

Practical: 5

AIM- Hosting Websites on Amazon EC2 Instances.

Submitted By: ADESHARA BRIJESH G.
Enrollment number: 21012021001



**Ganpat
University**
॥ विद्यया समाजोत्कर्षः ॥

U.V. Patel
College of
Engineering

Department of
Information Technology

Title: Hosting Websites on Amazon EC2 Instances.

Introduction:

- Cloud computing has transformed web hosting, providing scalable and reliable infrastructure solutions.
- This report explores the deployment of websites on Amazon EC2 instances, highlighting procedures and benefits.

Purpose:

- Gain practical experience in deploying websites on Amazon EC2 instances.
- Understand components, architecture, and procedures involved in cloud-based web hosting.

Components:

- **Amazon EC2:**
 - Virtual computing environments (instances) configurable with various operating systems.
 - Key component of Amazon Web Services (AWS) for scalable and customizable computing resources.
- **Web Hosting:**
 - Deployment of website files on a server accessible via the internet.
 - Components include web server software, website files, domain names, and optional databases.

Architecture:

- **Amazon EC2 Architecture:**
 - Instances provisioned with selected configurations (OS, instance type).
 - Network configurations (VPC, subnets, security groups) for accessibility.
 - Storage volumes (EBS) for data persistence and file storage.
- **Web Hosting on EC2:**
 - Configuration of web server software (e.g., Apache, Nginx) on EC2 instances.
 - Deployment of website files to the server.
 - DNS settings adjustment to point to the instance's IP address or domain name.

Practical: 5

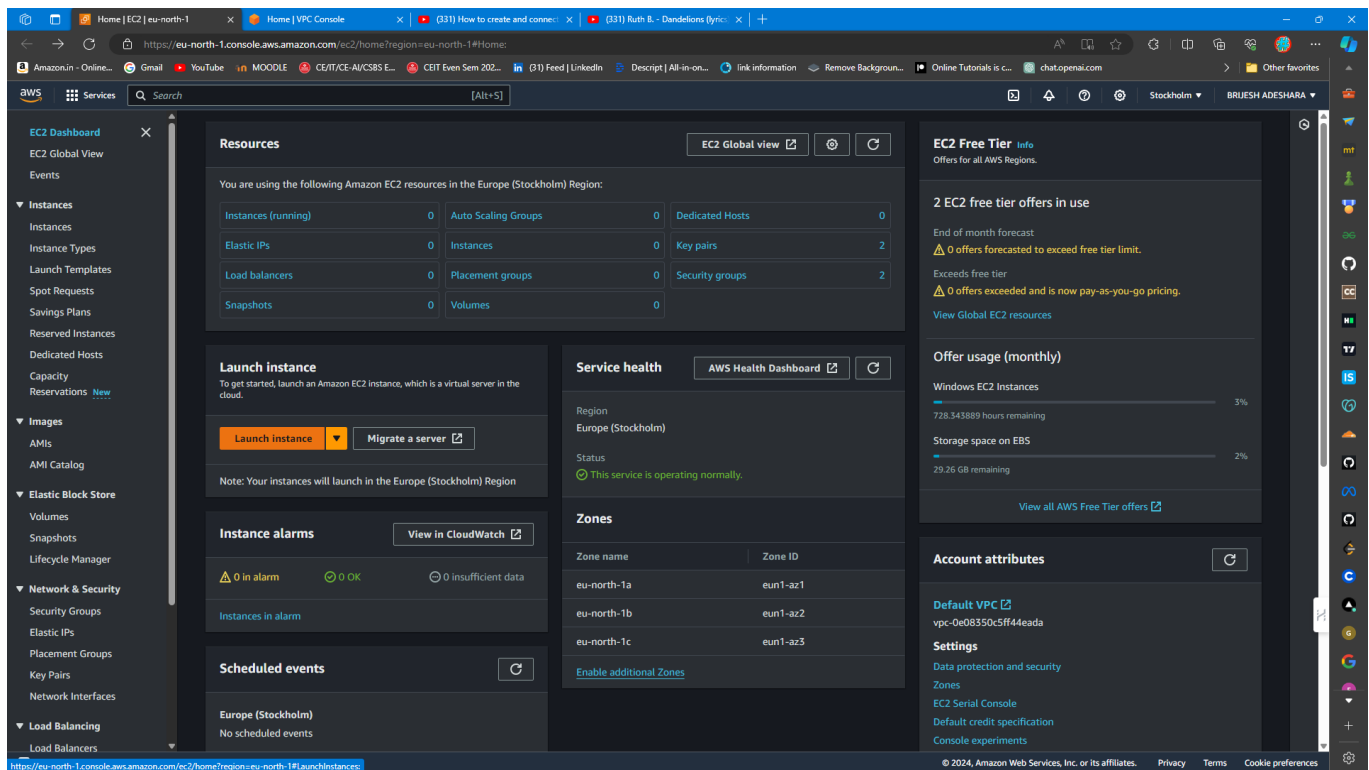
Pros/Cons:

- **Pros:**
 - Scalability: Resources can be scaled up or down based on demand.
 - Flexibility: Choose from a variety of instance types and configurations.
 - Reliability: Benefit from AWS's robust infrastructure and redundancy.
- **Cons:**
 - Cost: Pay-per-use model may become expensive for high-traffic websites.
 - Complexity: Requires knowledge of AWS services and configurations.
 - Maintenance: Regular updates and monitoring are necessary.

Implementation Steps:

1. Sign Up for AWS Account:

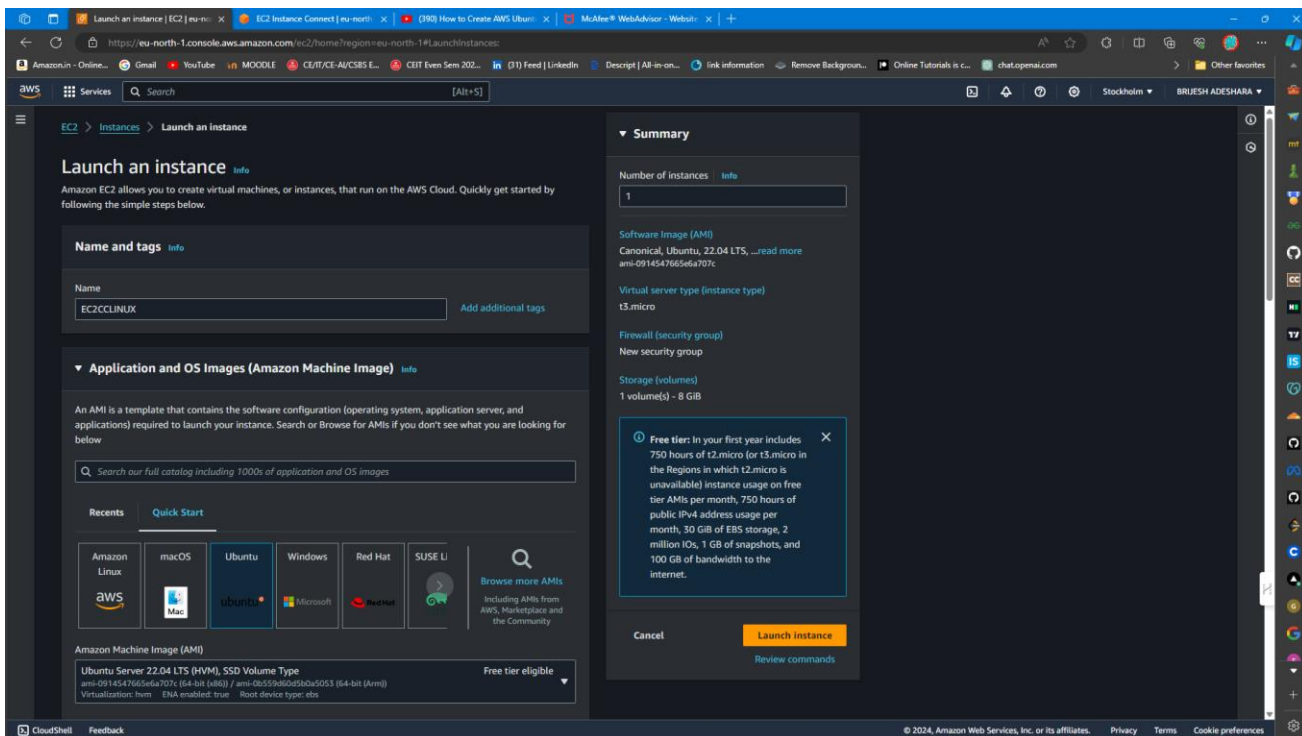
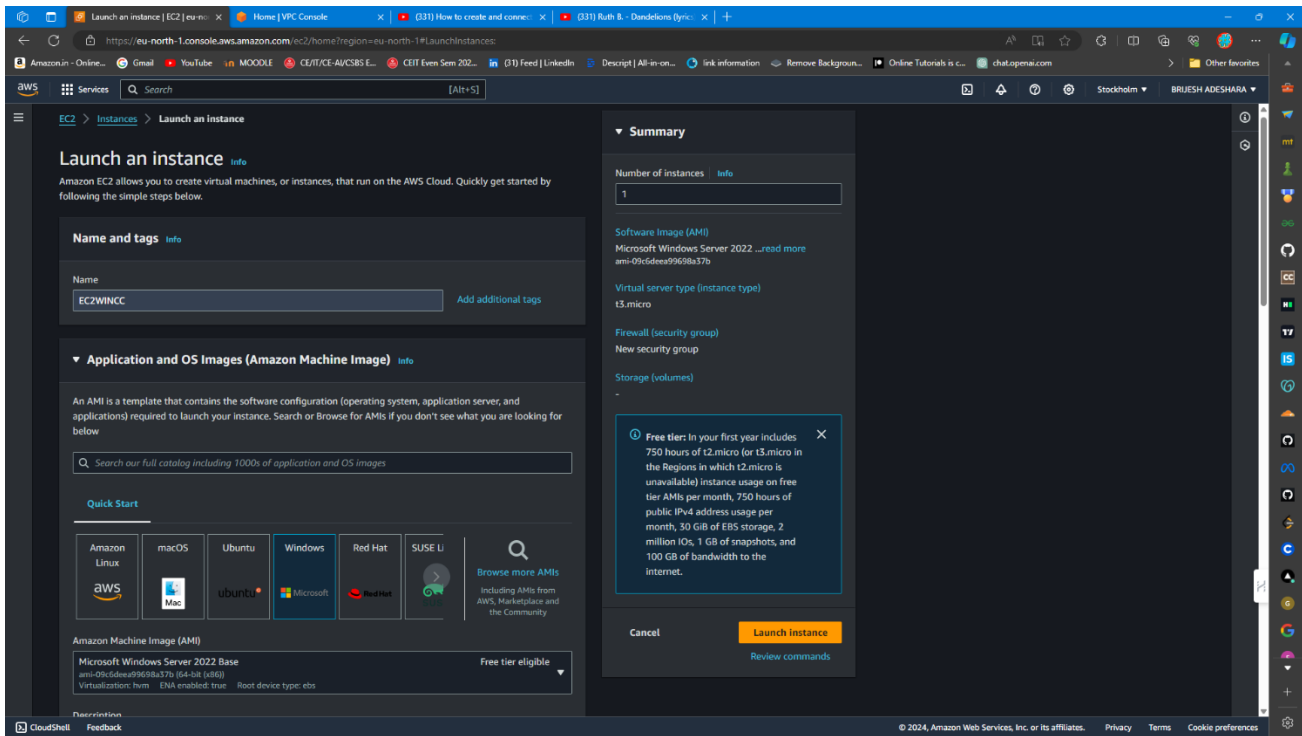
- Create an AWS account if not already registered.



Practical: 5

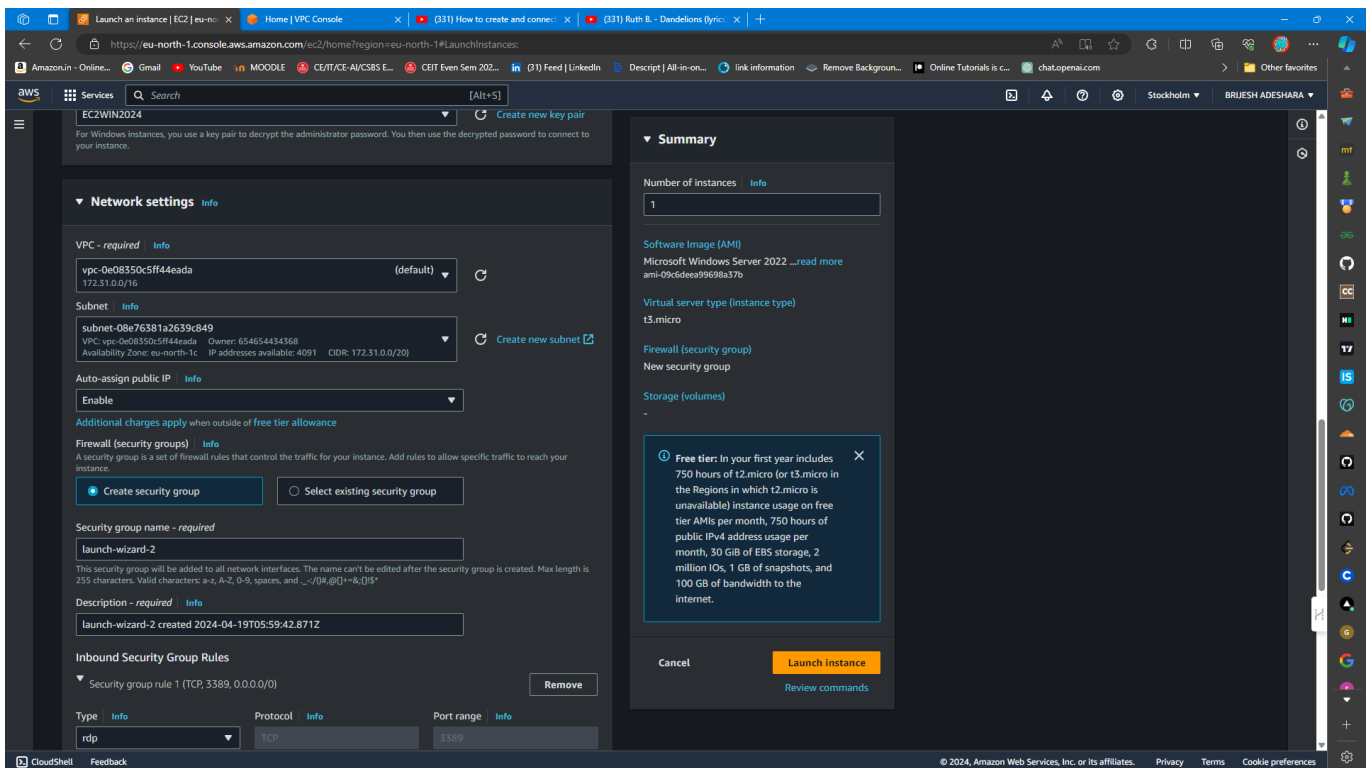
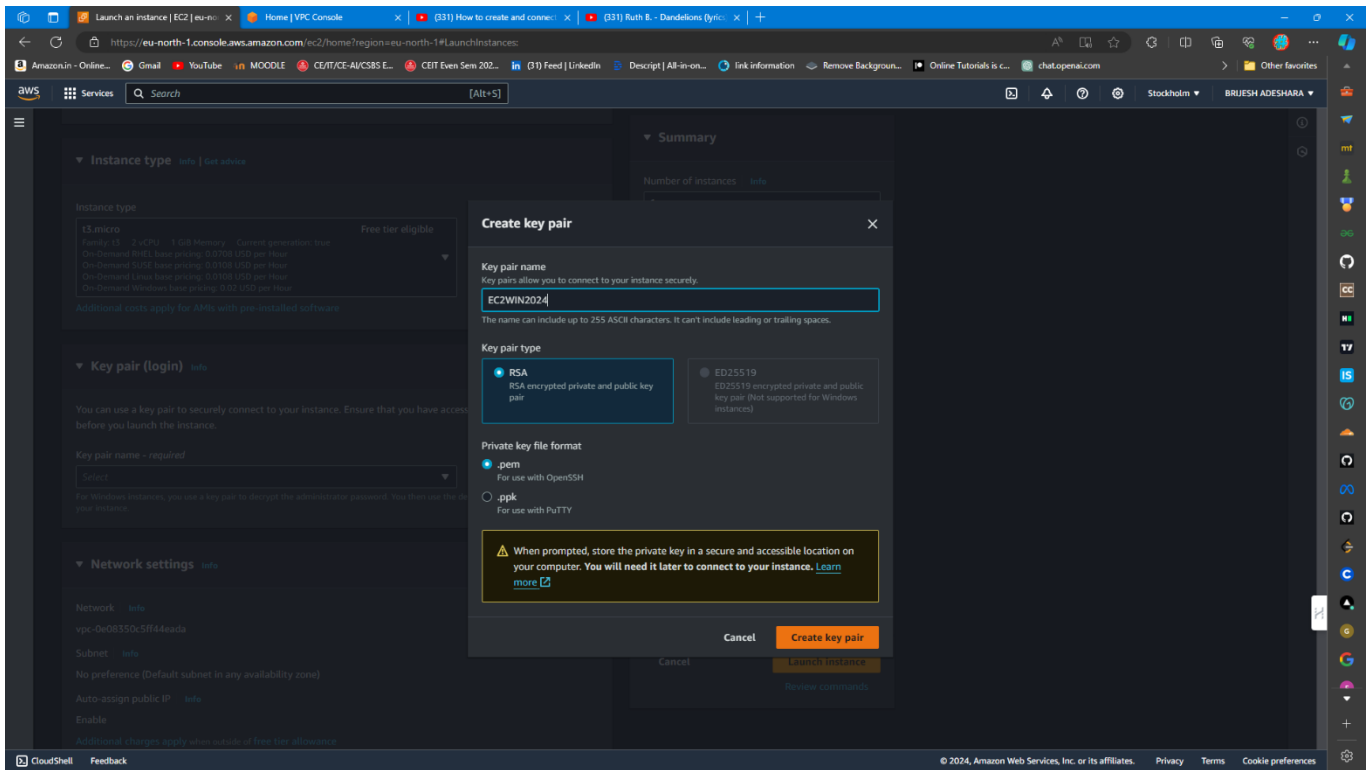
2. Launch EC2 Instance:

- Choose an appropriate AMI (Amazon Machine Image) based on your operating system preference (Windows/Linux).



Practical: 5

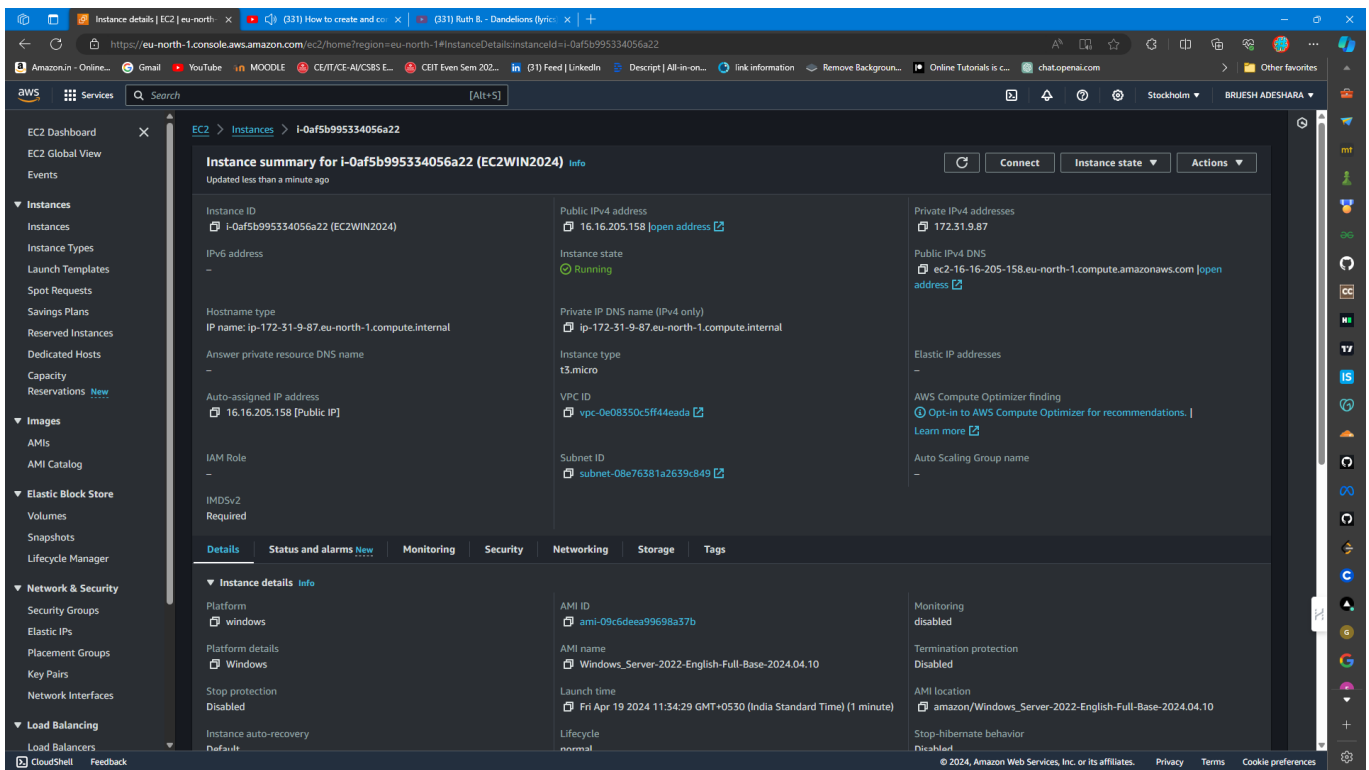
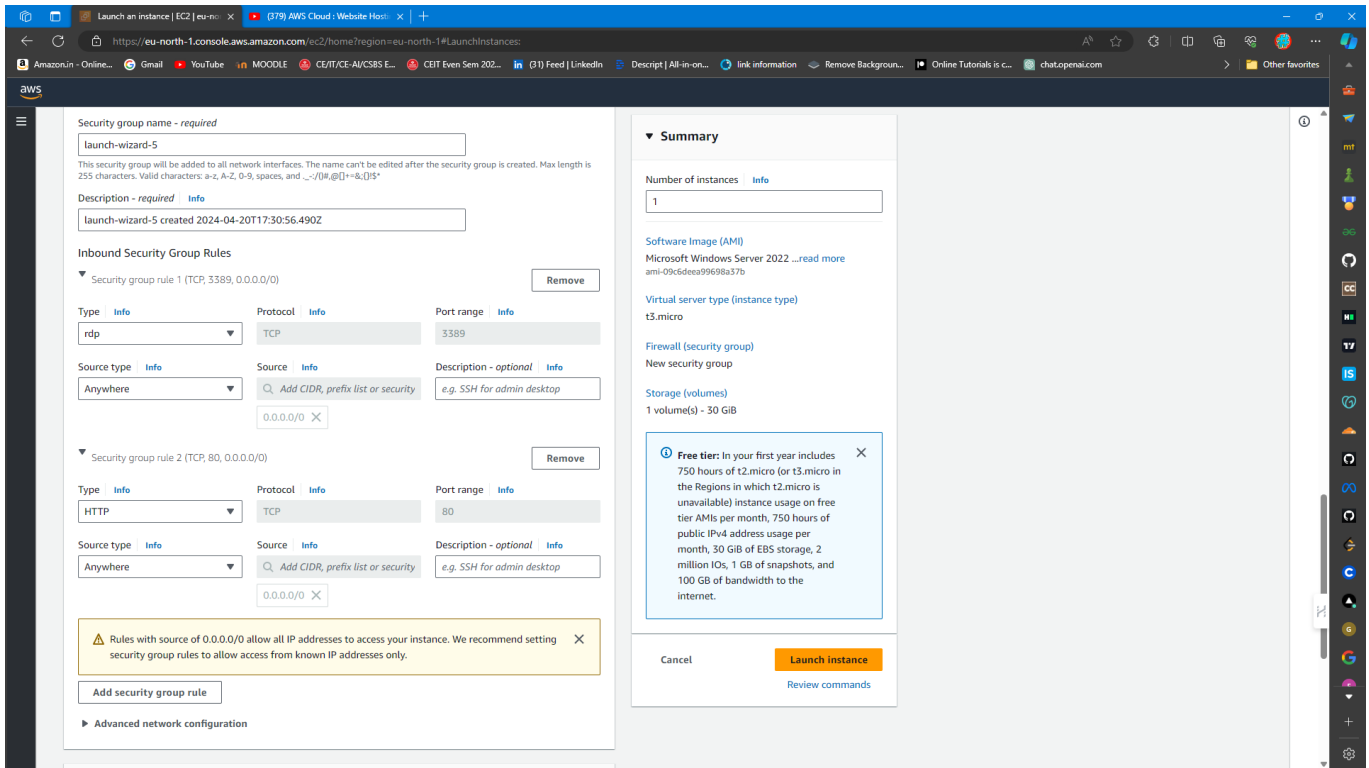
- Configure instance type, networking settings (VPC, subnet), and storage (EBS volume).



