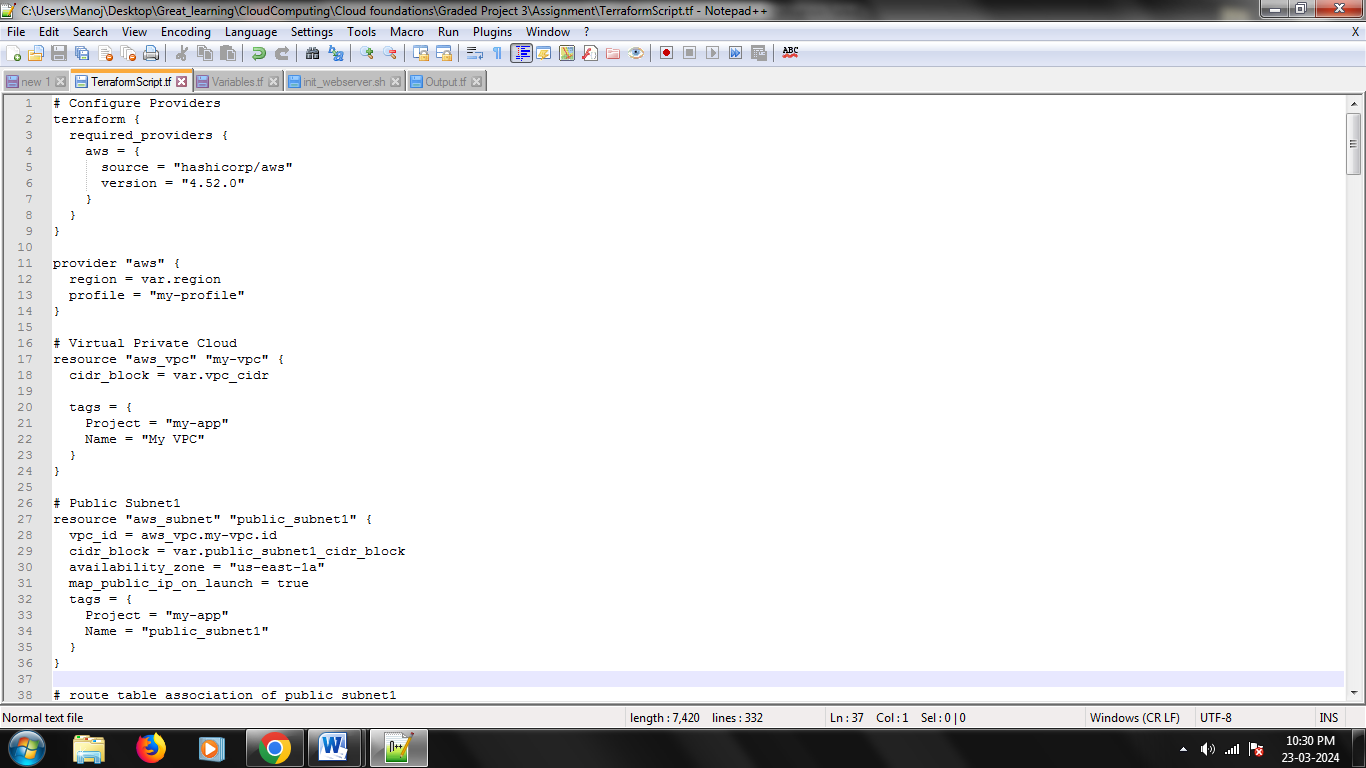
* 1. **Elastic Load balancer**

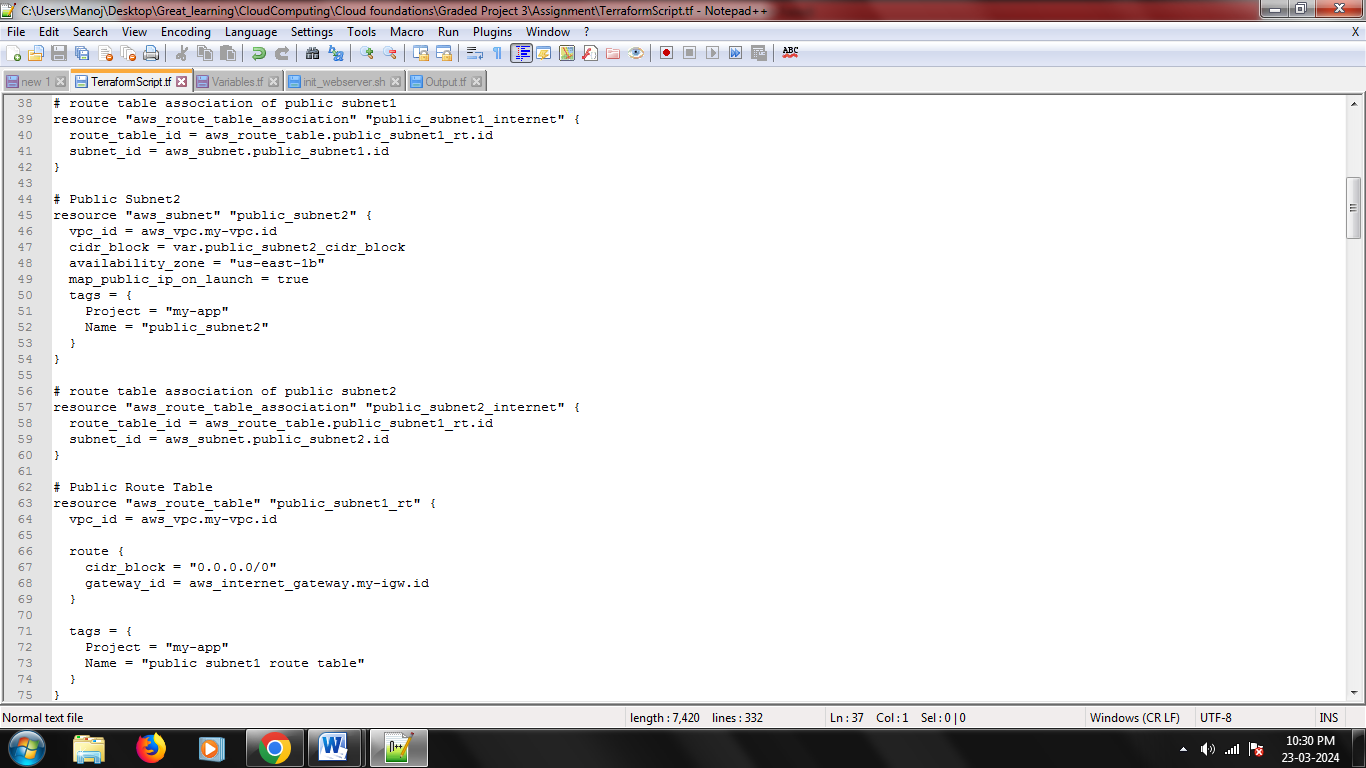


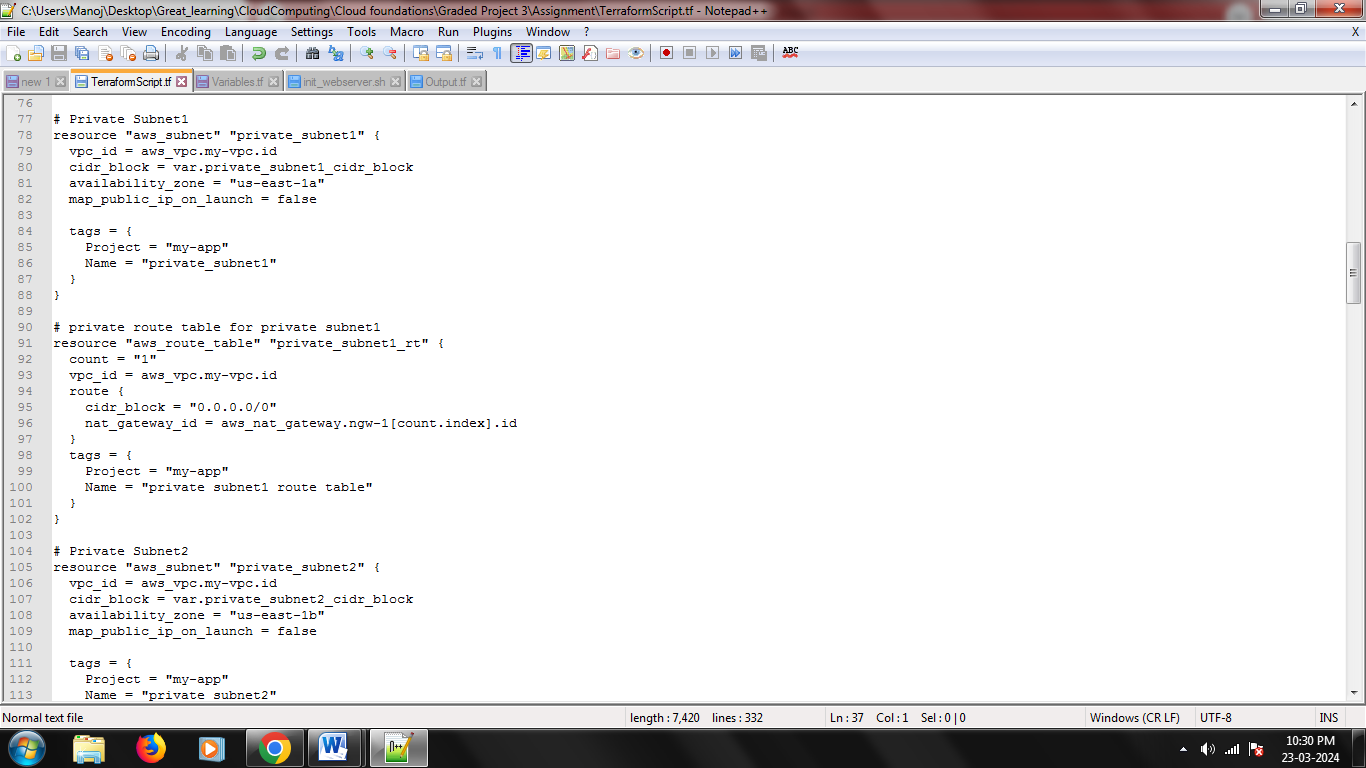
