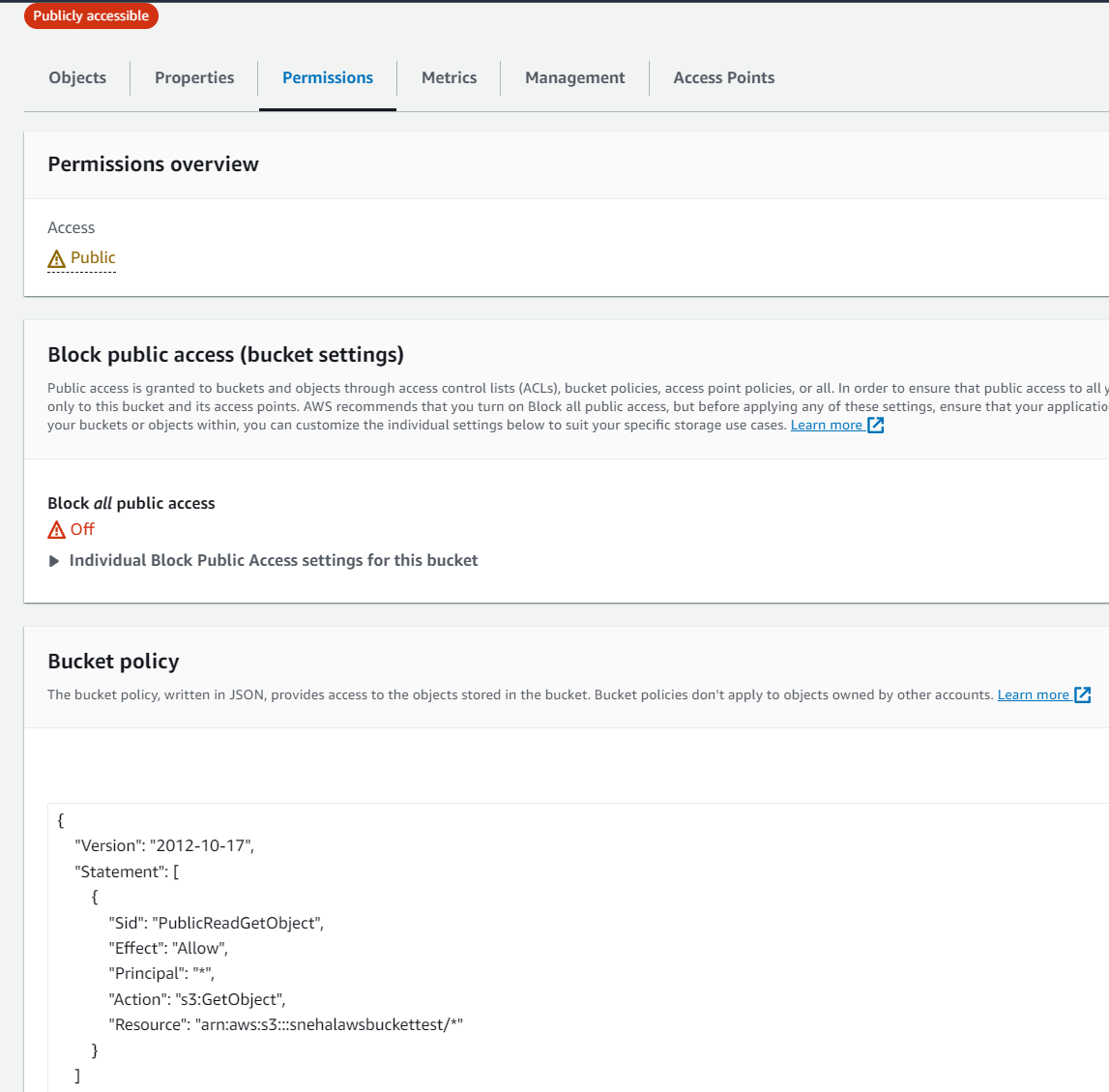
**AWS Projects**

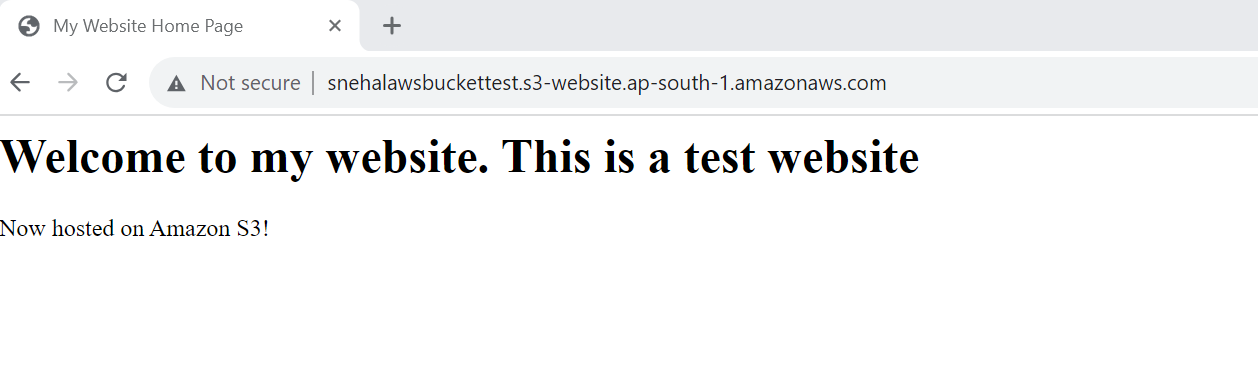
1. **Building a Personal Website**: Students will create a simple static website hosted on Amazon S3. This project serves as an introduction to AWS and basic web hosting.

-- Create an S3 bucket.

Permission for public access needs to be enabled.

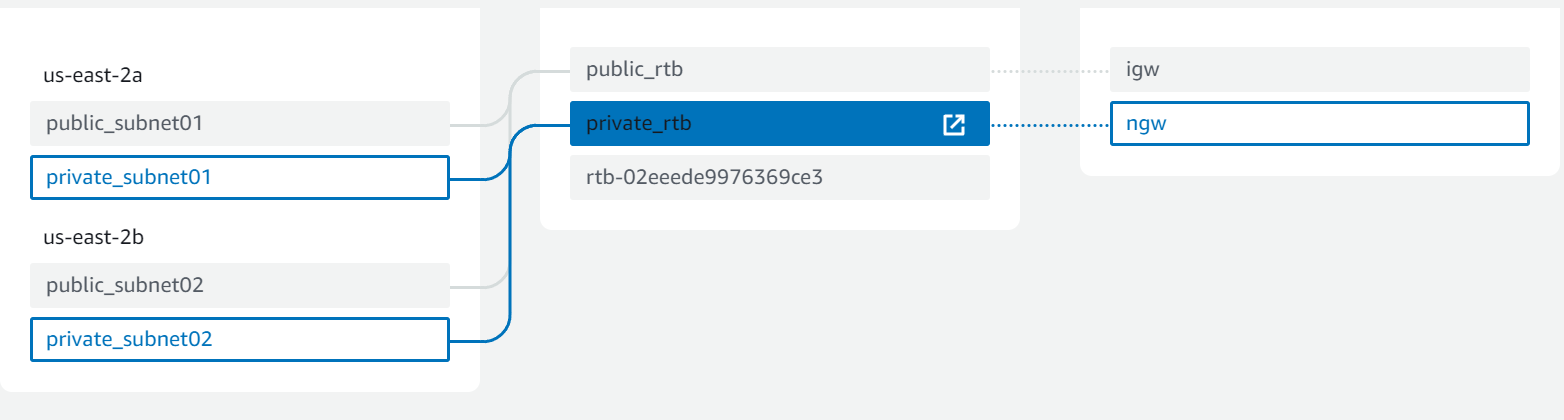


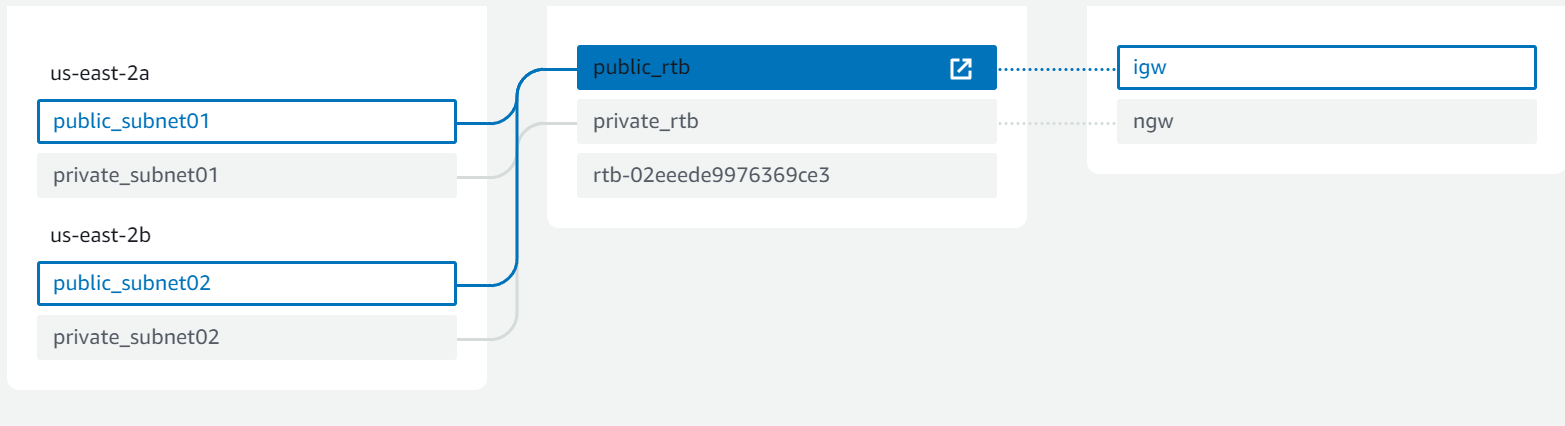
Enable the Static website hosting, by uploading the html document



2. **Setting Up a Virtual Private Cloud (VPC):** This project guides students through the process of setting up a custom VPC with subnets, route tables, and security groups. It is crucial for understanding networking in AWS.

-- Creating VPC (virtual private cloud)





Step 1. Creation of Manual VPC

10.2.0.0/22

Step 2. Creating the subnets

Private\_subnet01 10.2.0.0/24

Private\_subnet02 10.2.1.0/24

Public\_ subnet 01 10.2.2.0/24

Public\_ subnet02 10.2.3.0/24

Step 3. Internet Gateway – create – attach to VPC

Step 4. Nat Gateway – create and allocate elastic IP – create in Public for internet connectivity

Step 5. Route table

5.1 Private –

Allocating with Private\_subnet01 10.2.0.0/24 and

Private\_subnet02 10.2.1.0/24

5.2 Public

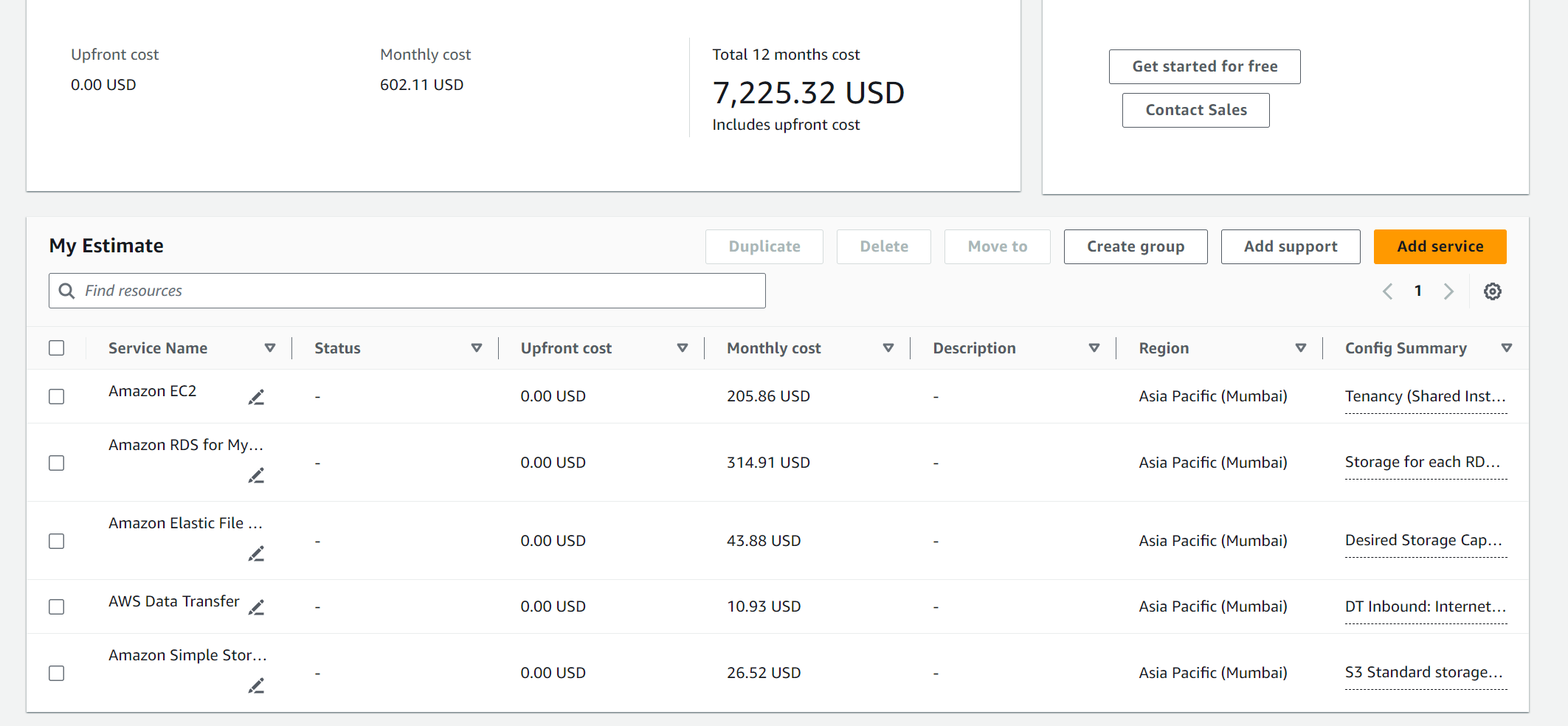
Allocating with

Public\_ subnet 01 10.2.2.0/24

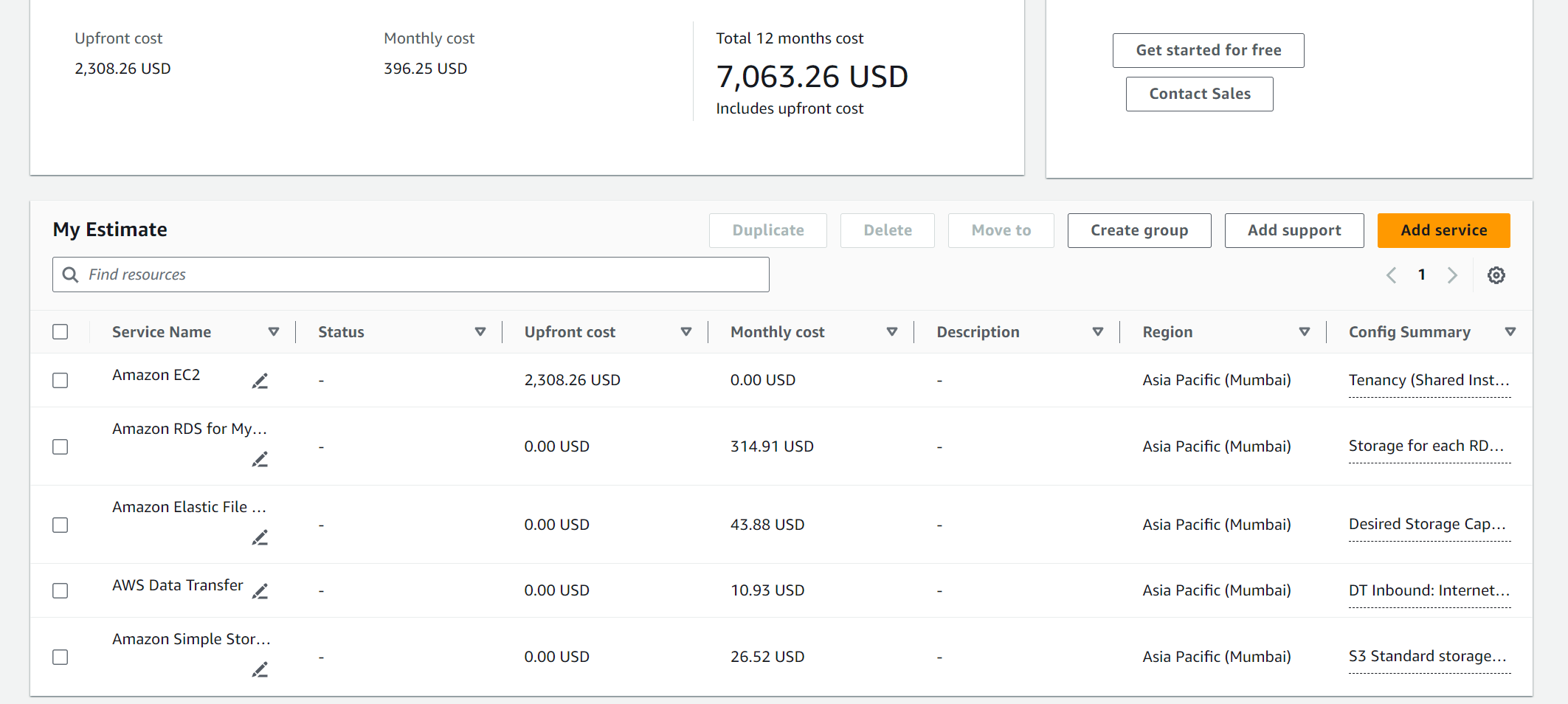
Public\_ subnet02 10.2.3.0/24

3. **Cost Optimization Analysis:** Students will be tasked with analysing AWS cost management tools and optimizing a sample AWS environment to reduce costs while maintaining performance.

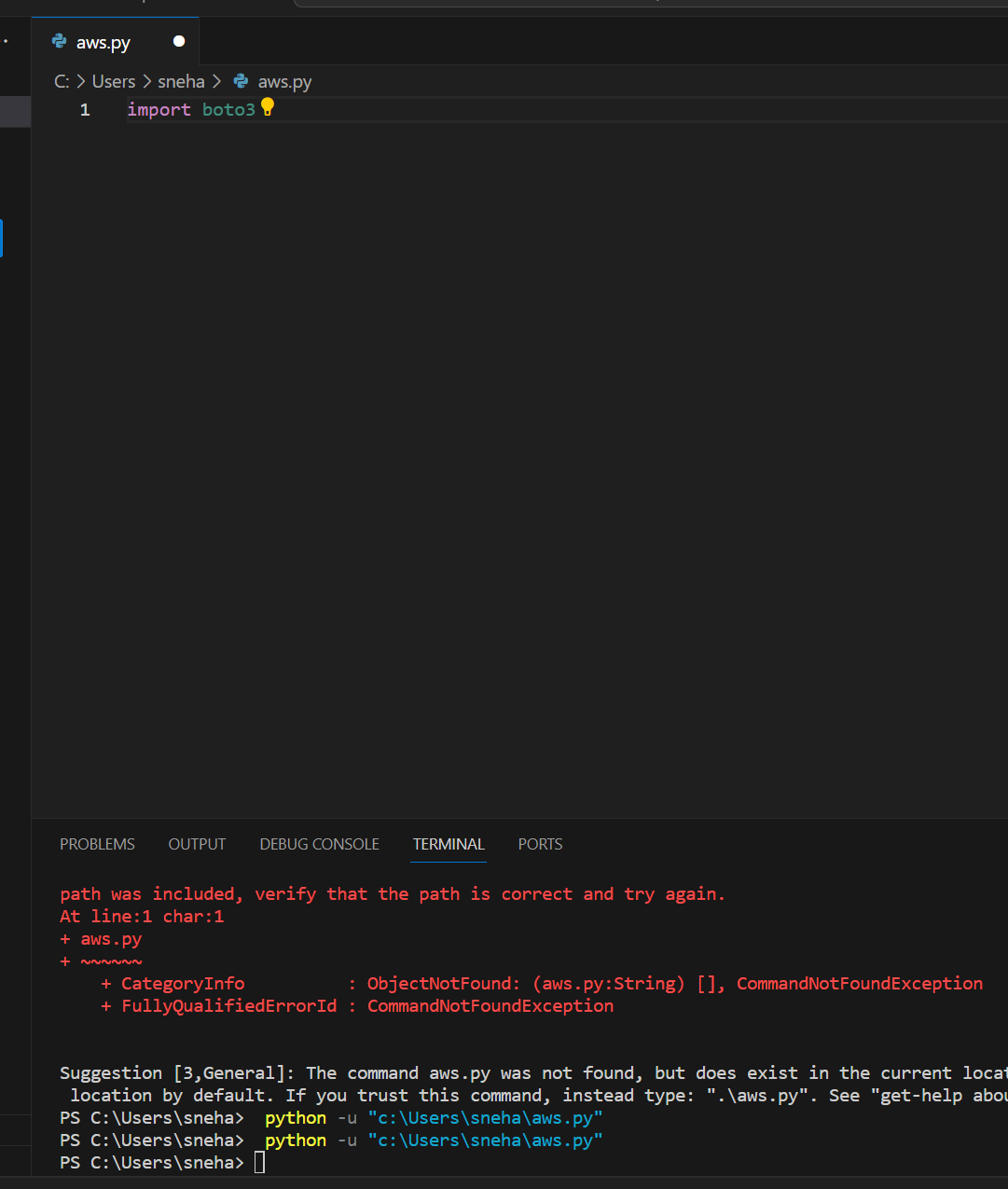
---Savings Plan with No upfront cost



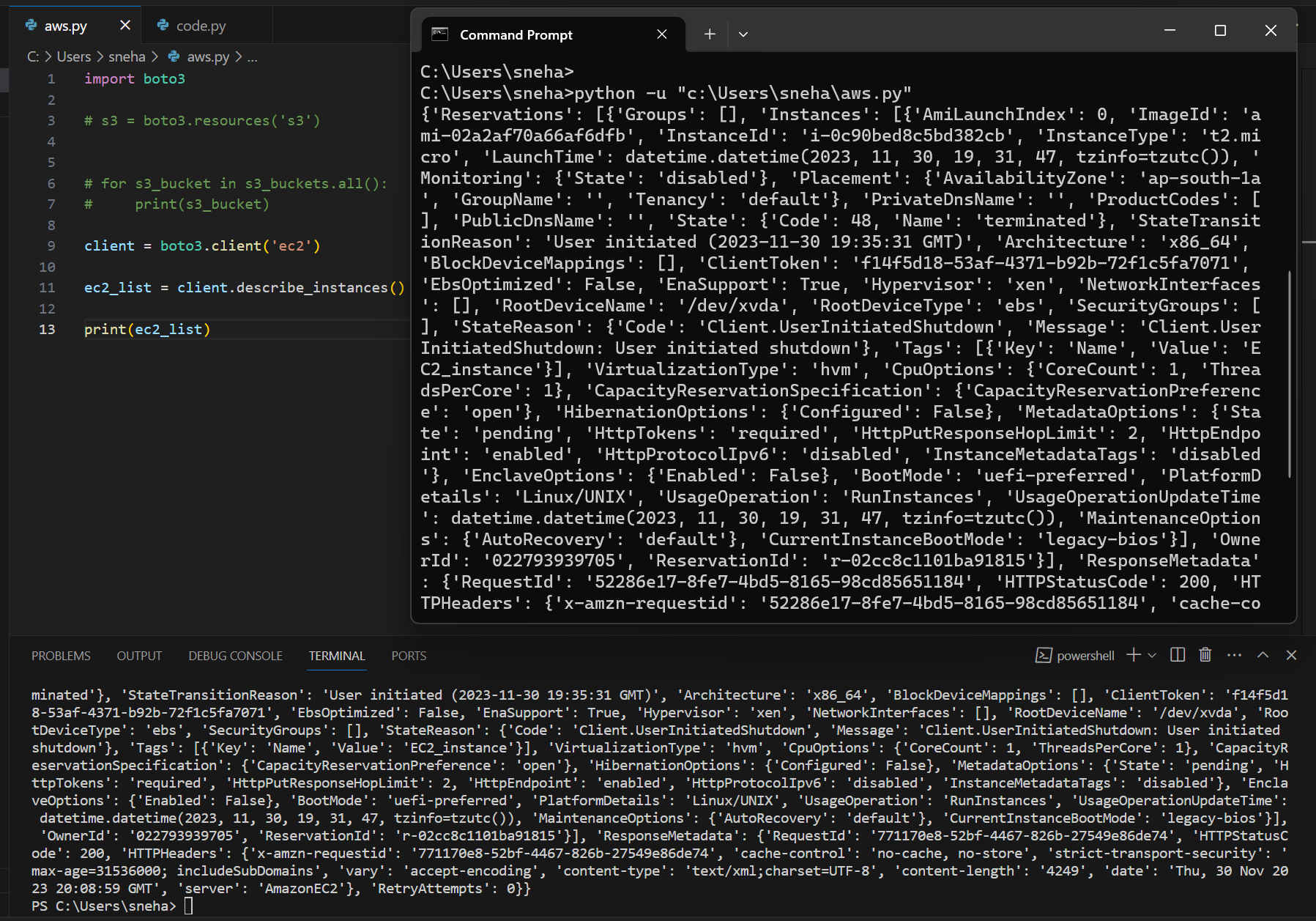
---Savings Plan with all upfront cost.



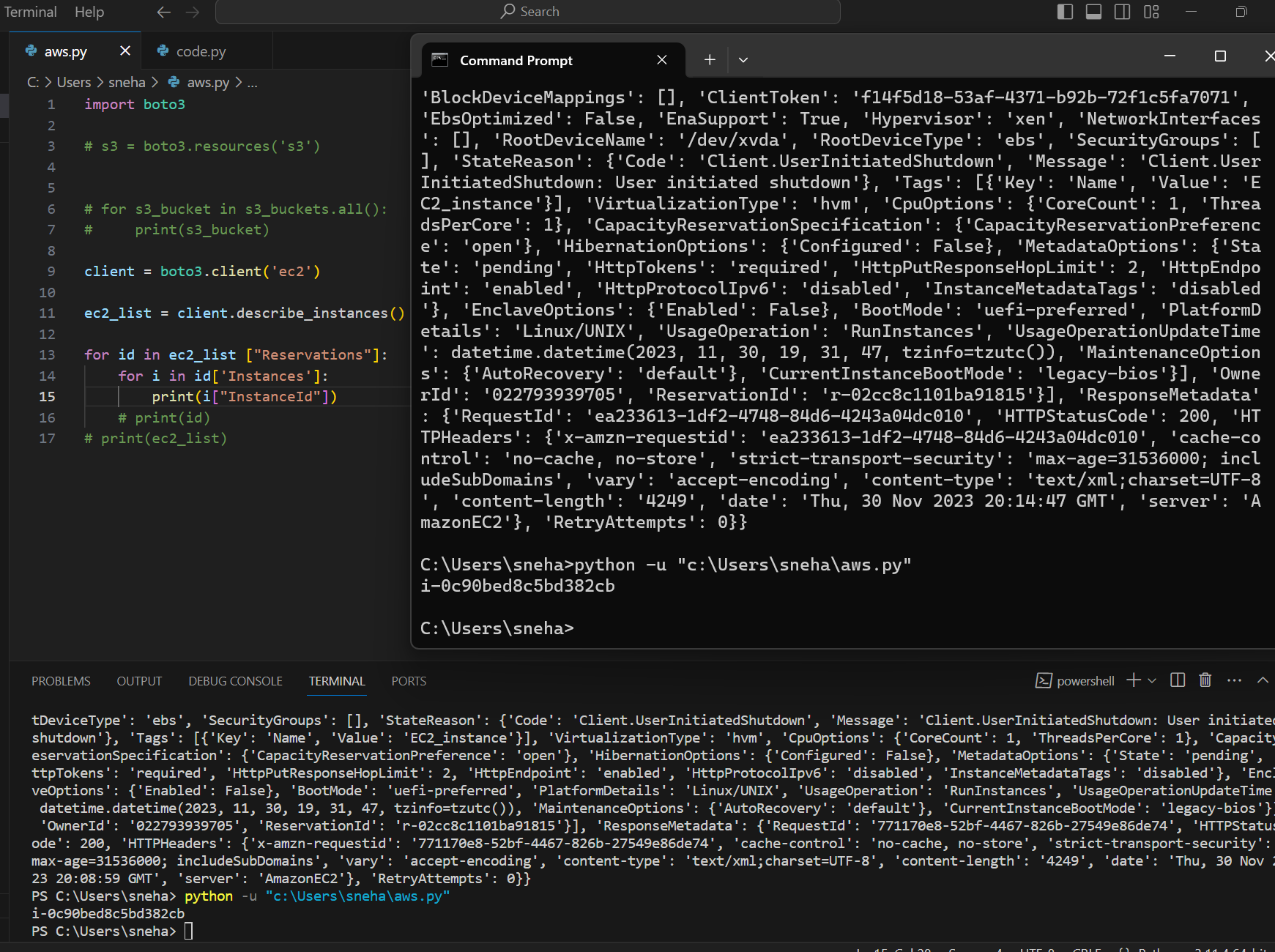
4. **Python Script for AWS**: In this project, students will develop a Python script that interacts with AWS services using Boto3, gaining hands-on experience with AWS SDKs.



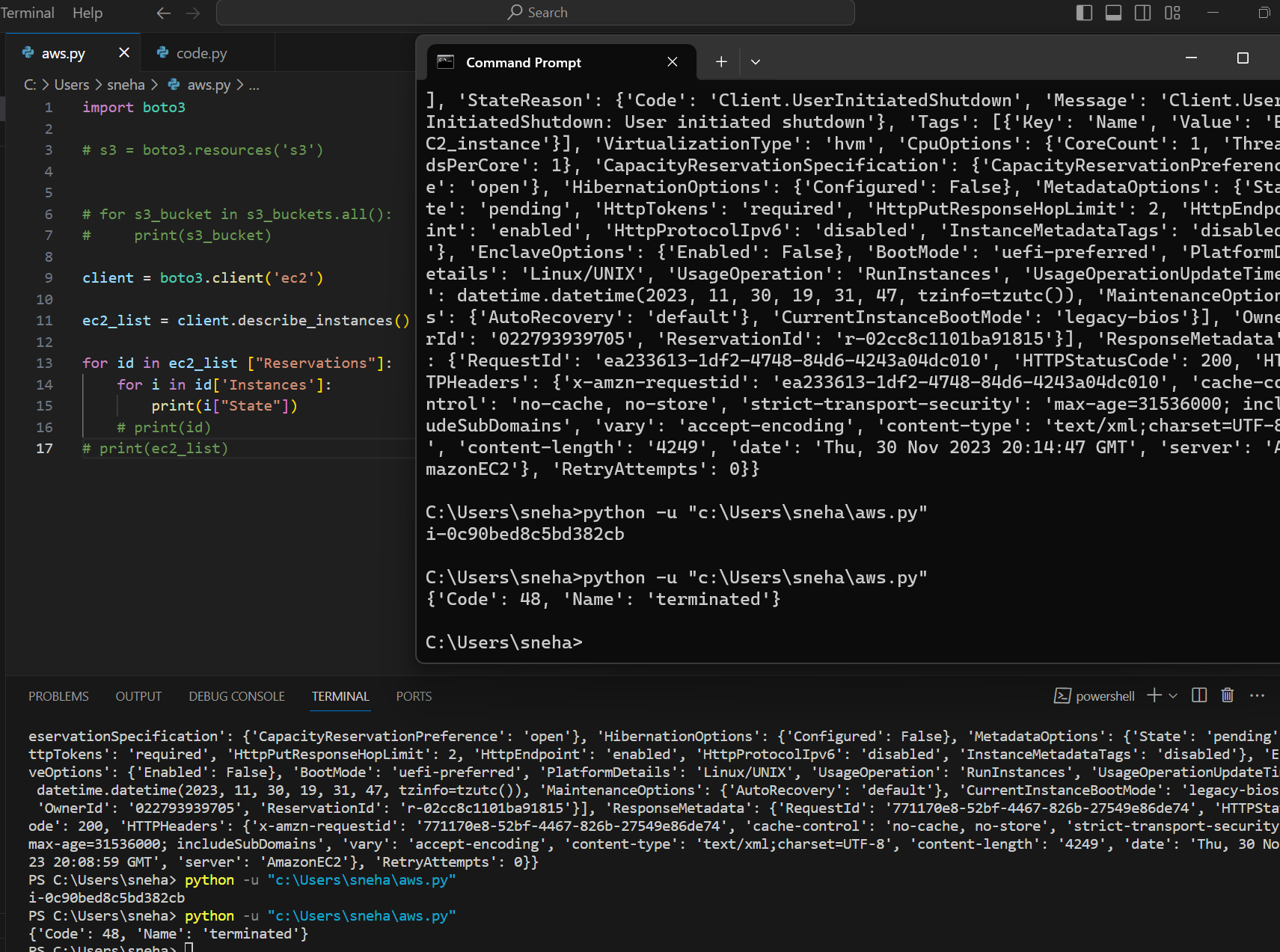
I was able to get the list of ALL EC2 instances -



To fetch a particular instance id -

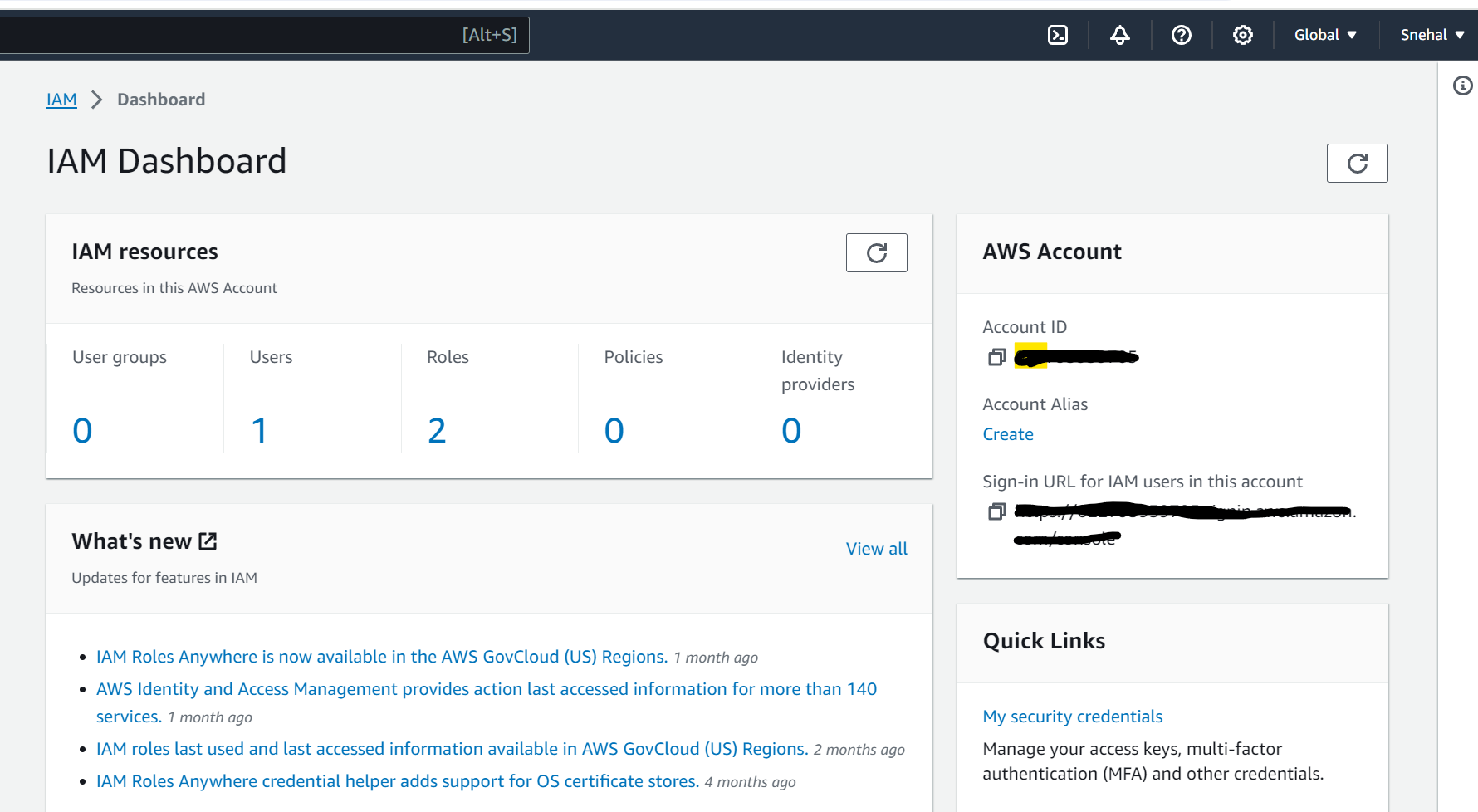


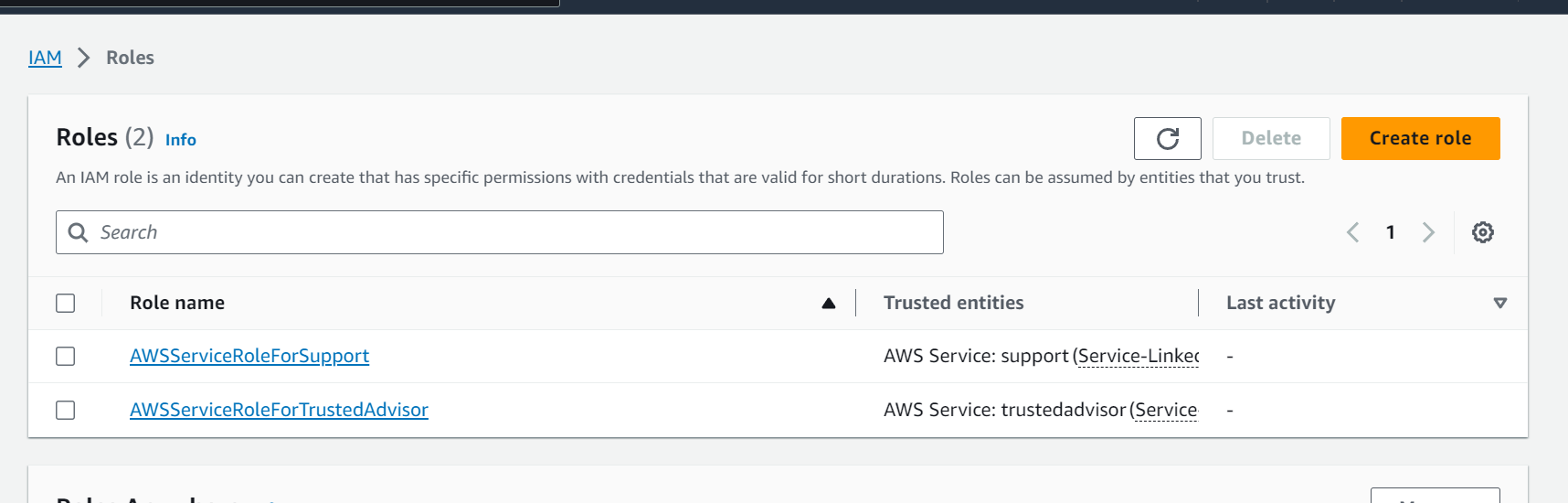
To get the state of an Instance -



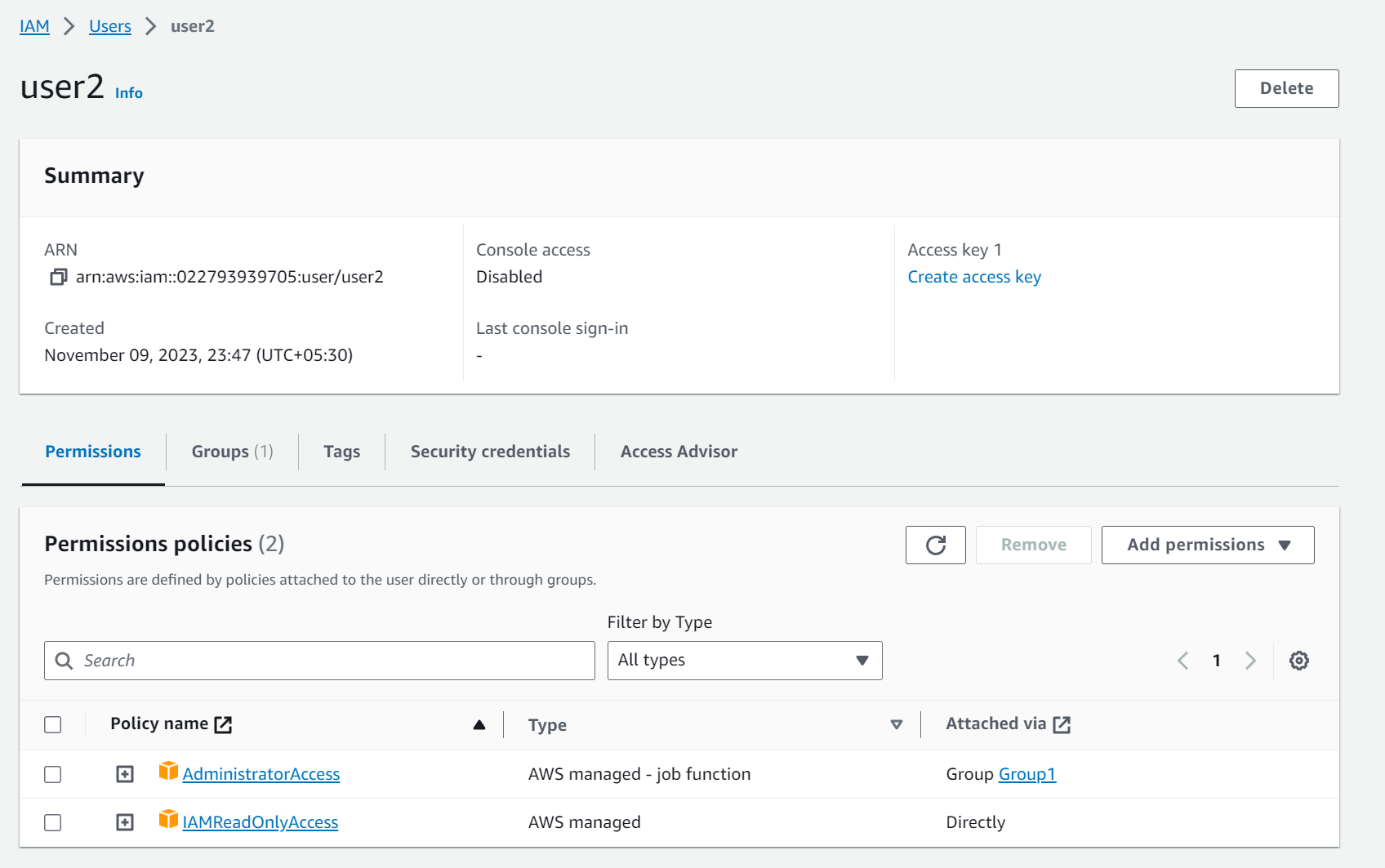
5. **IAM and Access Control**: This project involves creating IAM users, roles, and policies to manage access control within an AWS environment.