Practical: 5

3. Configure Security Groups:

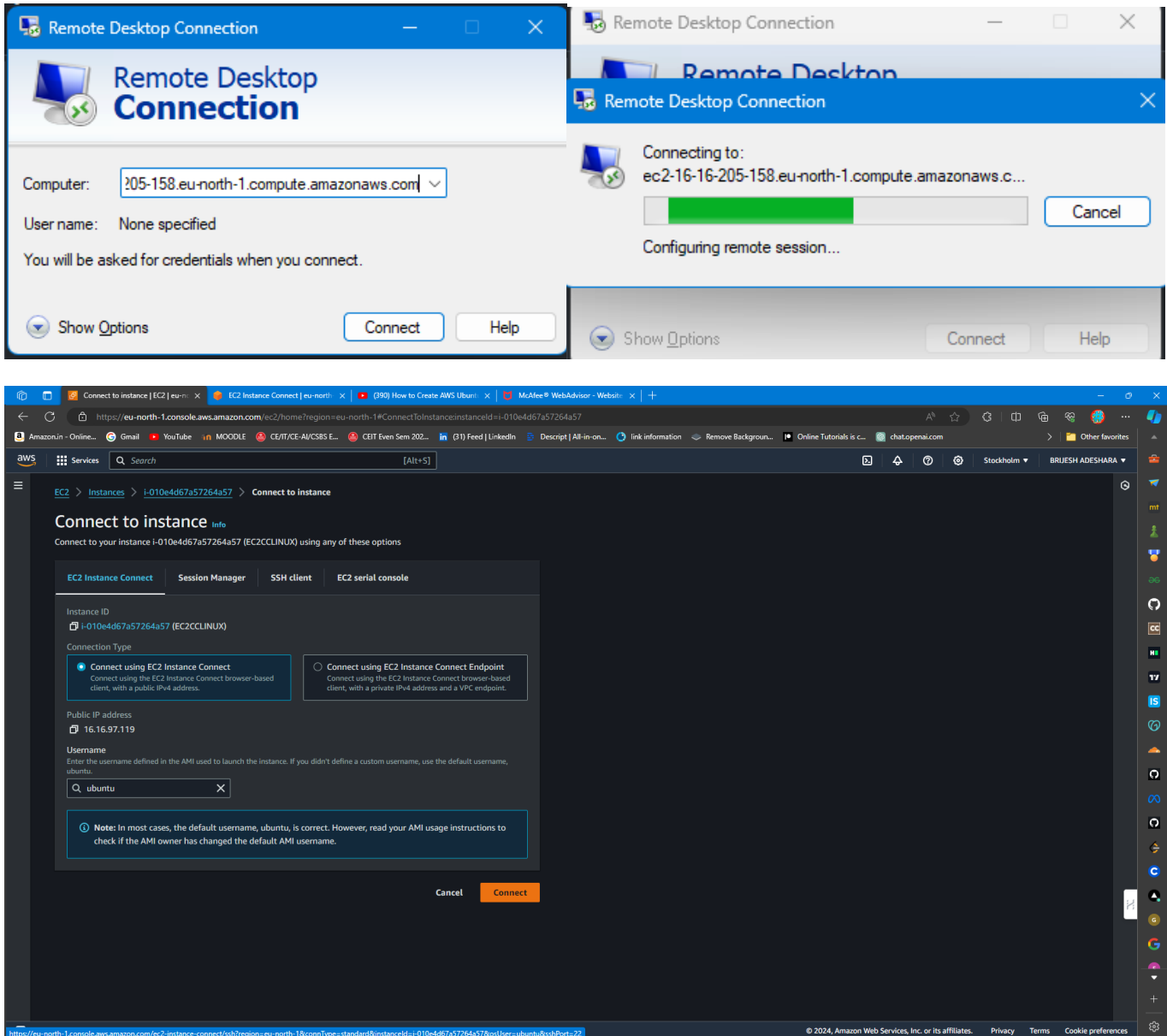
- Define inbound rules to allow HTTP (port 80) and/or HTTPS (port 443) traffic.



Practical: 5

4. Connect to EC2 Instance:

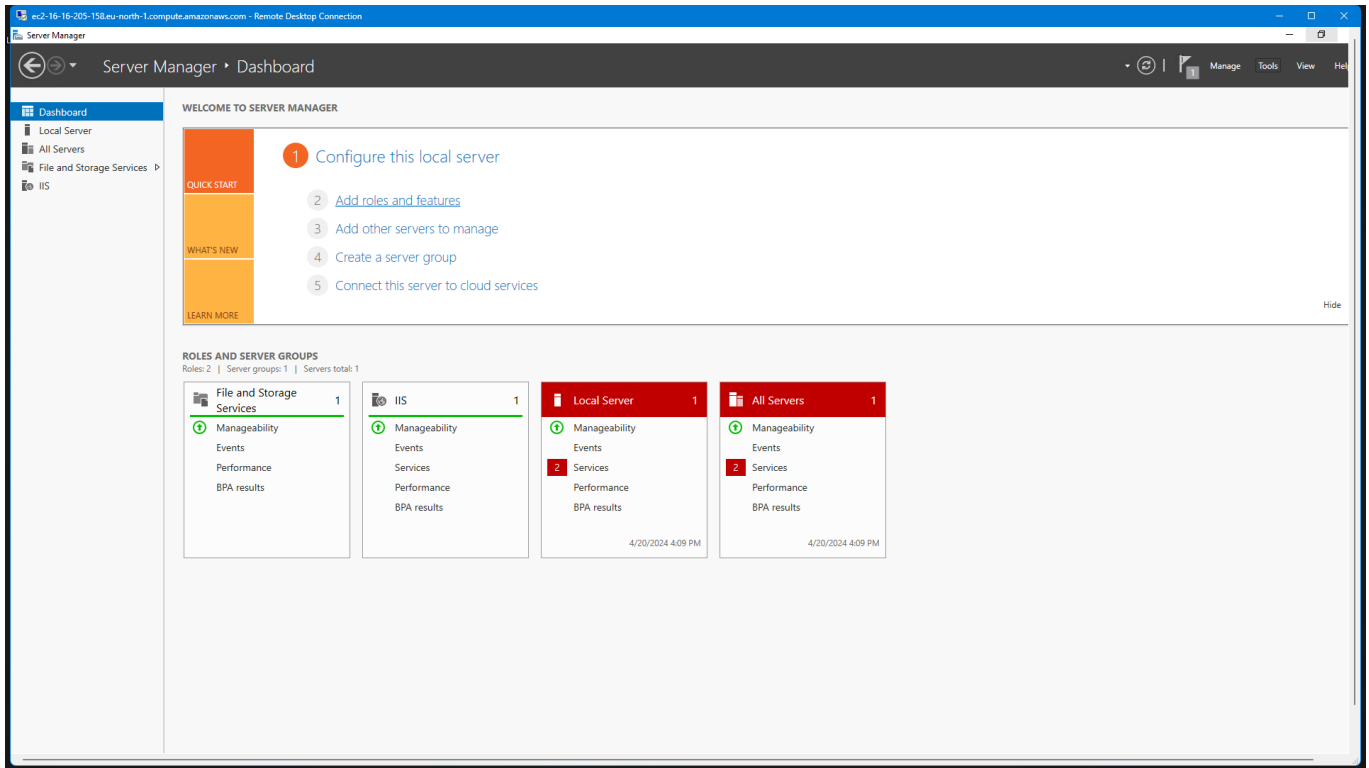
- Use Remote Desktop Protocol (RDP) for Windows instances or Secure Shell (SSH) for Linux instances to connect to your EC2 instance.