* 1. **Autoscaling group**

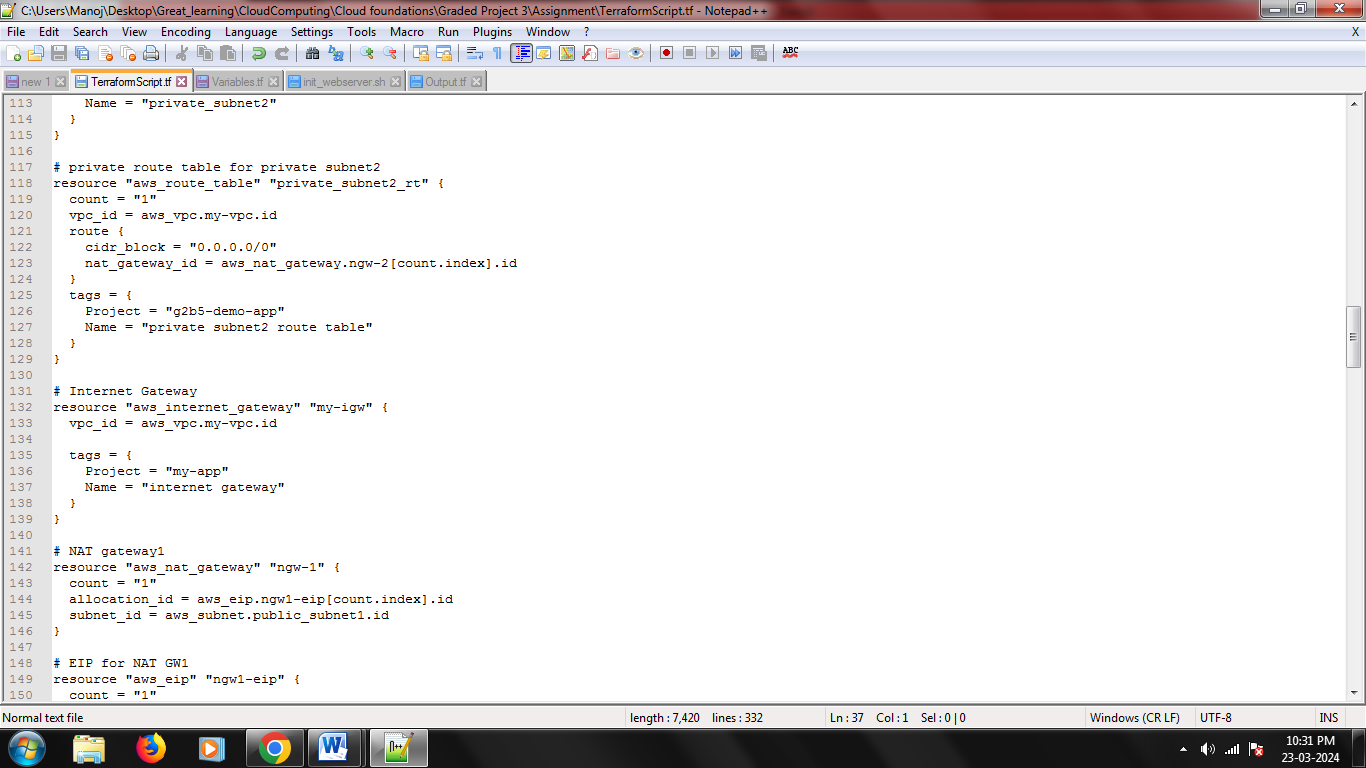


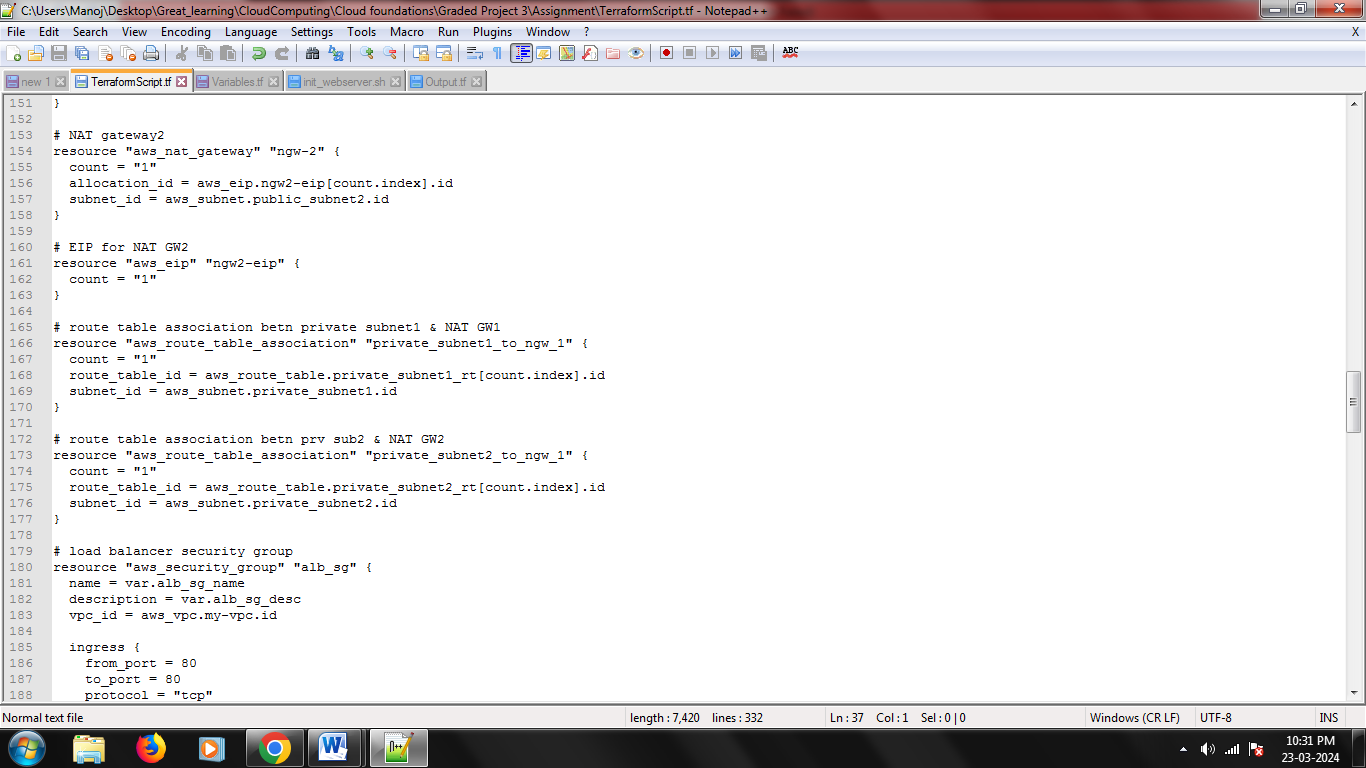
* 1. **Whole Terraform Script (In Case you need it)**

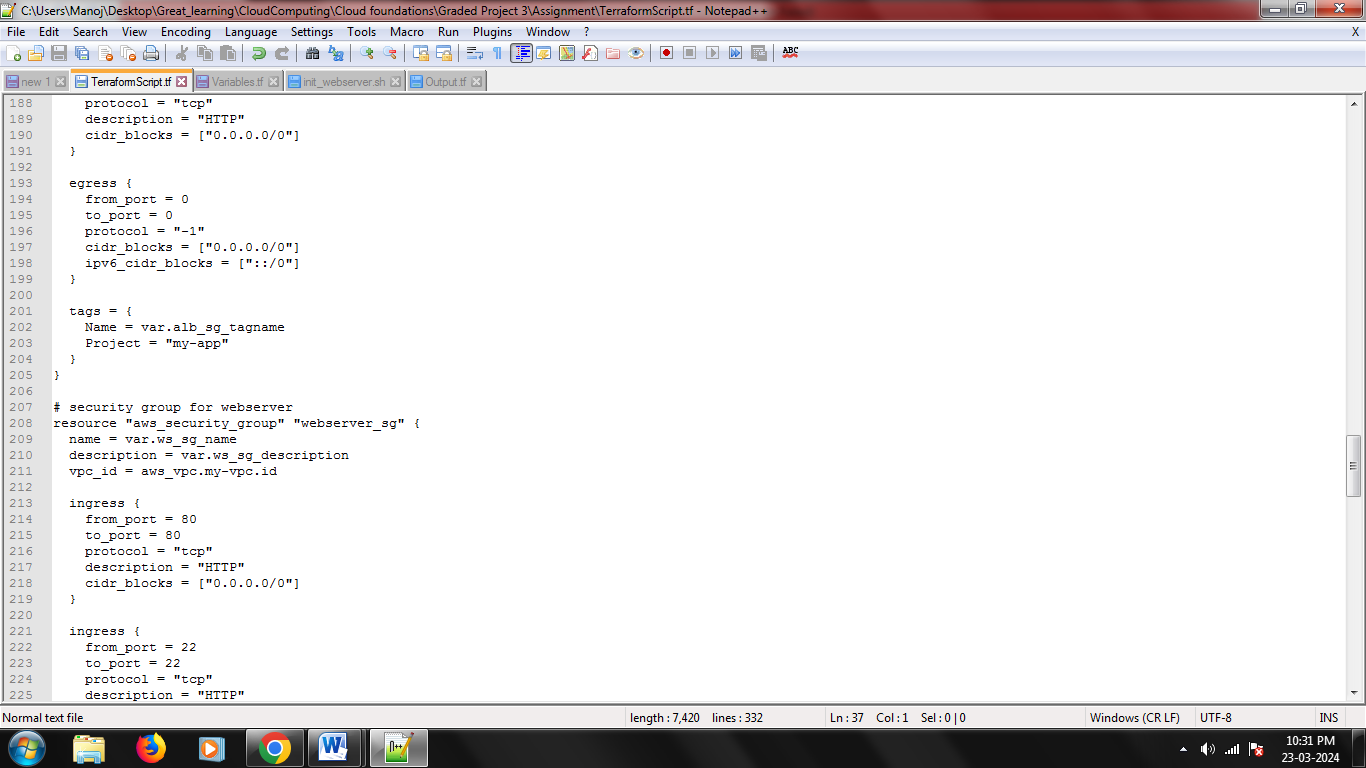


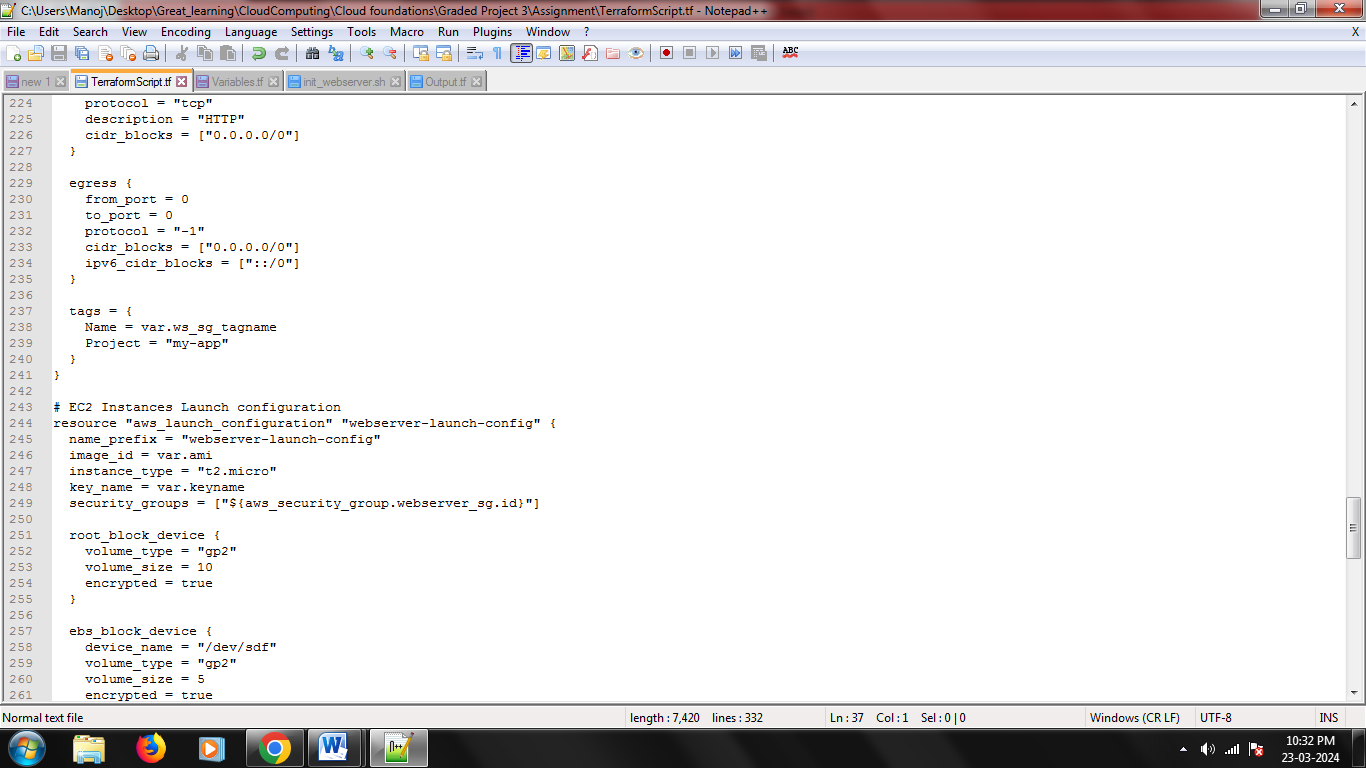


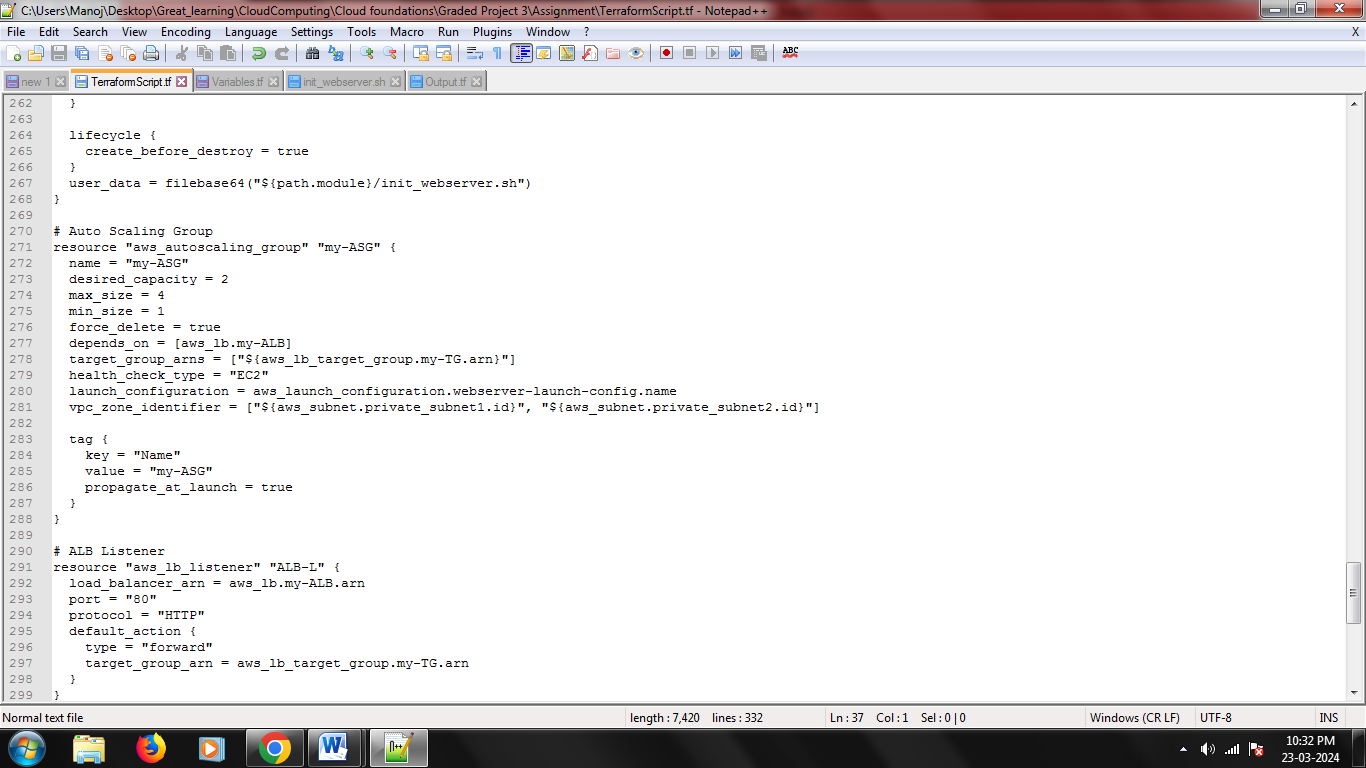


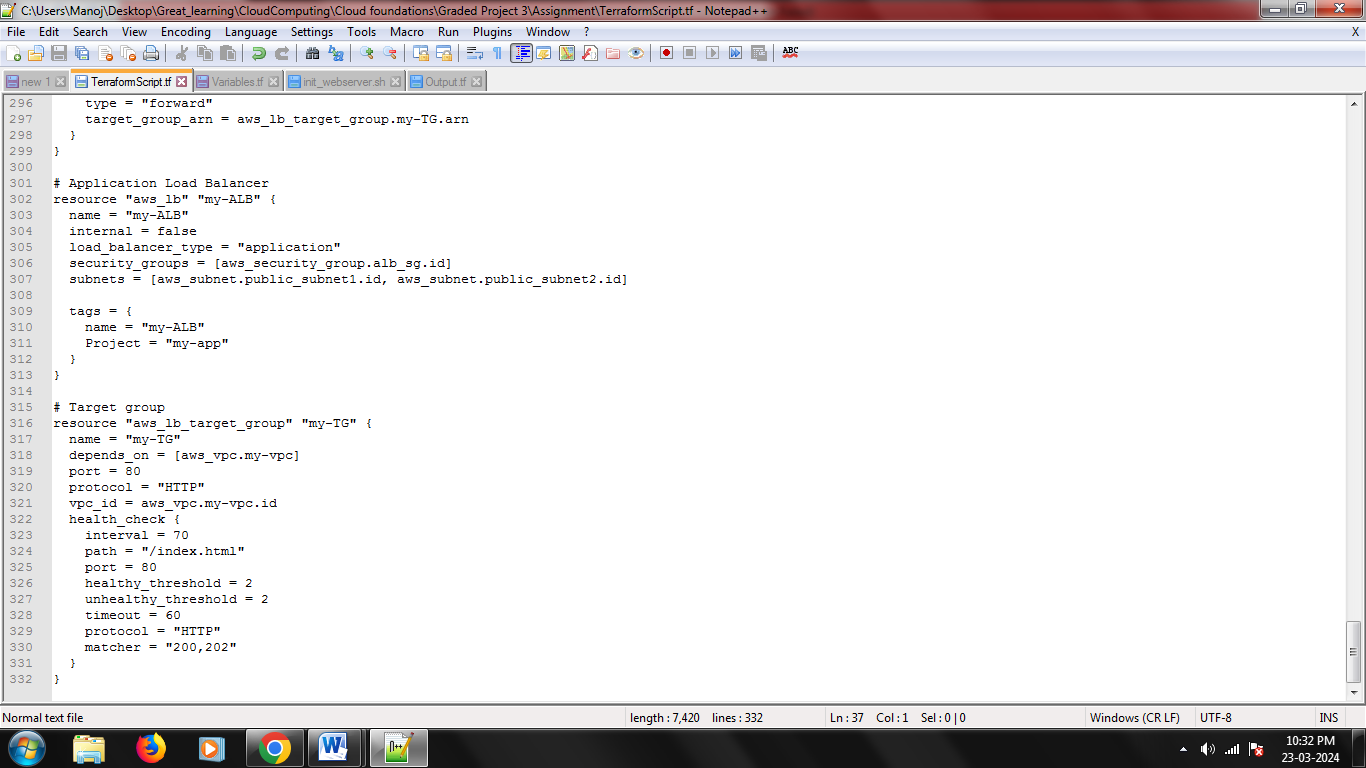




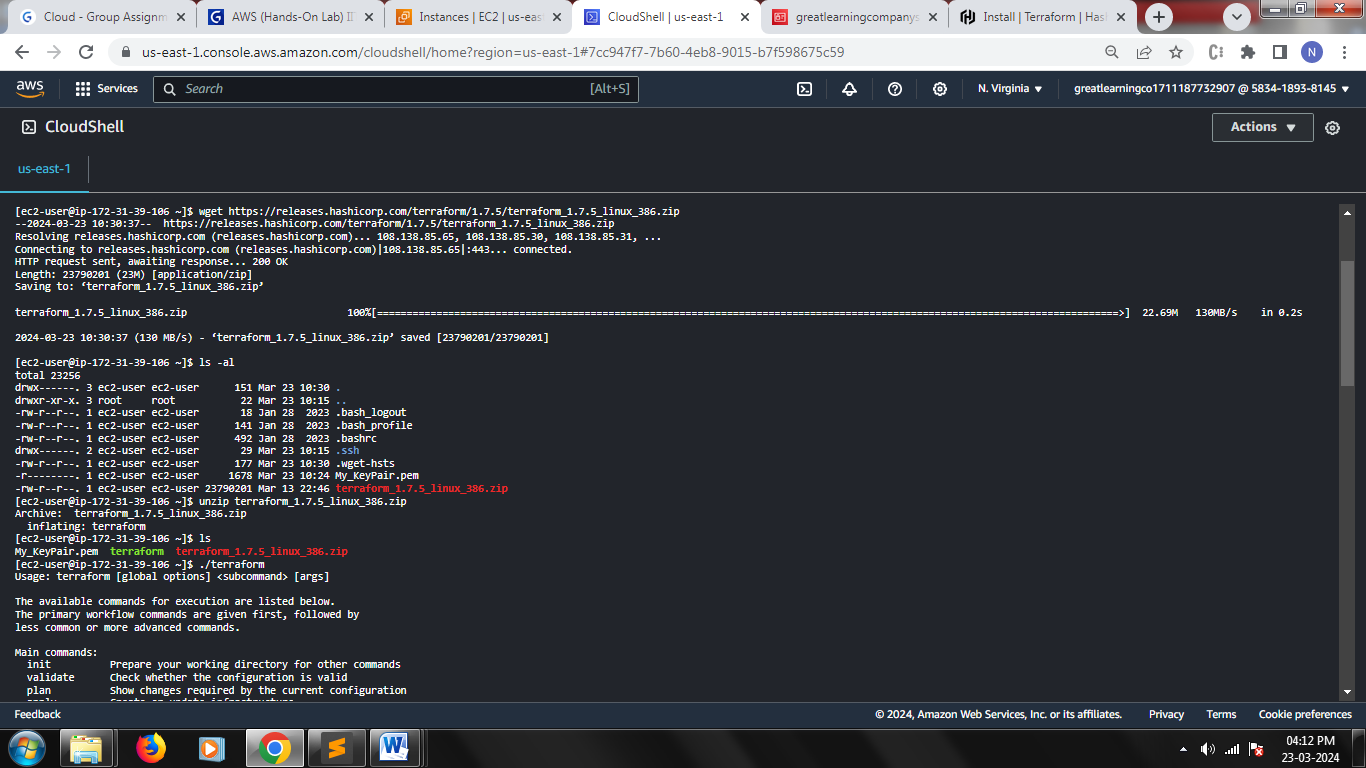




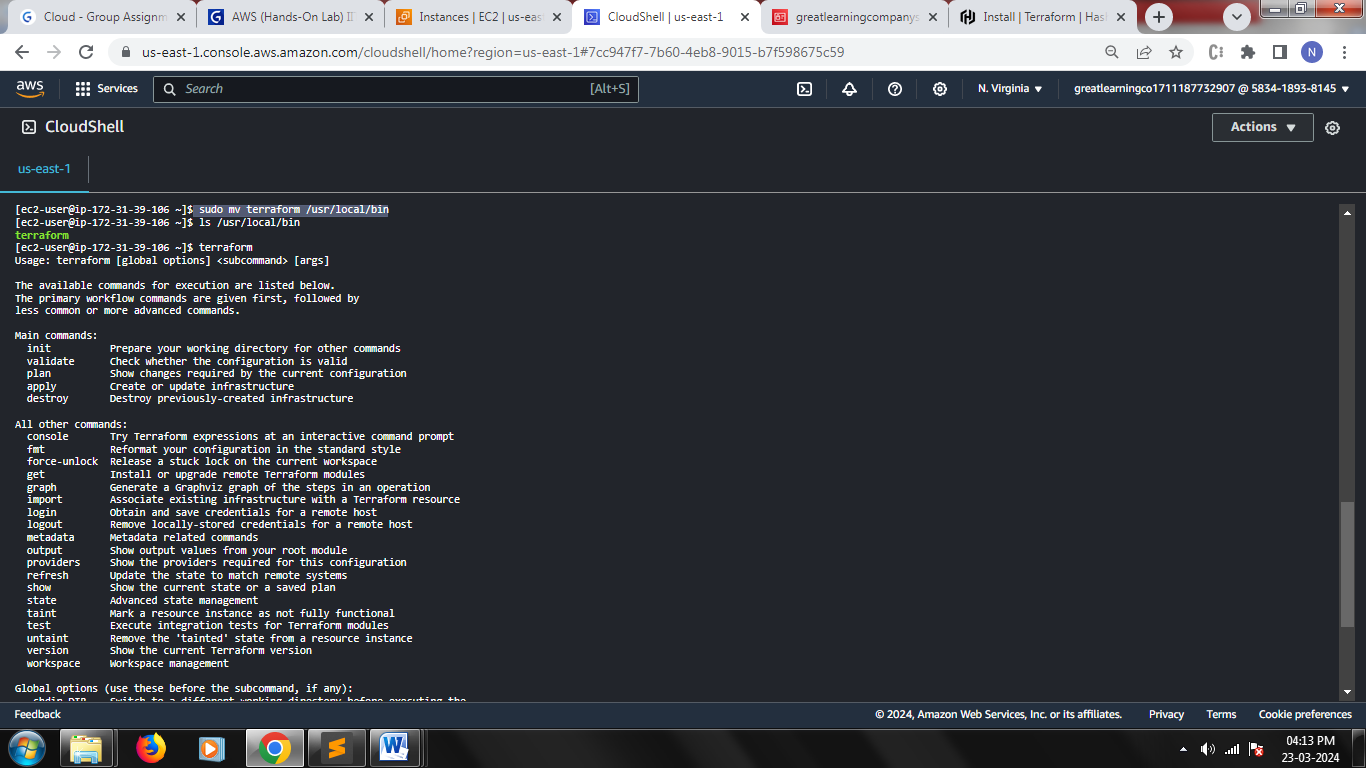




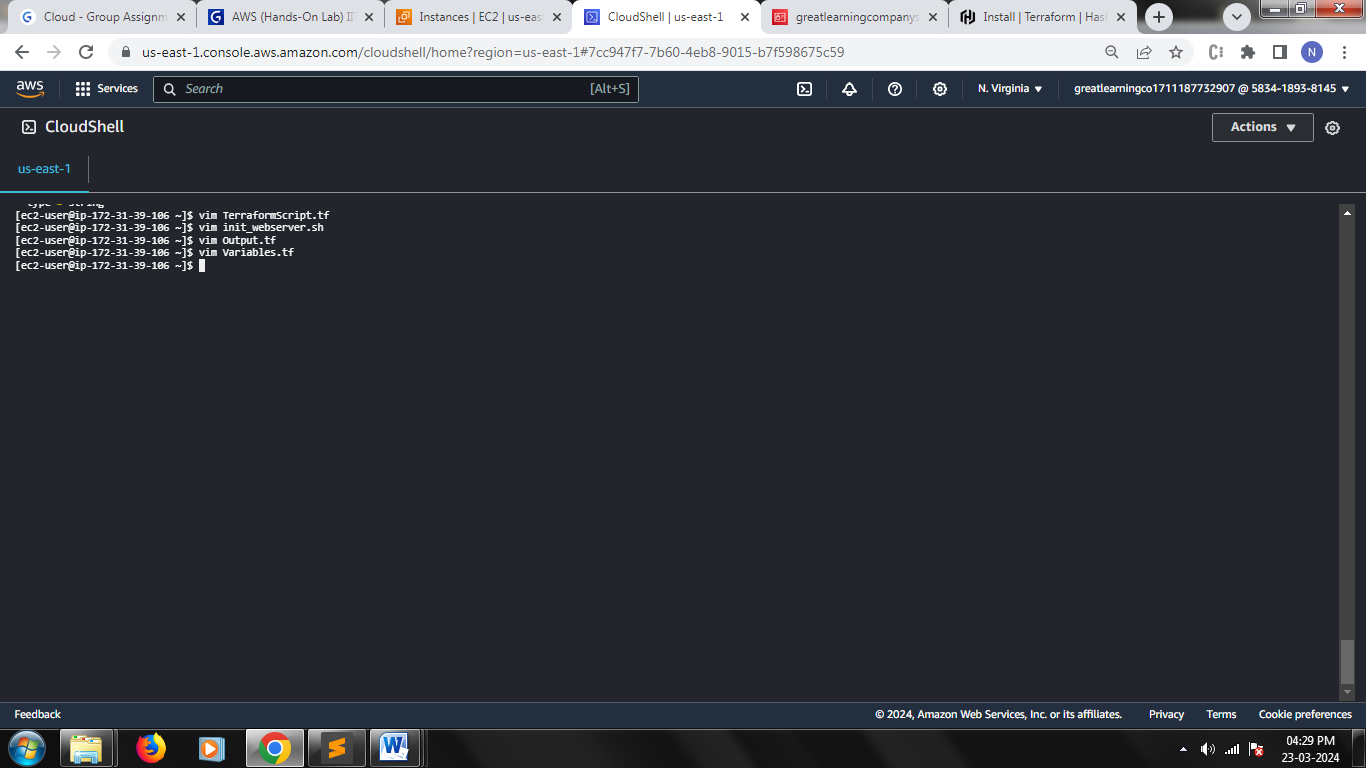
1. **Screenshot showing successful execution of Terraform script**