User-

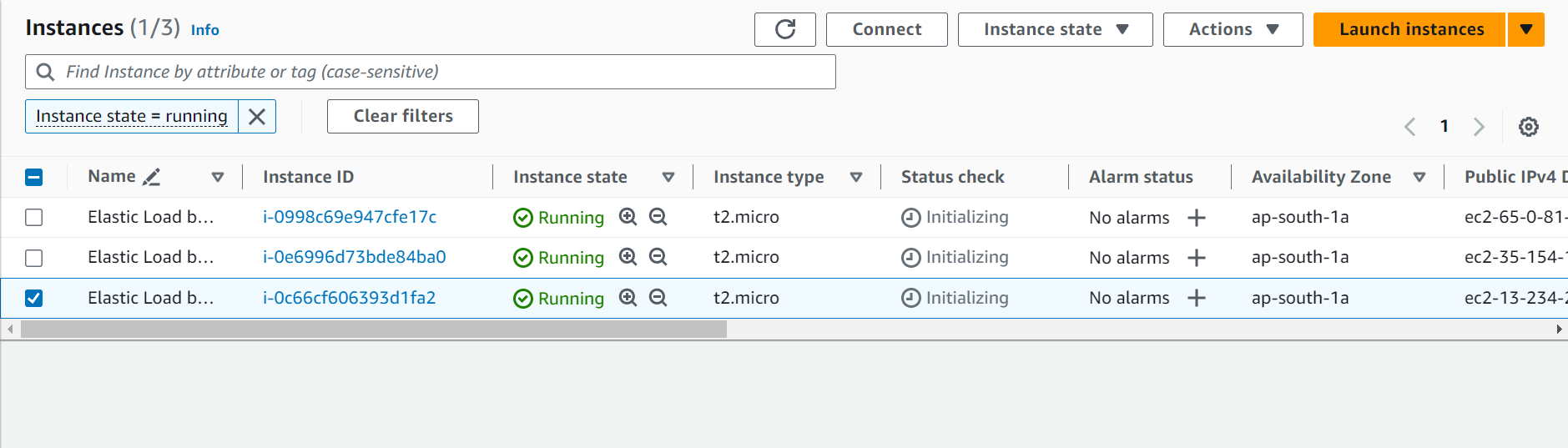


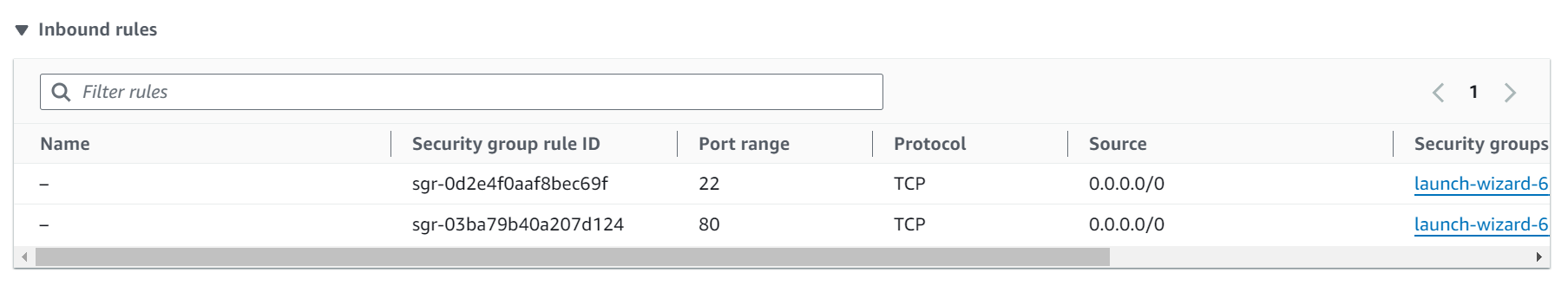
Roles - 

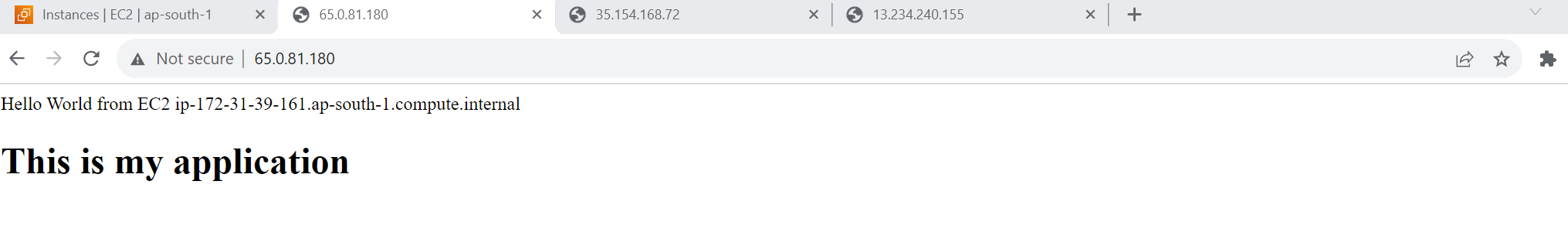
Policies-

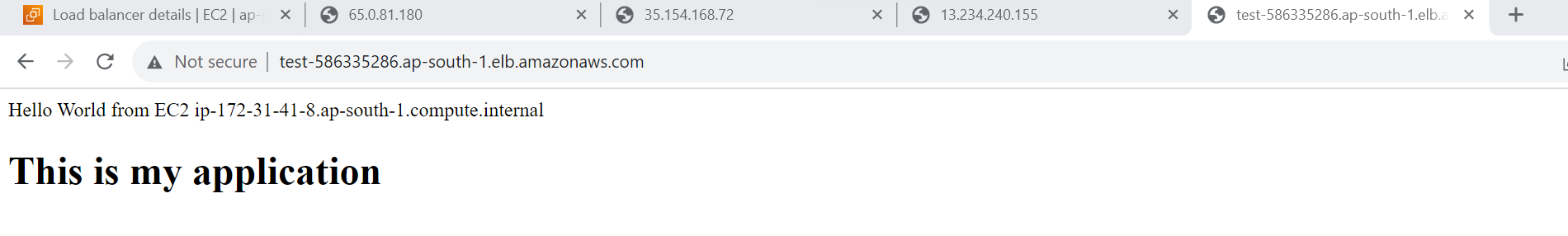


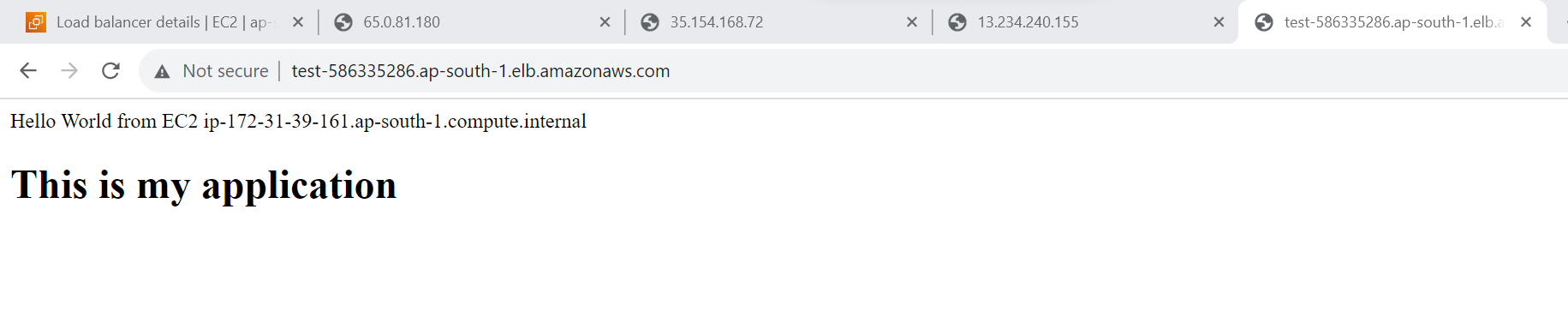
6. **Auto Scaling and Load Balancing**: Students will set up auto scaling and load balancing for a web application, ensuring high availability and efficient resource usage.

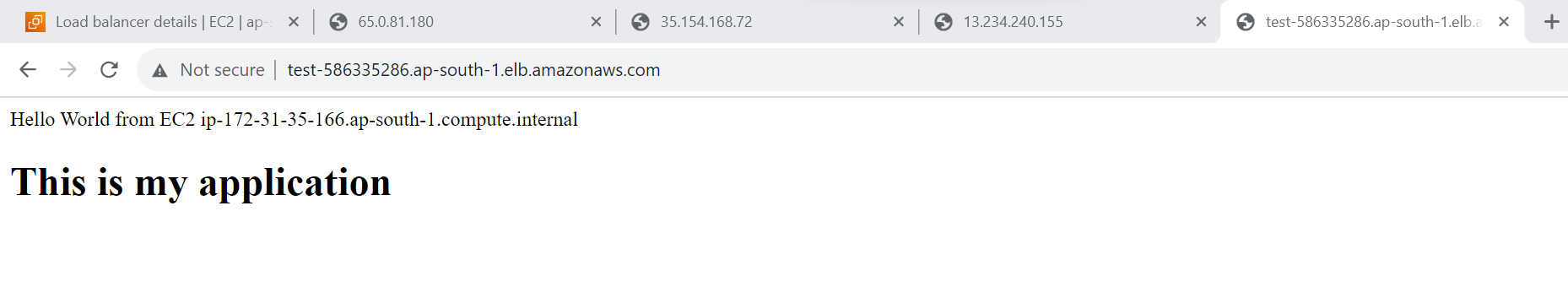






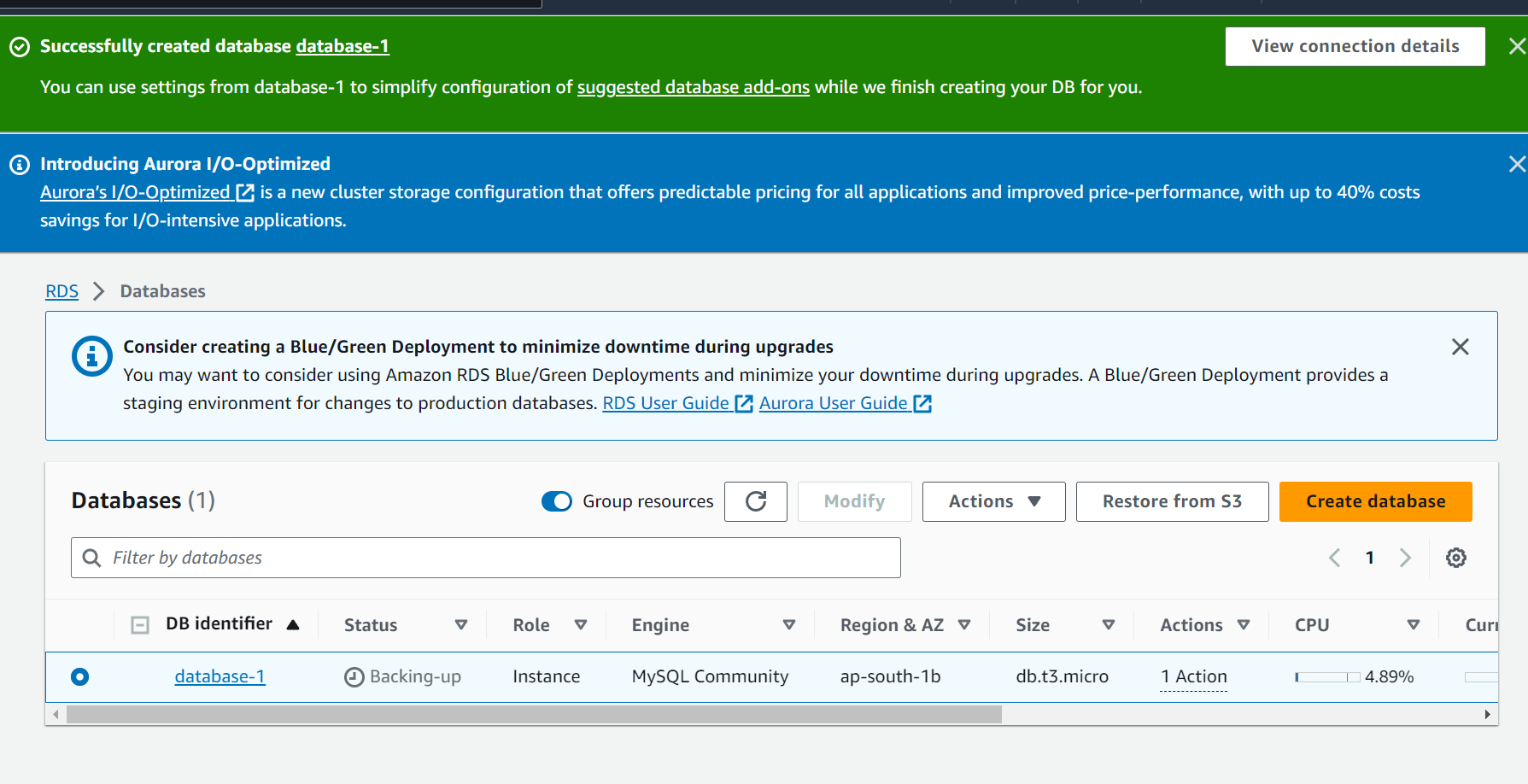


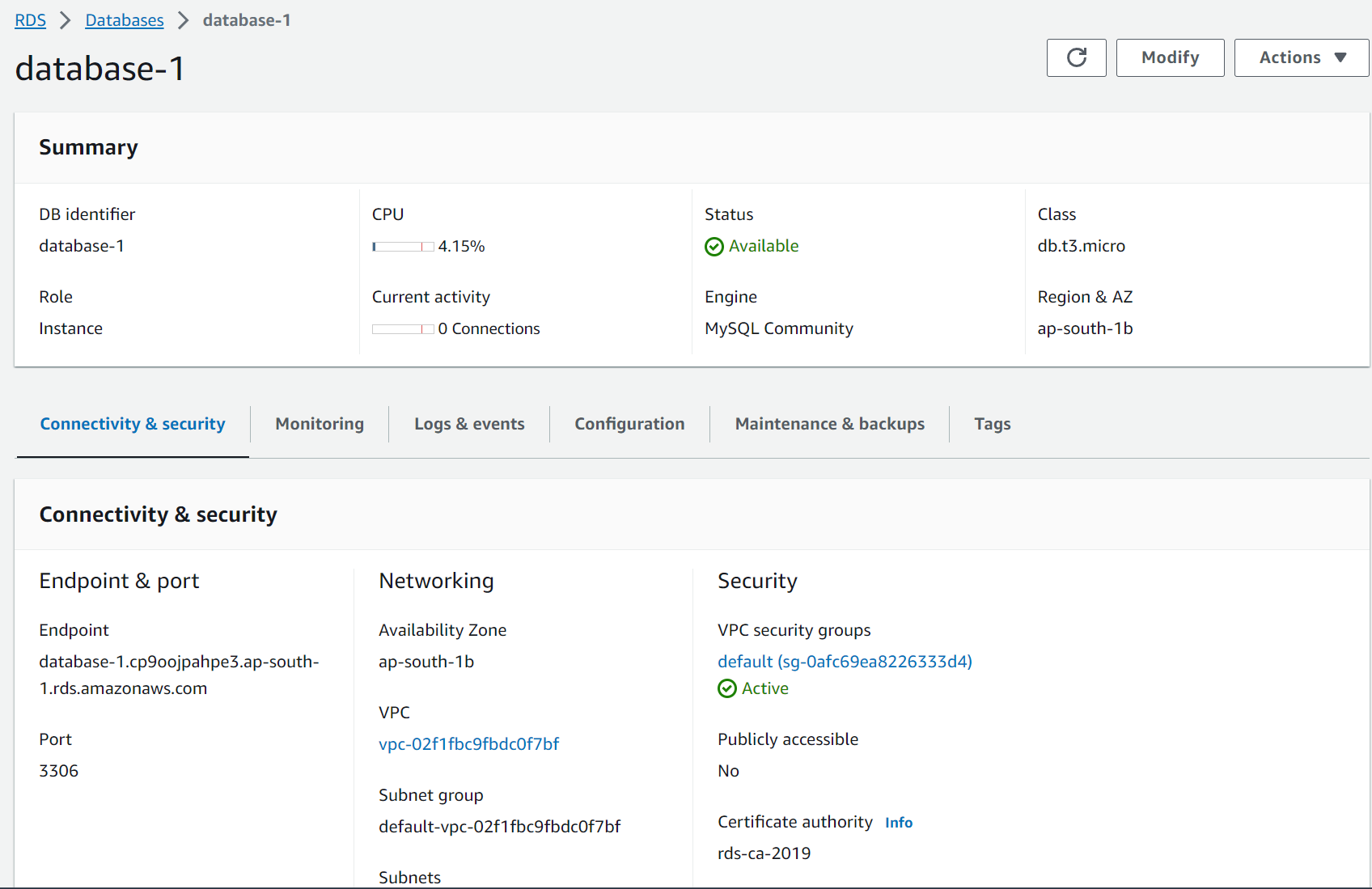




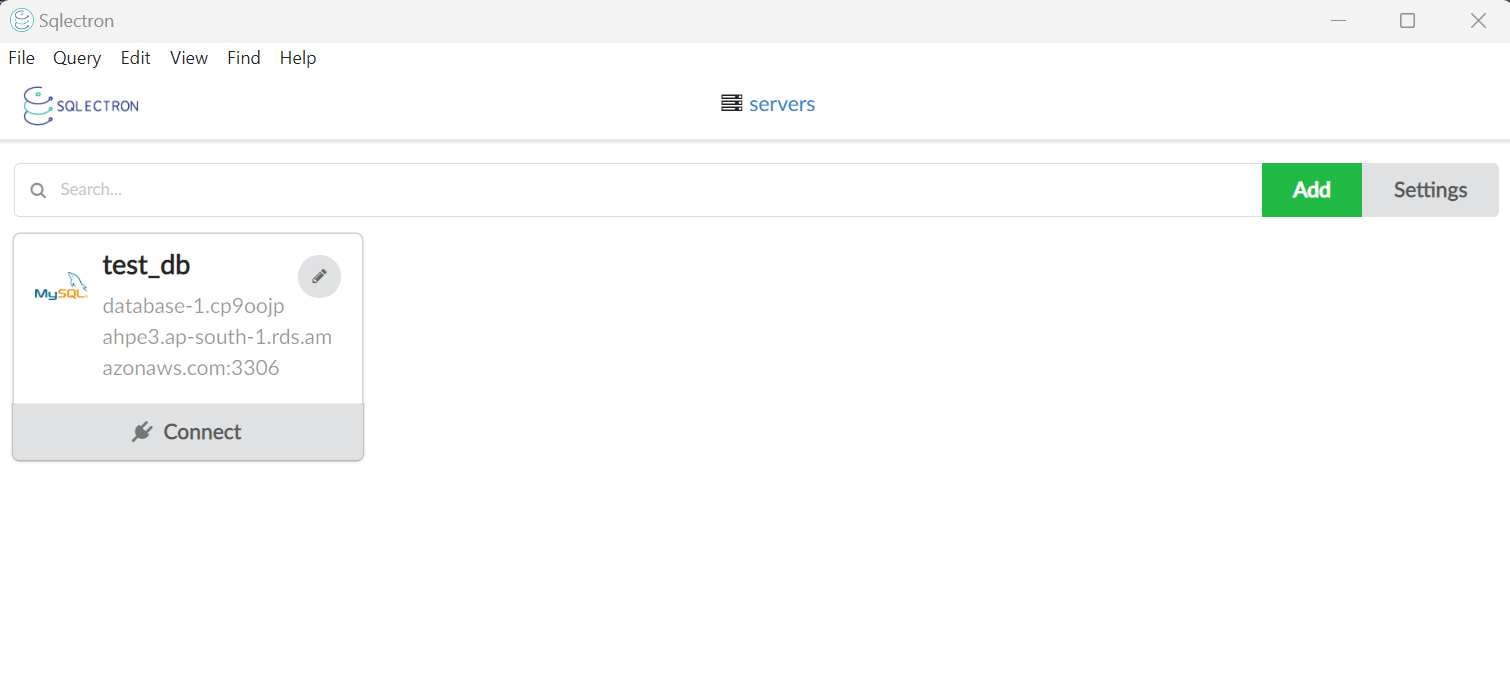
7. **Amazon RDS Deployment**: This project focuses on deploying and configuring a relational database using Amazon RDS, exploring multi-AZ deployments and read replicas.

--- Created database with the help on AWS RDS





Access DB with the help of SQelectron through RDS endpoint



8. **VPC Peering and Transit Gateway**: In this project, students will implement VPC peering and transit gateway to connect multiple VPCs and enable secure communication between them.

---We need to peer the two different VPC

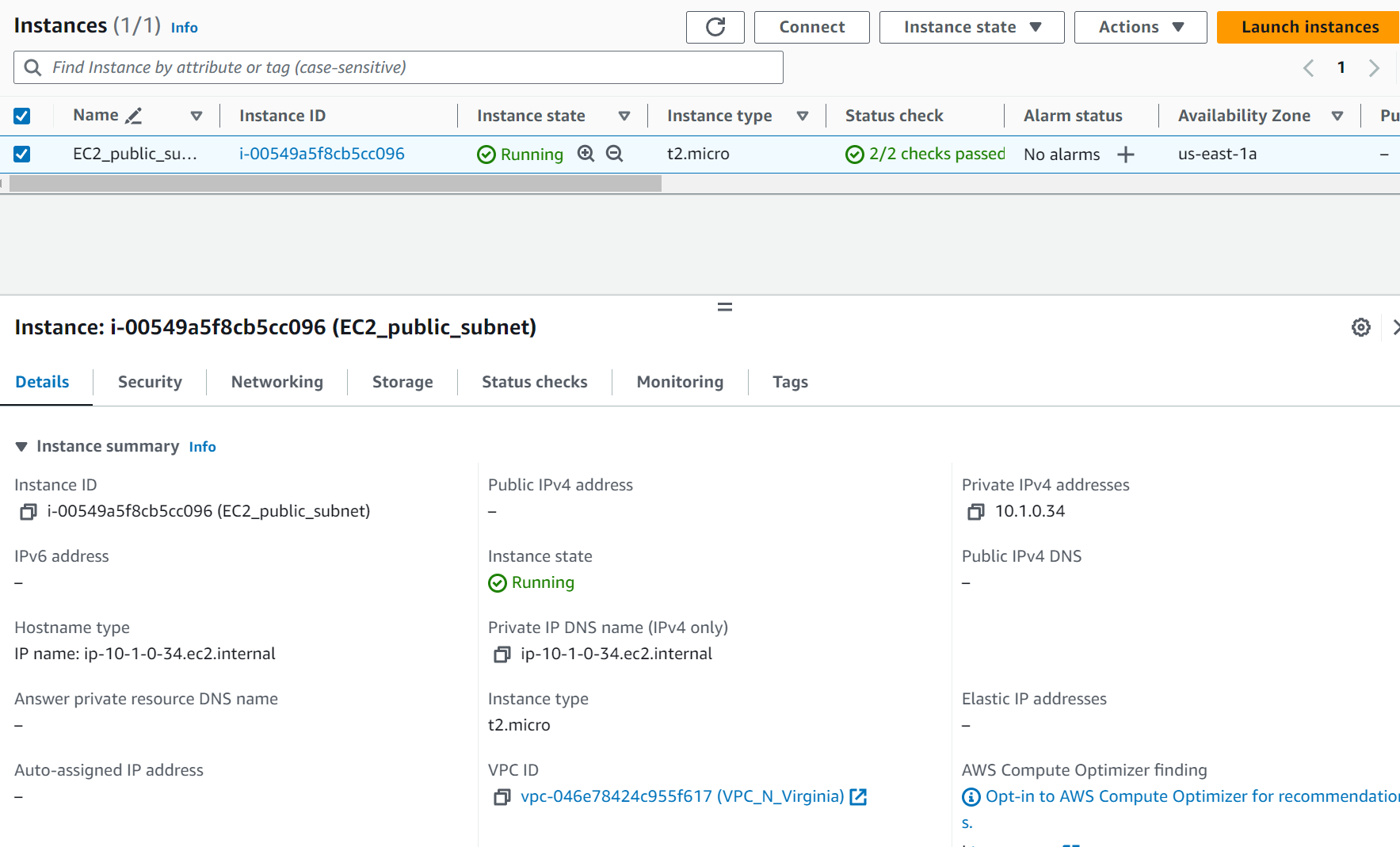
10.1.0.0/22 (N Virginia)

10.2.0.0/22(Ohio)

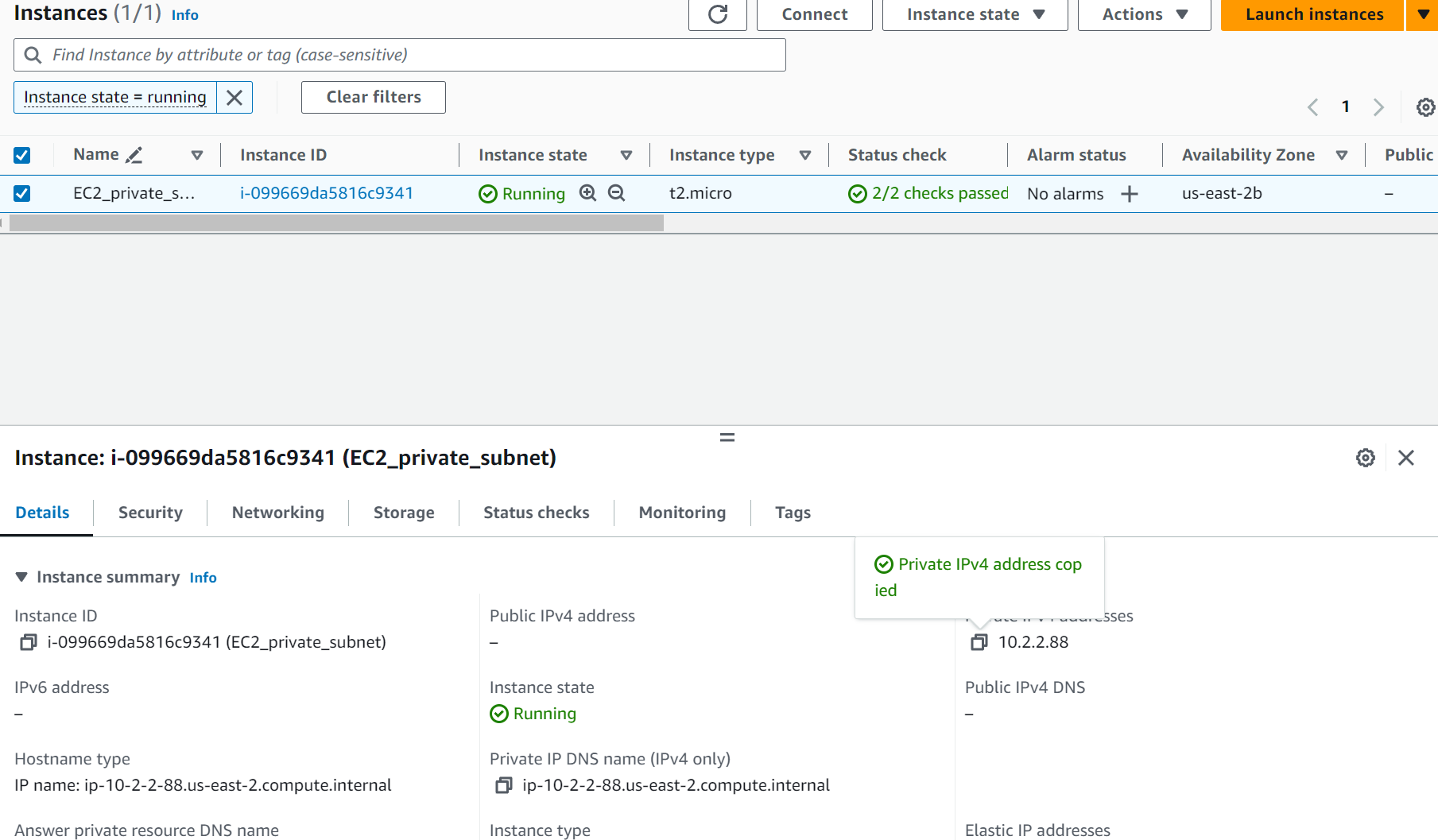
Modifying the Route tables

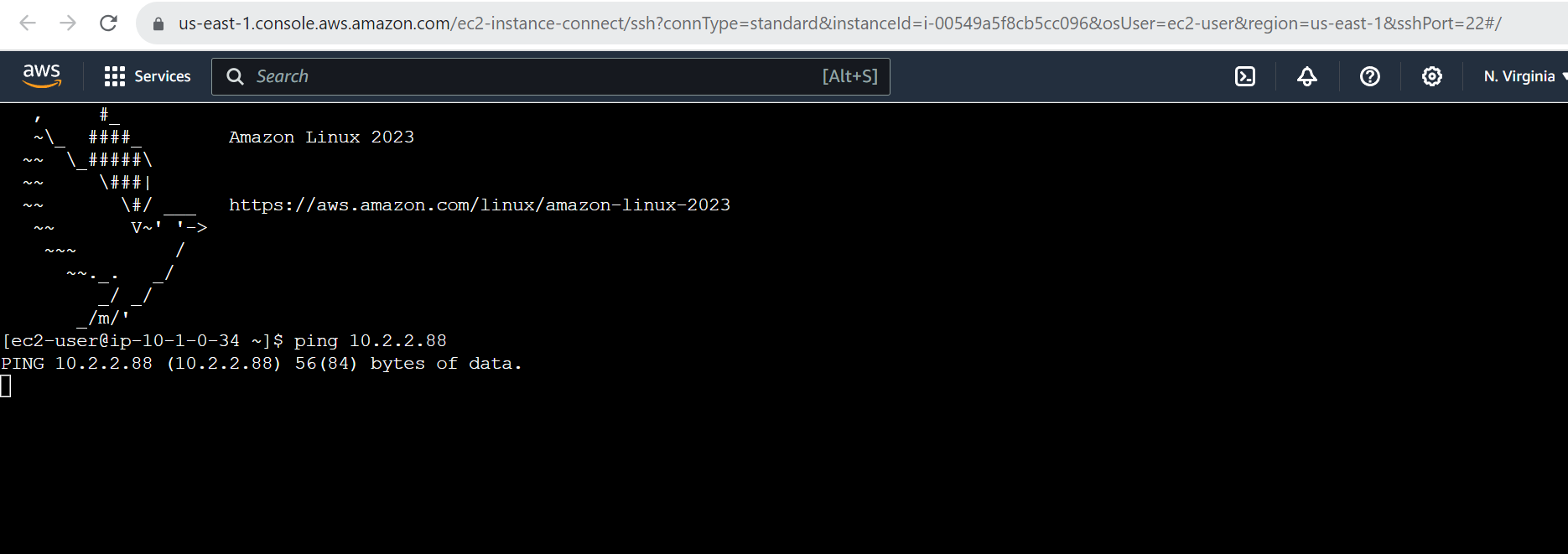
1. For N. Virginia 10.1.0.0/22 – select the IP in security group as 10.2.0.0/22
2. For Ohio 10.2.0.0/22 – select IP in security group as 10.1.0.0/22

EC2 instance, Public Subnet in N. Virginia



EC2 instance in, Private subnet IP - 10.2.2.88 in Ohio





Through Peering Connection I was able to access private subnet of Ohio region from Public subnet of N.Virginia region.

9. **CloudWatch Monitoring**: This project involves configuring CloudWatch alarms and monitoring for the resources in an AWS environment.

CPU utilization