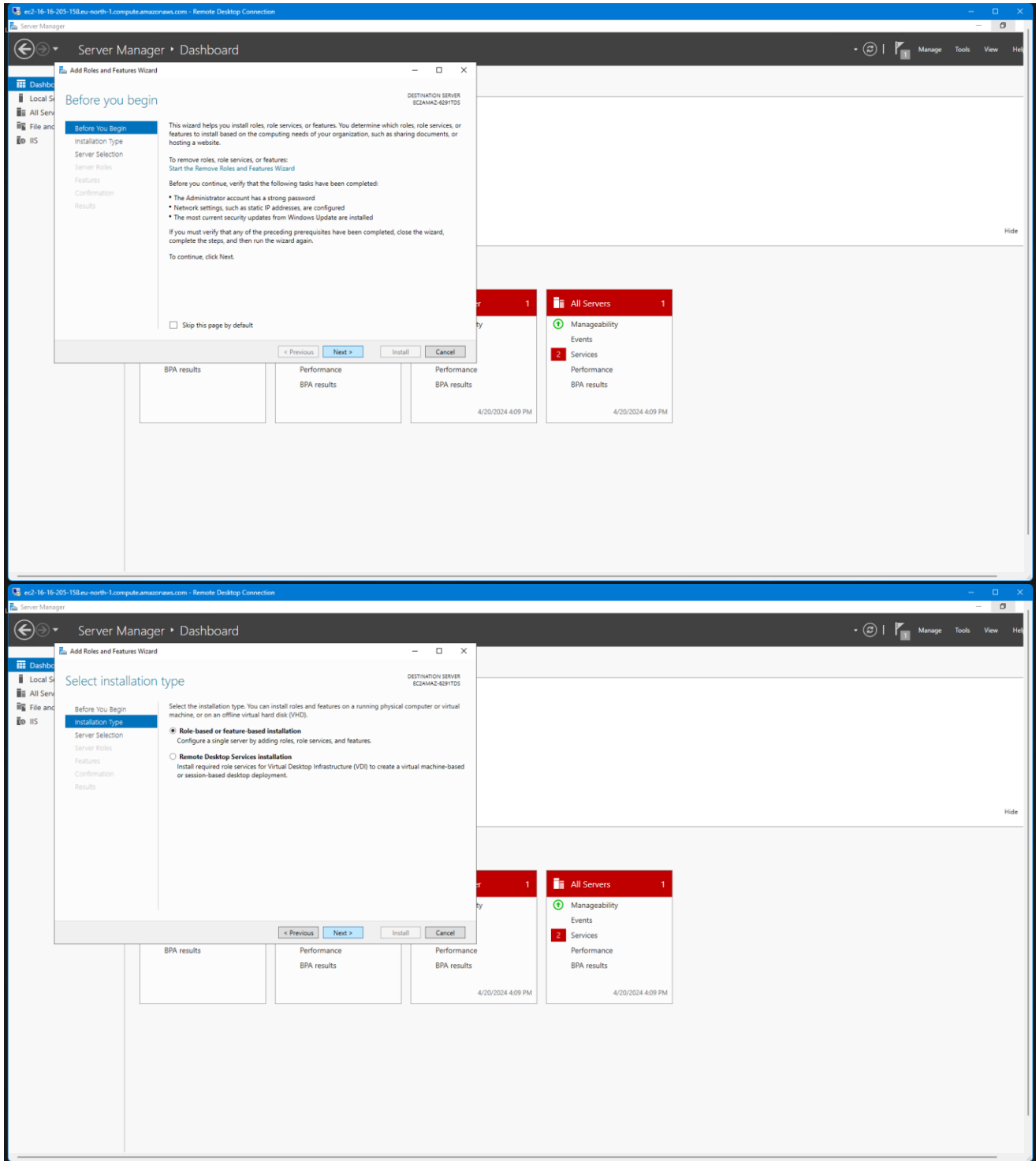
Practical: 5

5. Install Web Server Software:

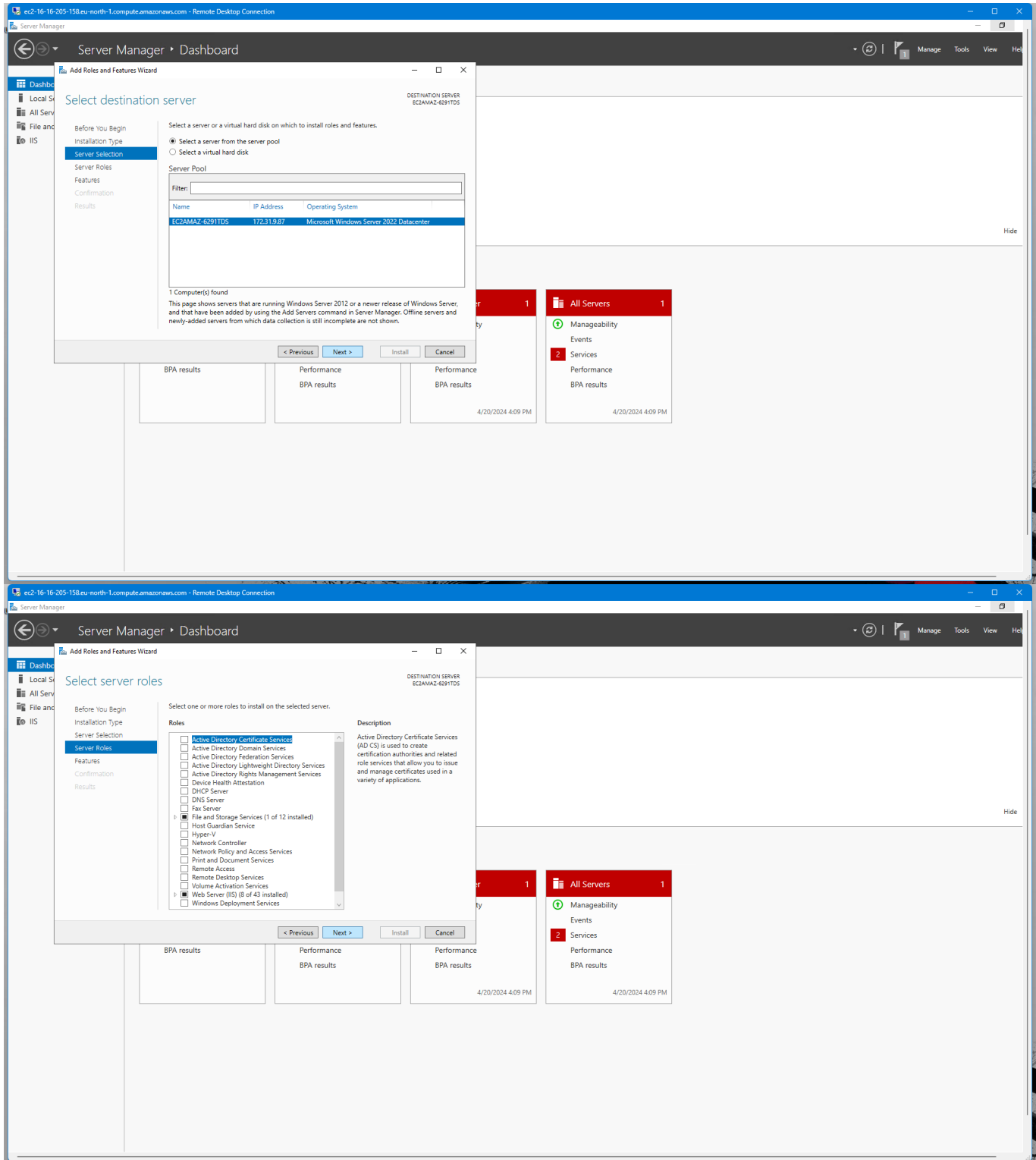
- For Windows: Install IIS (Internet Information Services) or another web server.



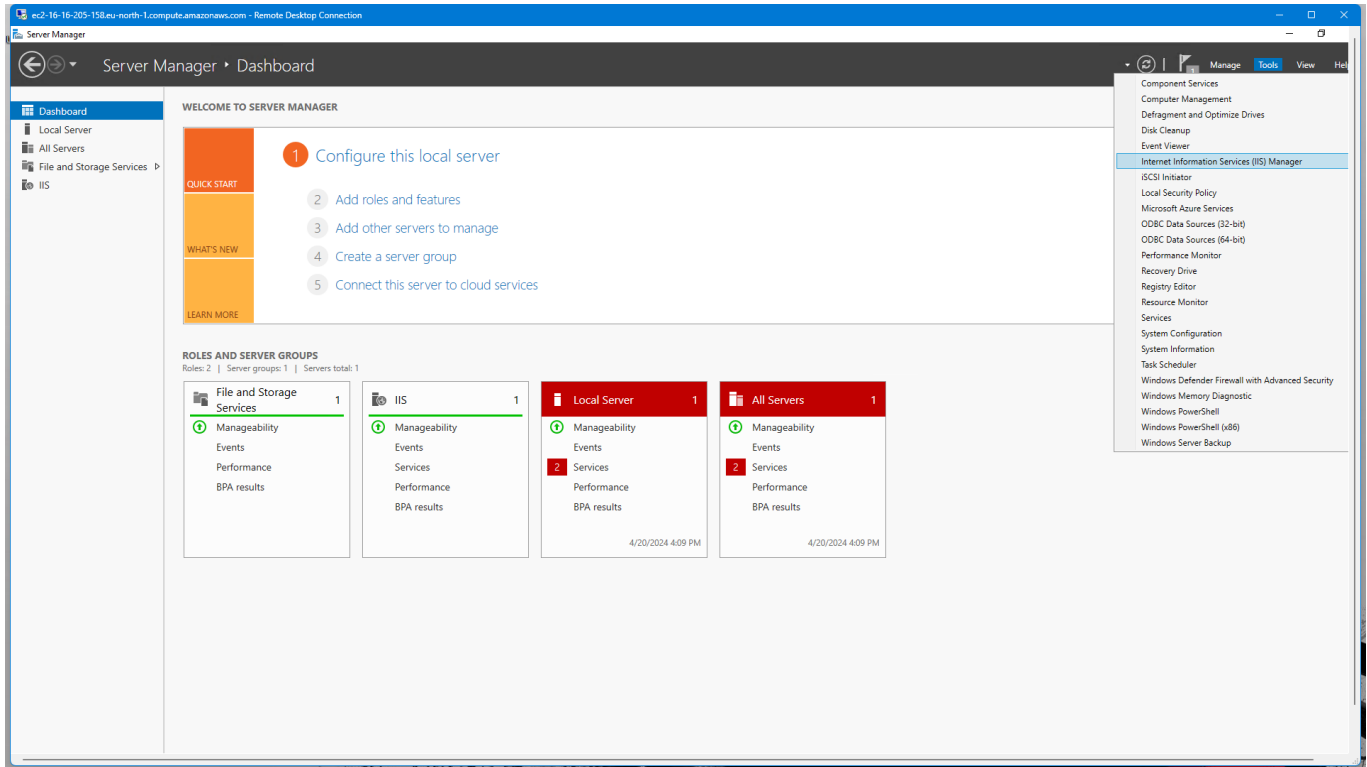
Practical: 5



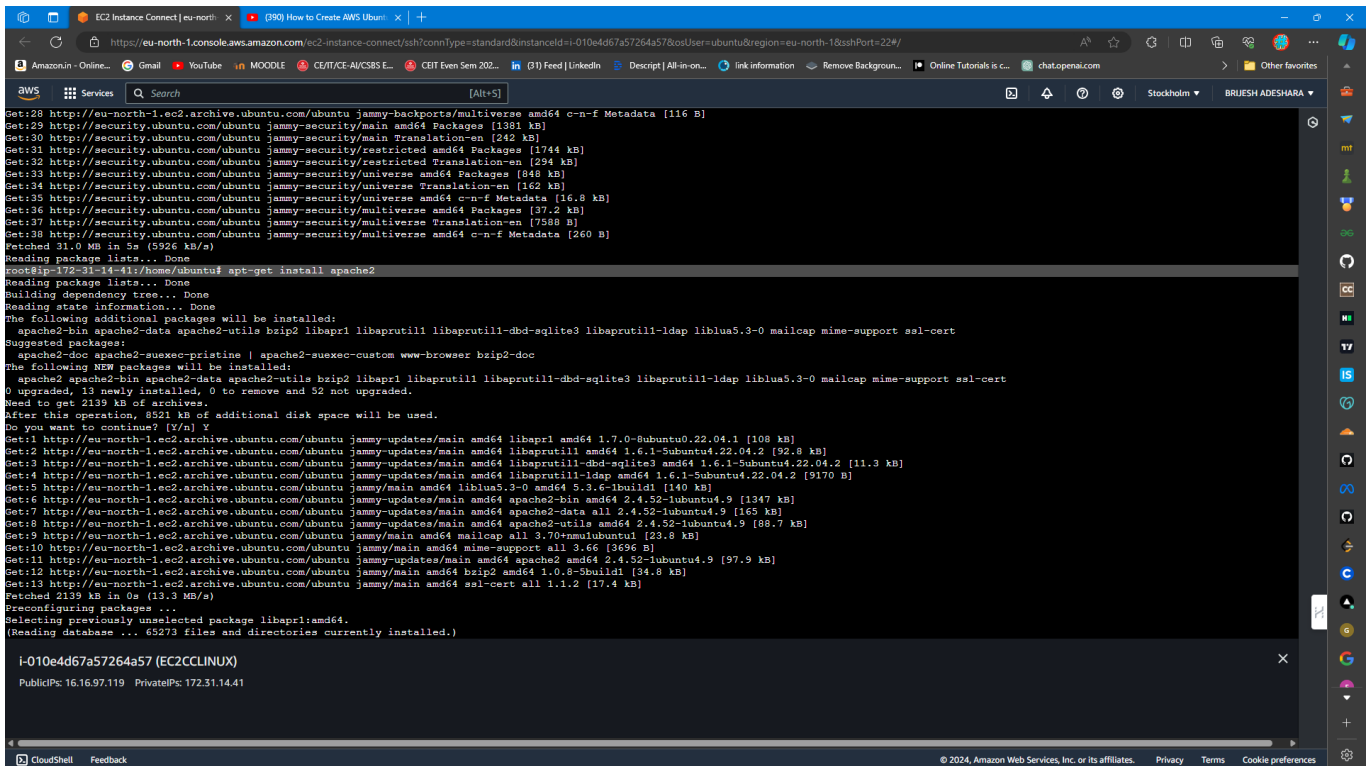
Practical: 5



Practical: 5



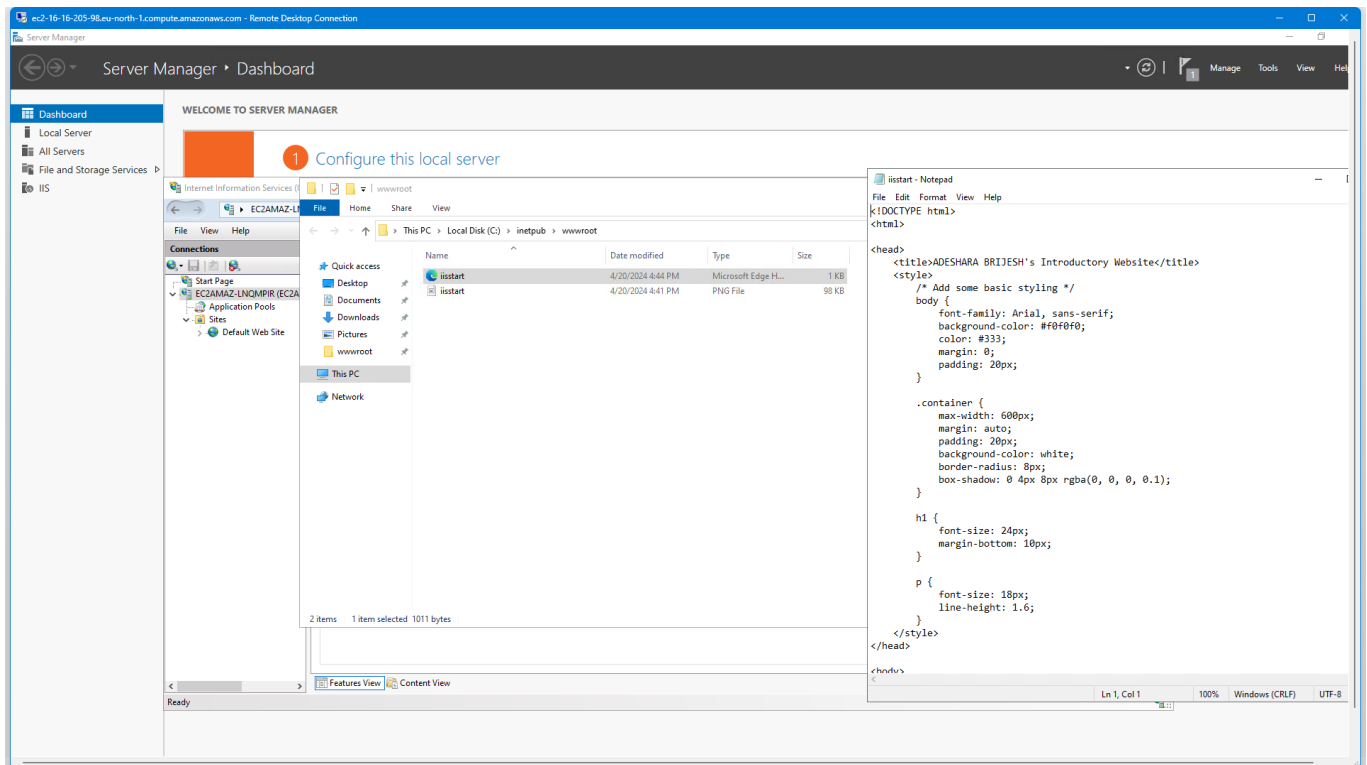
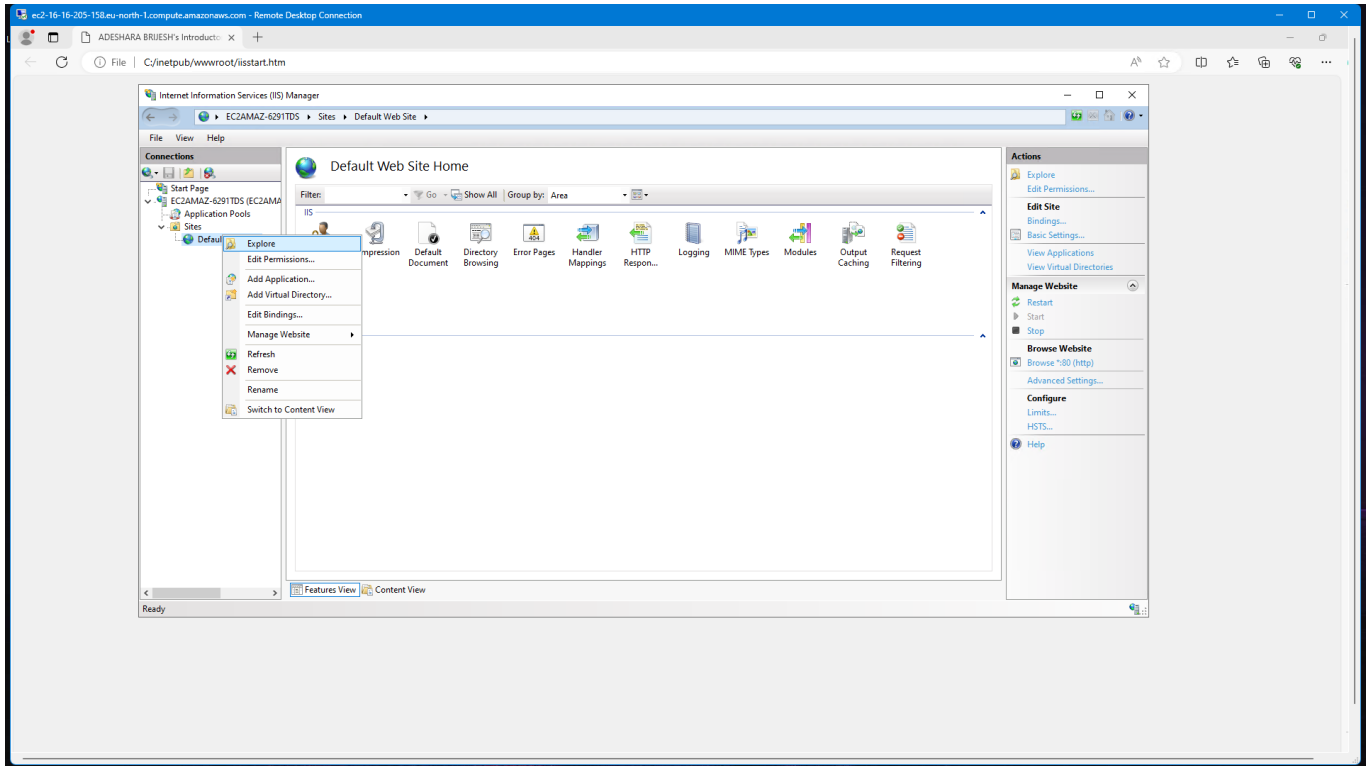
- For Linux: Install Apache, Nginx, or another preferred web server.



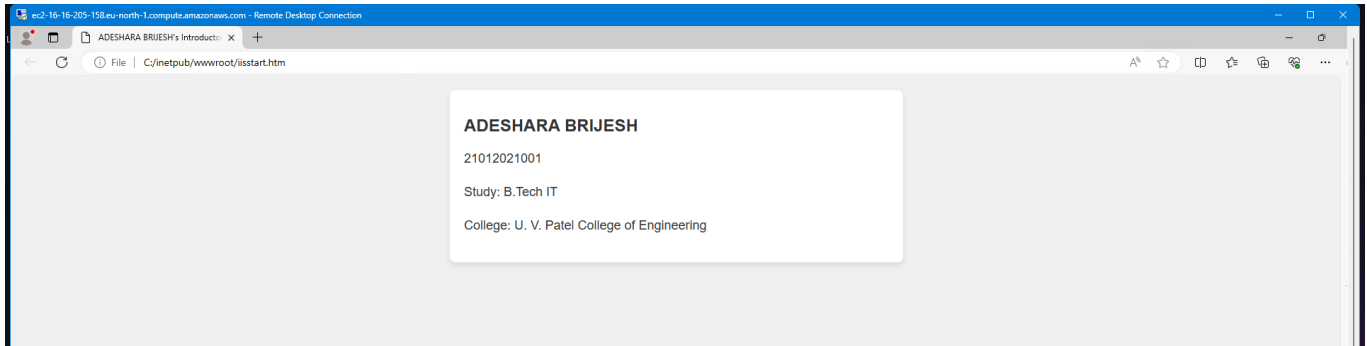
Practical: 5

6. Deploy Website Files:

- Transfer website files (HTML, CSS, JavaScript, etc.) to the appropriate directory on the EC2 instance.

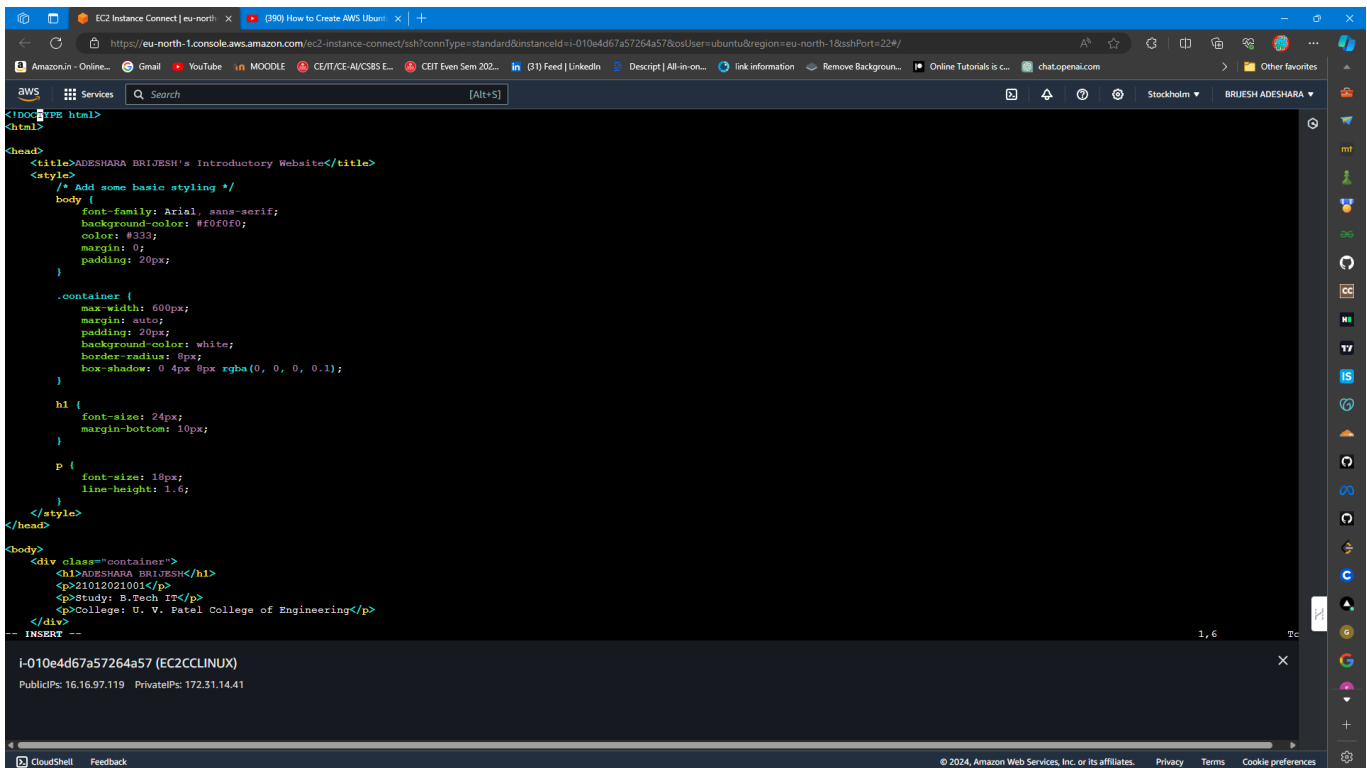


Practical: 5



- **LINUX**

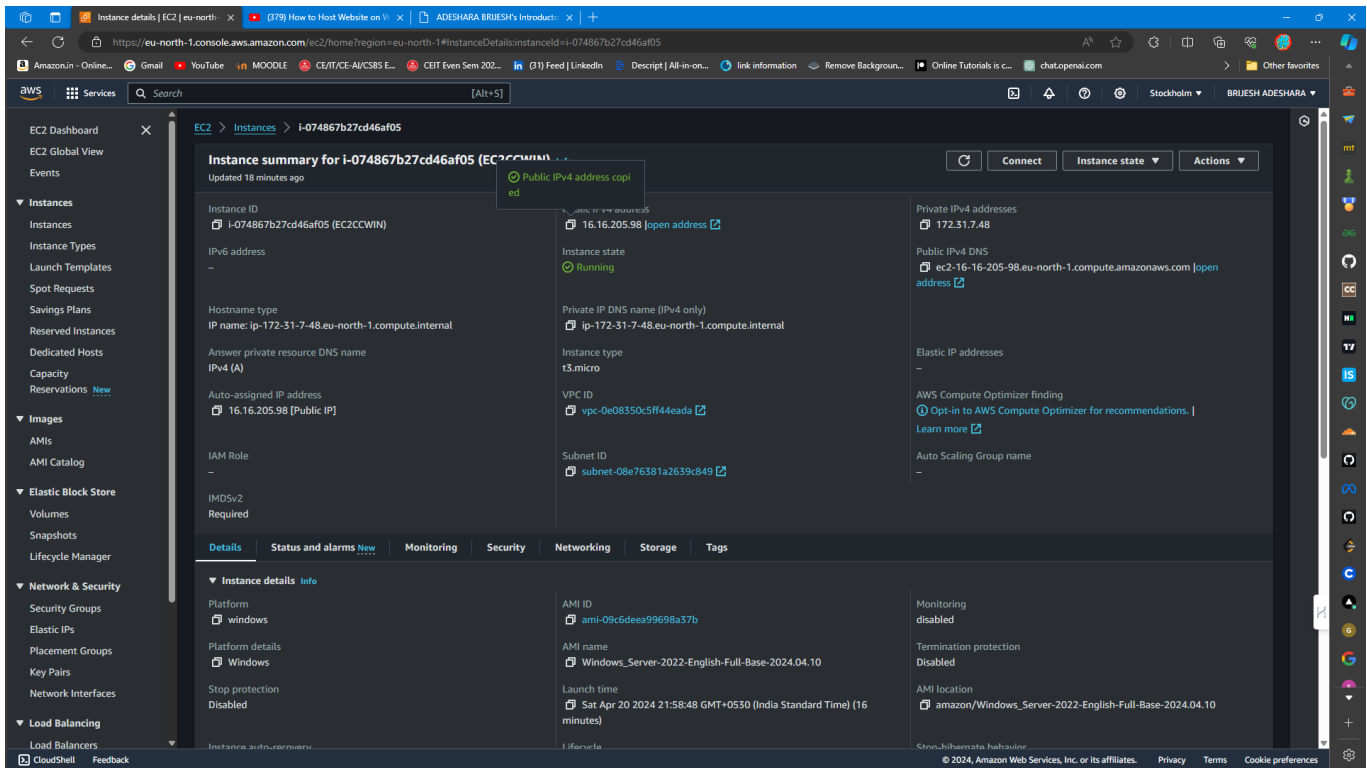
```
root@ip-172-31-14-41:/home/ubuntu# ls
root@ip-172-31-14-41:/home/ubuntu# cd /var/www/html
root@ip-172-31-14-41:/var/www/html# ls
index.html
root@ip-172-31-14-41:/var/www/html# vi index.html
root@ip-172-31-14-41:/var/www/html#
```



Practical: 5

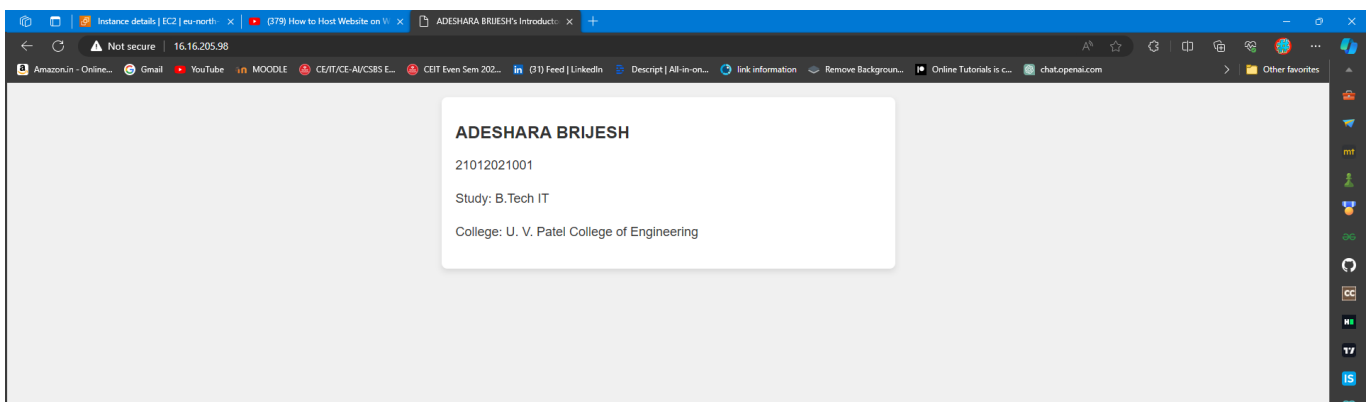
7. Configure DNS Settings:

- Update DNS records (A records or CNAME) to point to the public IP address or domain name associated with your EC2 instance.



8. Test Website Accessibility:

- Access the website using the public IP/domain name to ensure successful deployment.



Practical: 5

Comparison:

| Aspect | Amazon EC2 | Traditional Hosting | Other Cloud Providers |
|-------------|--|---|--|
| Scalability | Highly scalable, allowing resources to be scaled up or down based on demand. | Limited scalability, typically offering fixed hosting plans with limited resources. | Varies among providers, with some offering scalability similar to EC2 and others with limitations. |
| Flexibility | Offers a wide range of instance types, configurations, and additional services for customization. | Limited flexibility, with predefined hosting plans and configurations. | Flexibility varies among providers, with some offering customizable solutions comparable to EC2 and others with more restricted offerings. |
| Cost | Pay-per-use model, allowing users to pay only for the resources they consume, potentially cost-effective for varying workloads. | Often fixed monthly fees, which may become costly for unused resources or sudden spikes in traffic. | Pricing structures vary among providers, with some offering competitive pricing comparable to EC2 and others with different pricing models. |
| Ease of Use | Requires knowledge of AWS services and configurations but provides extensive documentation and management tools for ease of use. | Typically easier to set up and manage, with user-friendly control panels and customer support. | Ease of use varies among providers, with some offering intuitive interfaces and support comparable to EC2 and others with different levels of usability. |

Conclusion:

- Hosting websites on Amazon EC2 instances provides scalable and reliable solutions leveraging cloud infrastructure.
- This project enhances understanding of cloud computing principles and practices in web hosting.

Future Considerations:

- Explore advanced features of Amazon EC2 (e.g., auto-scaling, load balancing).
- Integrate additional AWS services (e.g., Amazon RDS for database hosting) for enhanced website functionality.
- Stay updated on evolving cloud technologies to optimize website hosting solutions.