**School of Computer Science**

**UNIVERSITY OF PETROLEUM AND ENERGY STUDIES**

**DEHRADUN, UTTARAKHAND**



**System Provisioning and**

**Configuration Management**

|  |  |
| --- | --- |
| Submitted To:  Dr. Hitesh Kumar Sharma | Submitted By:  Chinthakindi Sree Vardhan Reddy  Batch – 2(DevOps)  500093990 |

Lab Exercise 3–Provisioning an EC2 Instance on AWS

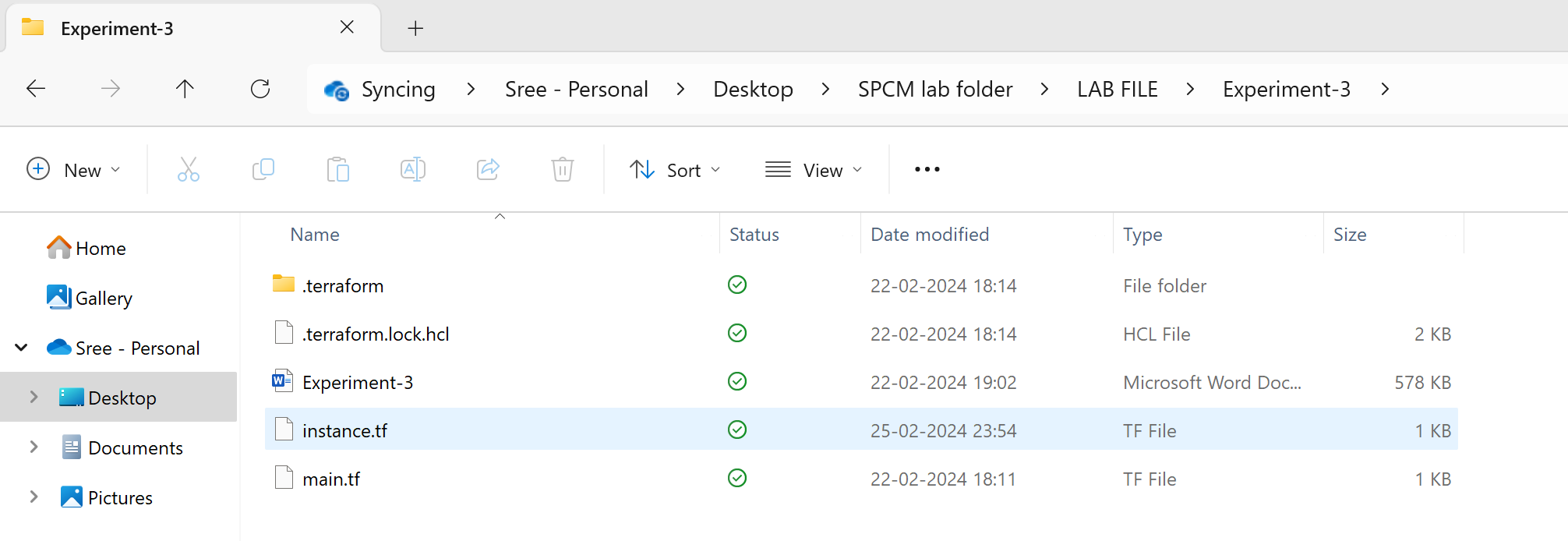
# Prerequisites: Terraform Installed: Make sure you have Terraform installed on your machine. Follow the official installation guide if needed.

AWS Credentials: Ensure you have AWS credentials (Access Key ID and Secret Access Key) configured. You can set them up using the AWS CLI or by setting environment variables.