   * **Install Terraform**



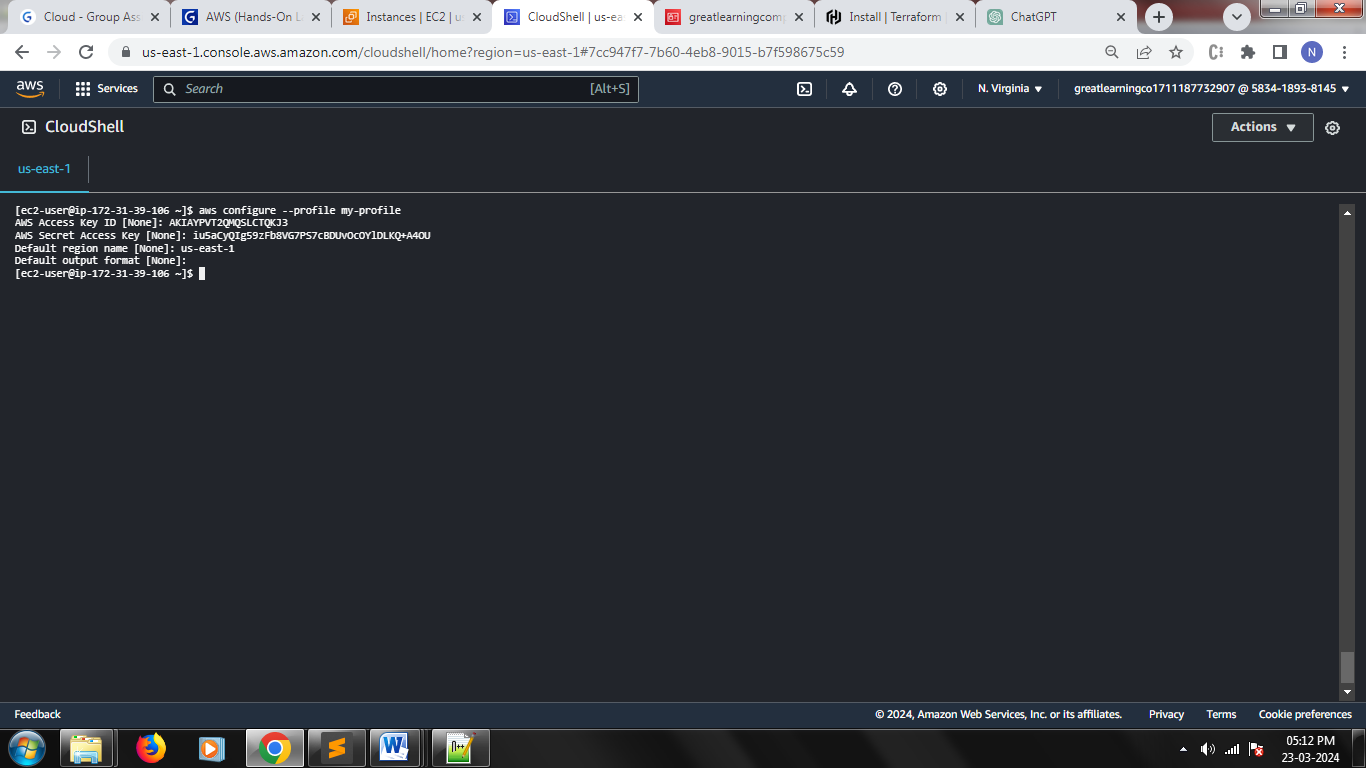
* + **Unzip terraform and move to bin folder so that we can globally access**



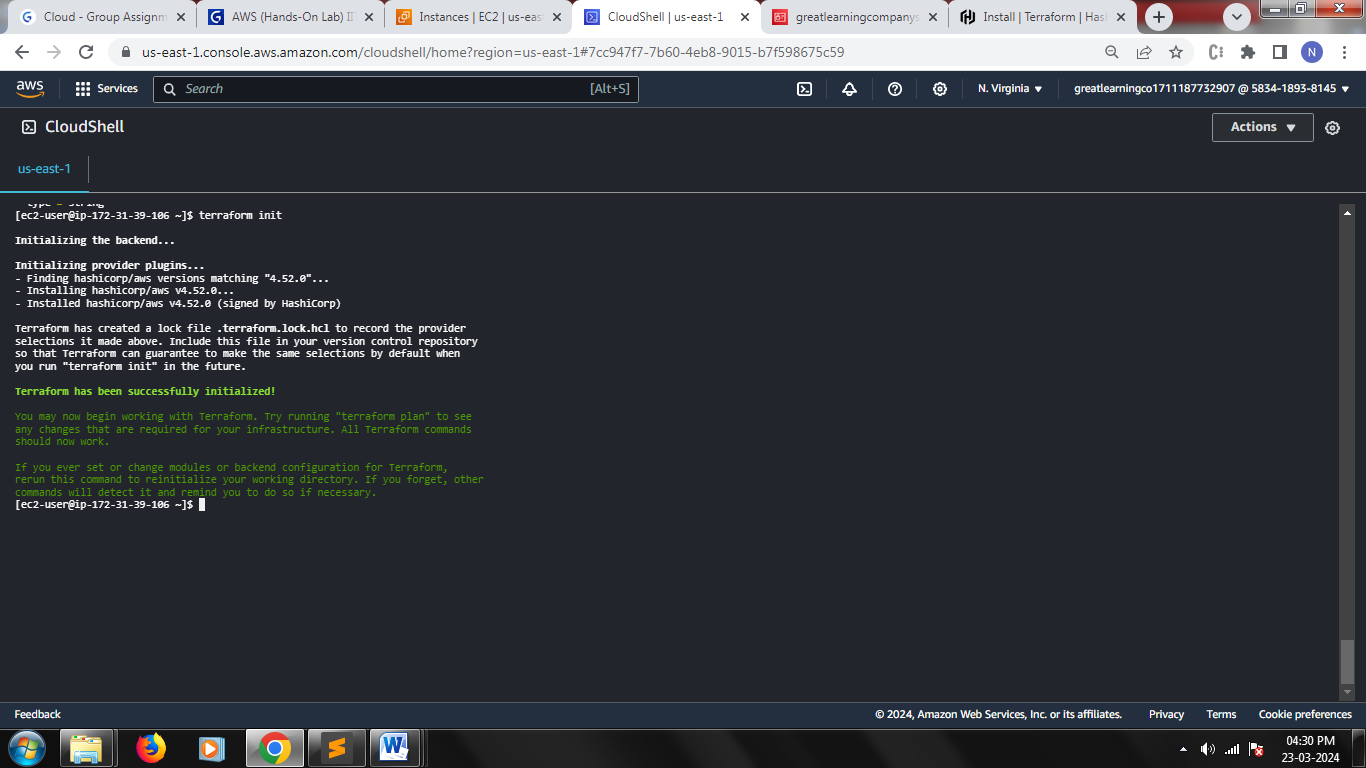
* + **Create Terraform, Variables, Output, Script files**



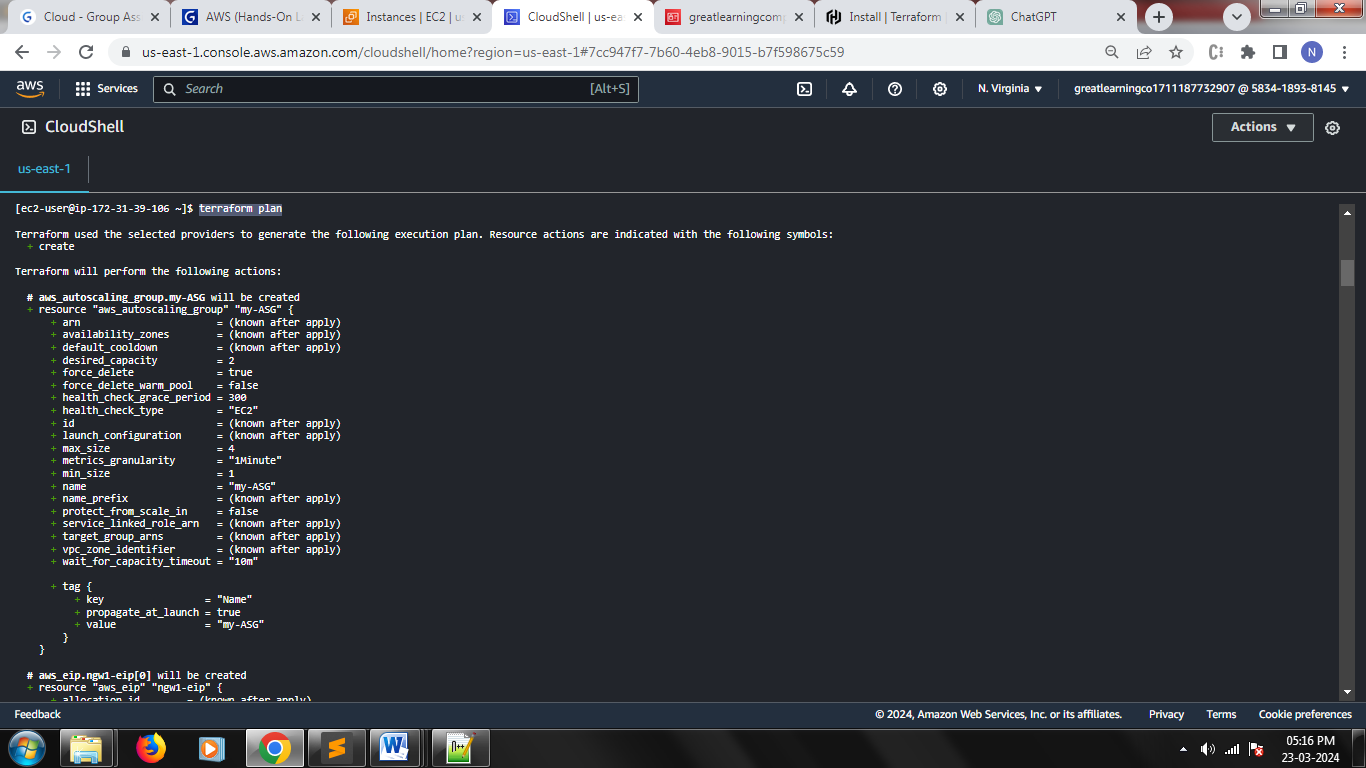
* + **Configure CLI Profile**

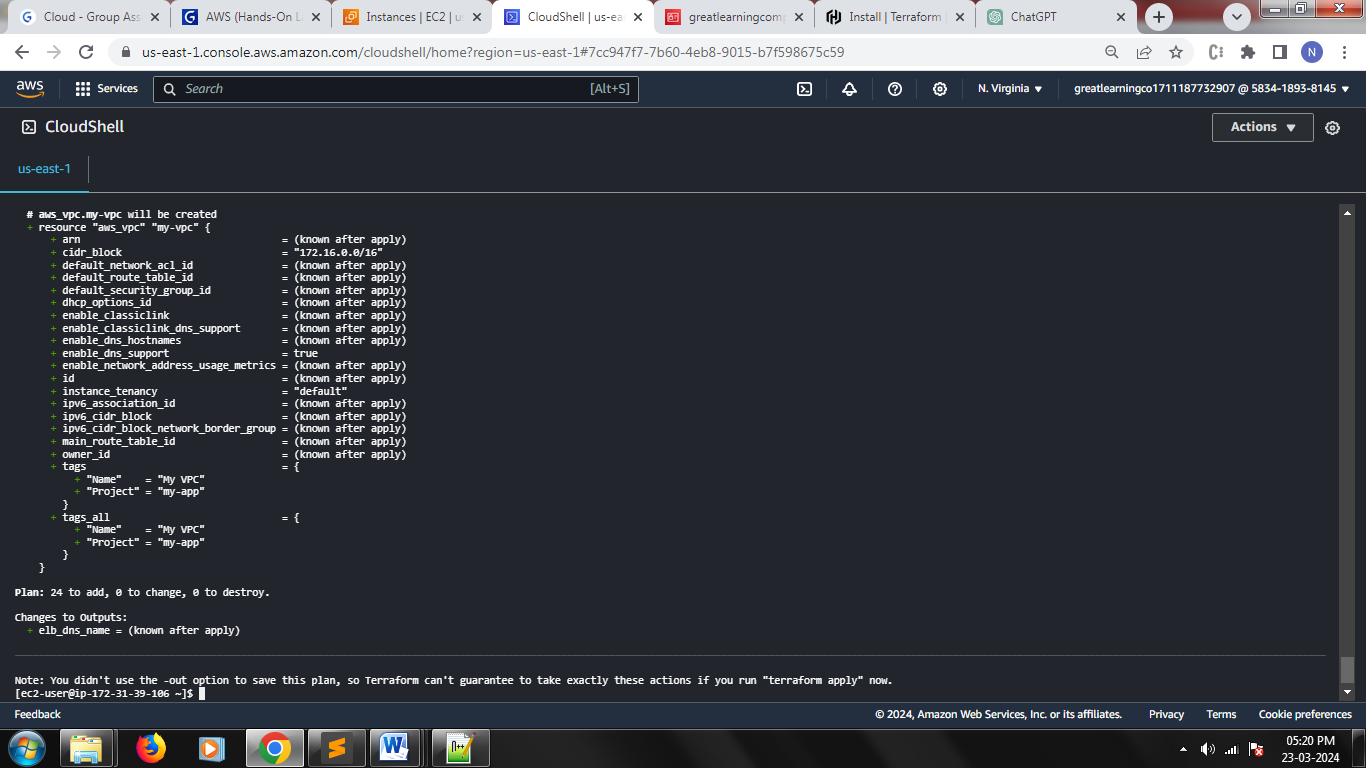


* + **Terraform init**

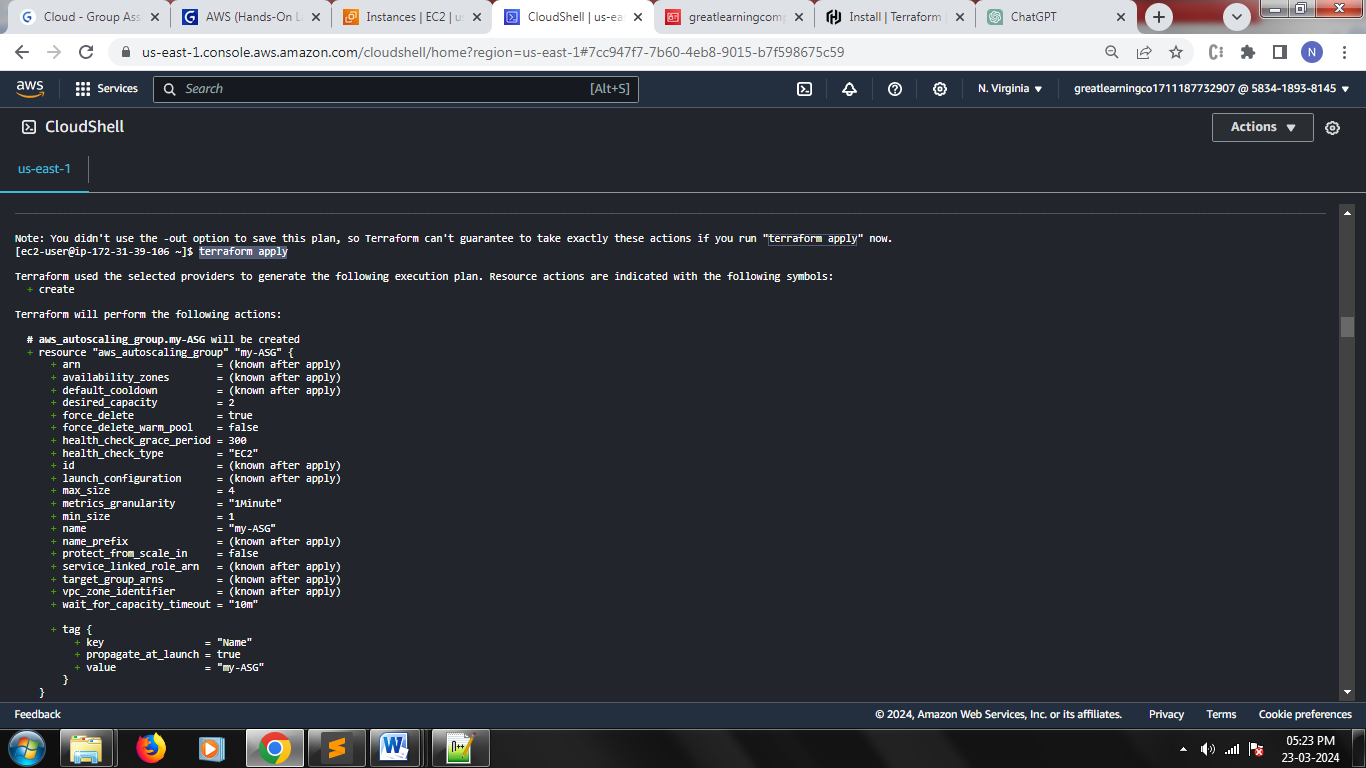


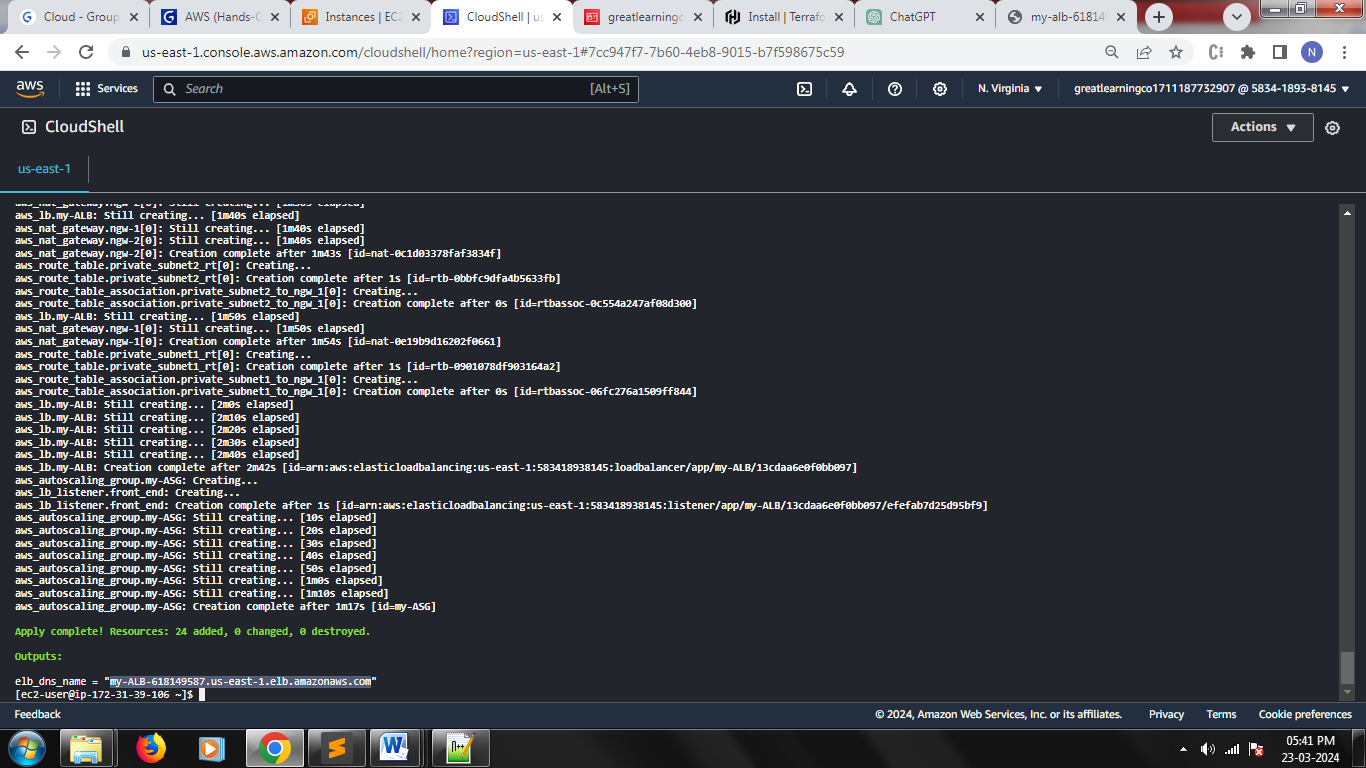
* + **Terraform plan**



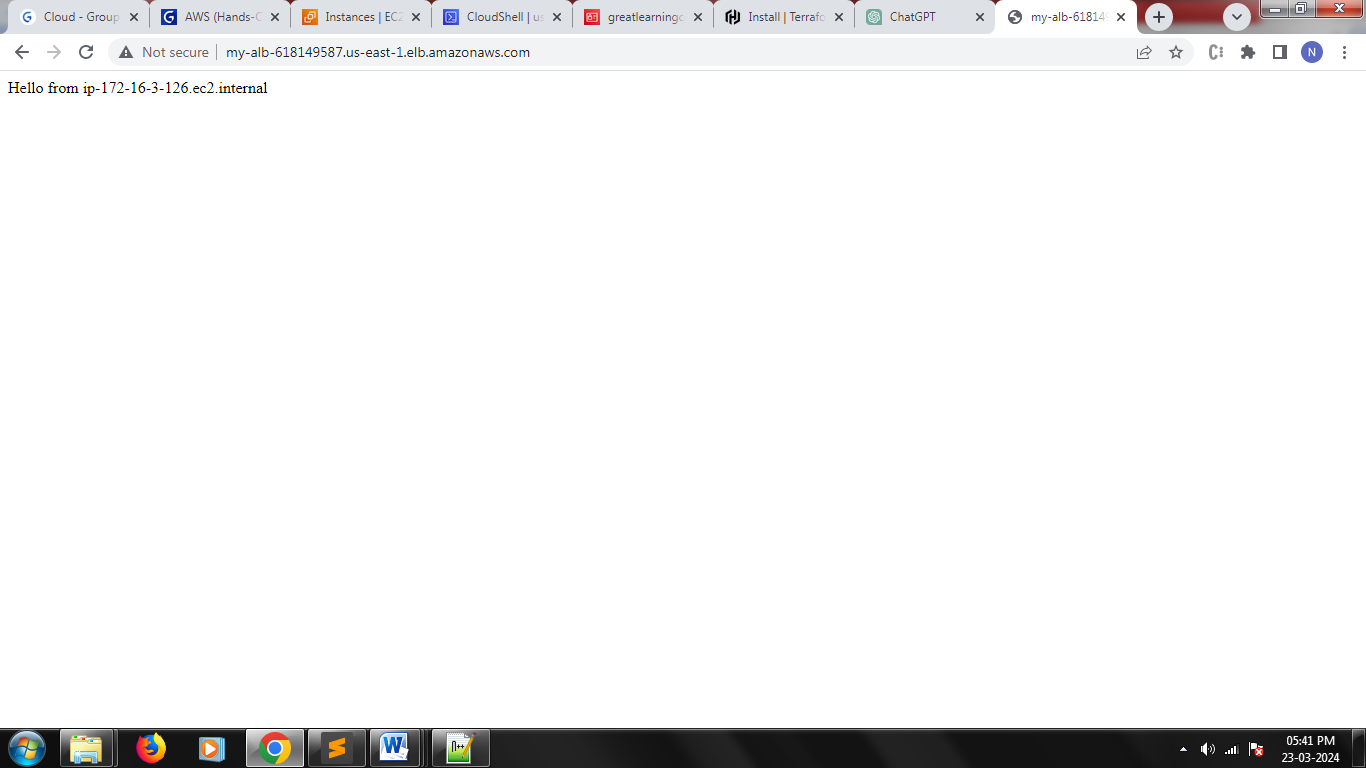


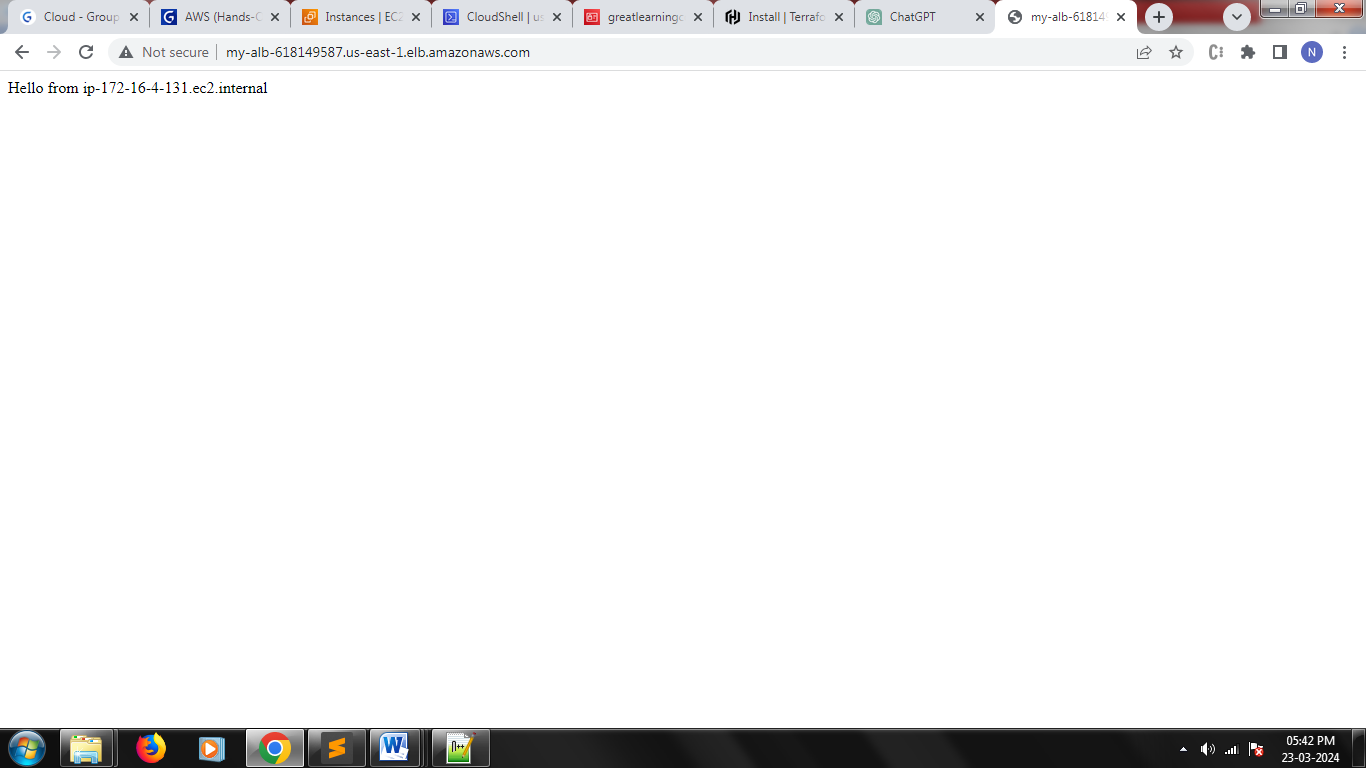
* + **Terraform apply**





* + **Running application on 2 EC2 instances created by ALB**





* + **Created Resources in AWS**