# Exercise Steps:

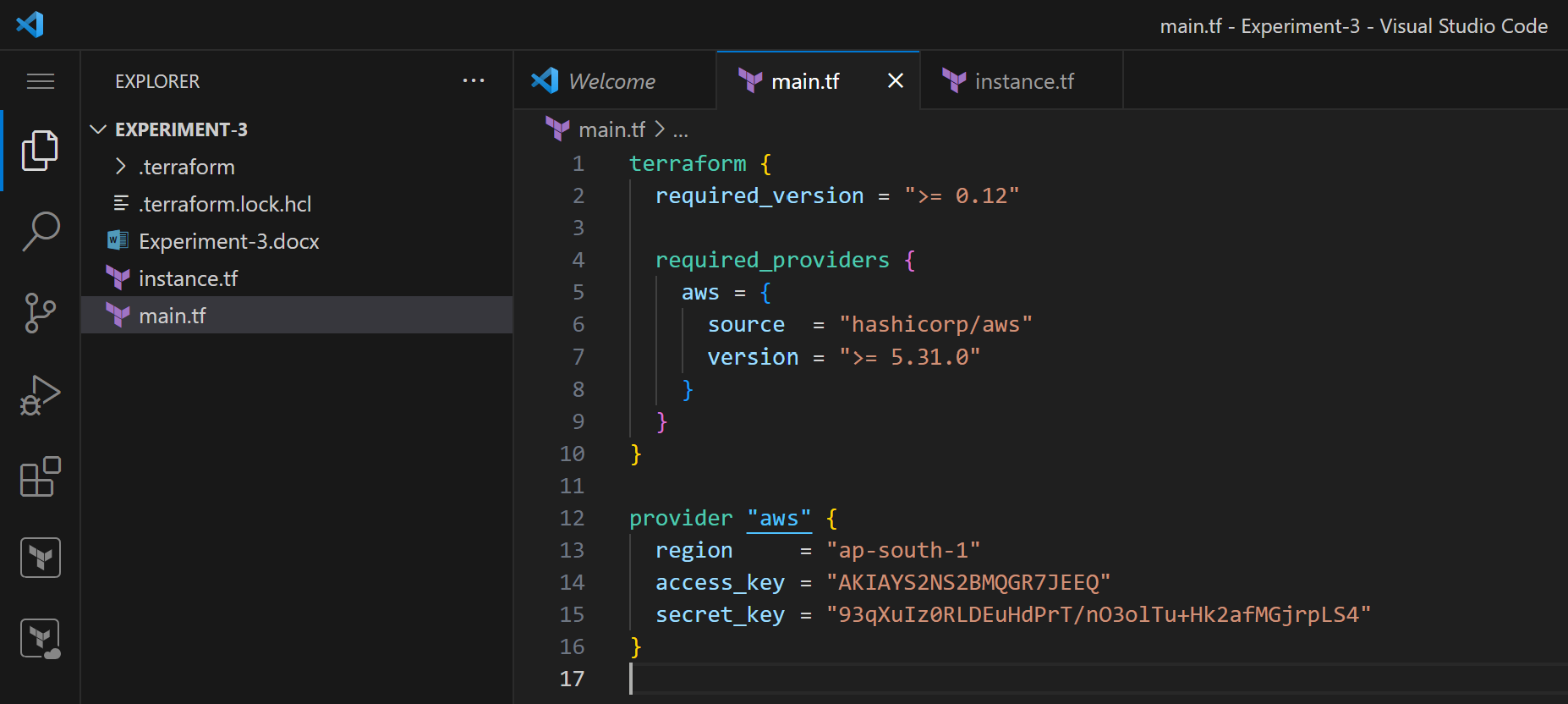
**Step 1: Create a New Directory:**

Create a new directory for your Terraform configuration:



# Step 2: Create Terraform Configuration File (main.tf):

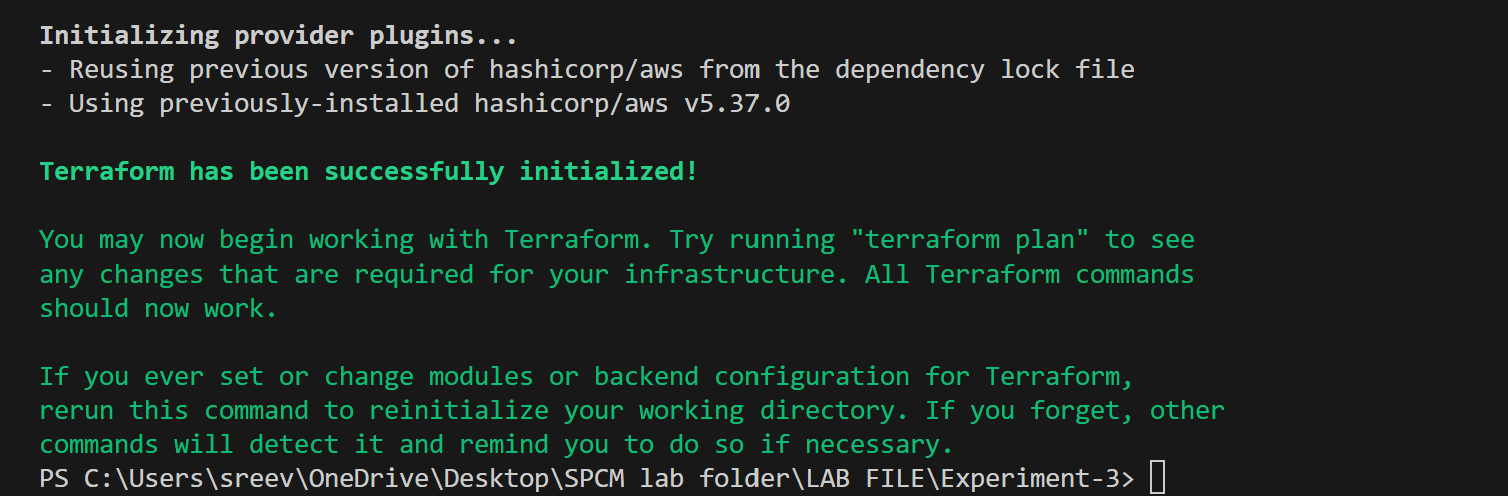
Create a file named main.tf with the following content:



This script defines an AWS provider and provisions an EC2 instance.

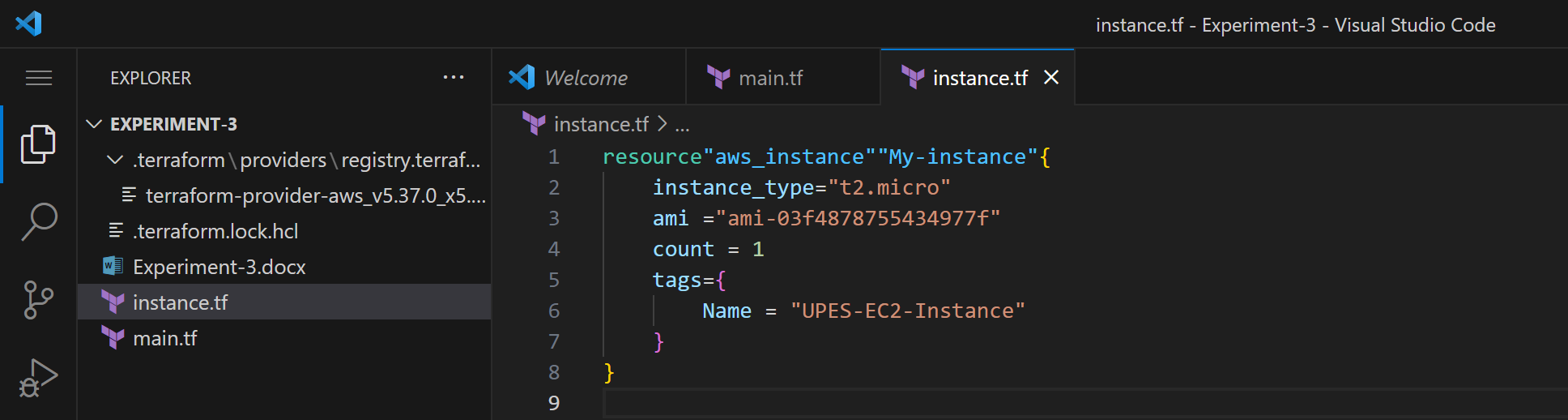
# Step 3: Initialize Terraform:

Run the following command to initialize your Terraform working directory:



# Step 4: Create Terraform Configuration File for EC2 instance (instance.tf):

Create a file named instnace.tf with the following content:

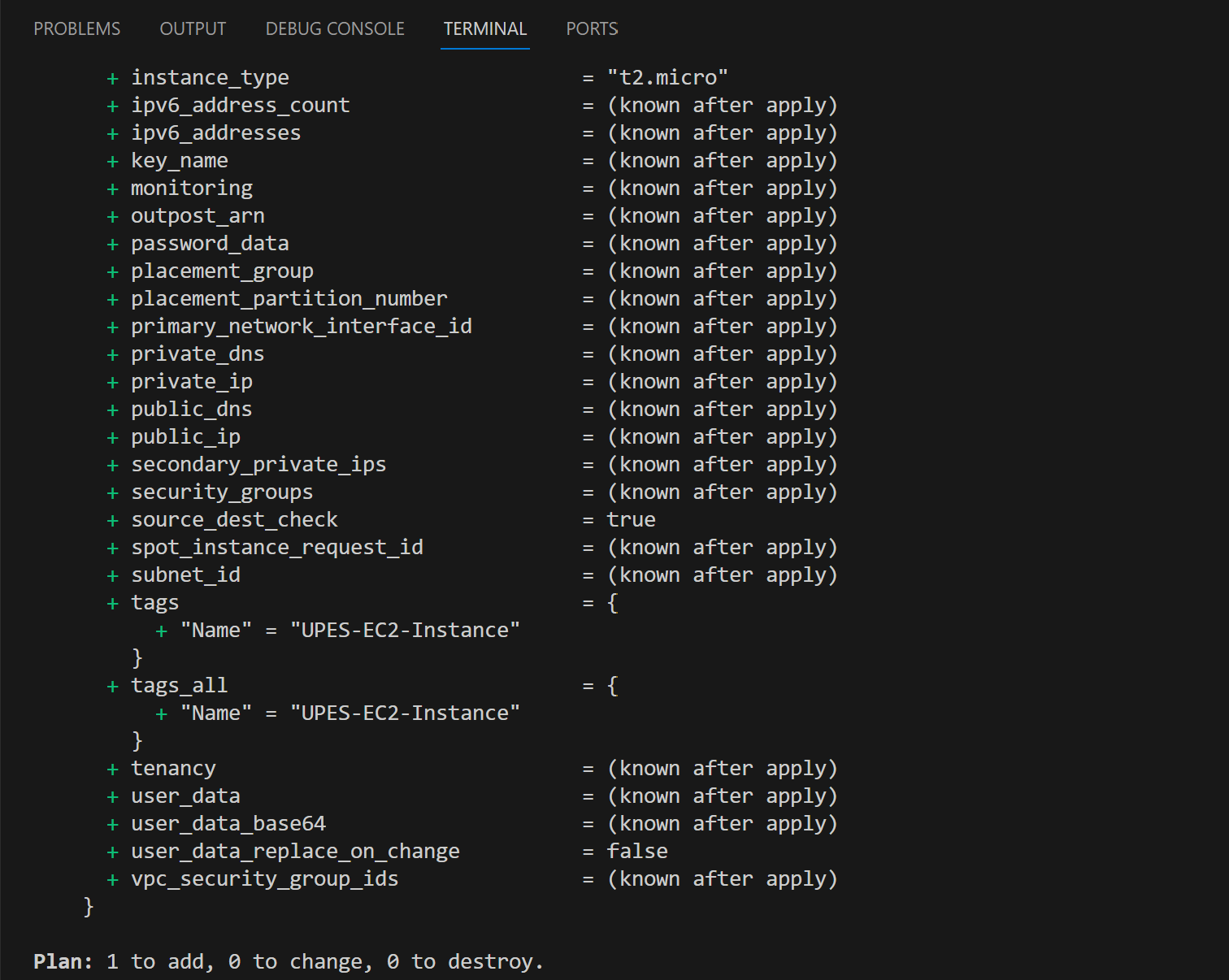


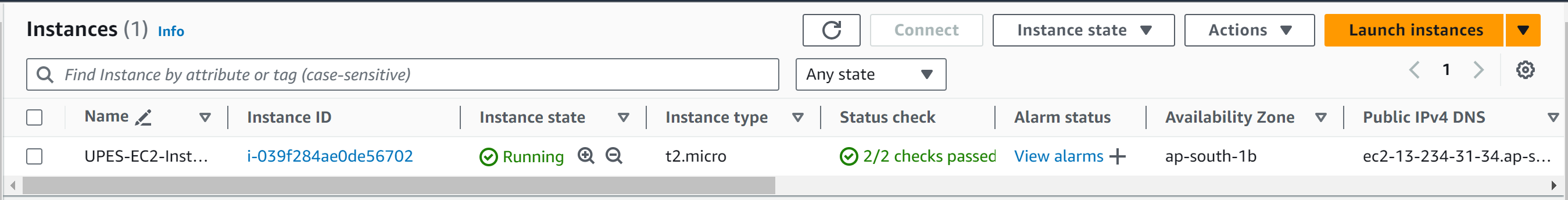
# Step 5: Review Plan:

Run the following command to see what Terraform will do:

Review the plan to ensure it aligns with your expectations.

Apply the changes to create the AWS resources:





# Step 7: Verify Resources:

After the terraform apply command completes, log in to your AWS Management Console and navigate to the EC2 dashboard. Verify that the EC2 instance has been created.

# Step 8: Cleanup Resources:

When you are done experimenting, run the following command to destroy the created resources:

