

Experiment No. 3

Title:

Working in Cloud9 to demonstrate different language.

Objective:

To learn Platform as service (PaaS) using Free trial of Cloud9 in AWS, because firstly it was an independent platform now it is integrated with AWS.

Tools used:

Internet, AWS, EC2, Cloud9

Prerequisite:

Understanding of Platform as service in Service model of cloud computing

Theory:

AWS Cloud9 is an integrated development environment, or *IDE*.

The AWS Cloud9 IDE offers rich code-editing experience with support for several programming languages and runtime debuggers, and a built-in terminal. It contains a collection of tools that you use to code, build, run, test, and debug software, and helps you release software to the cloud.

You access the AWS Cloud9 IDE through a web browser. You can configure the IDE to your preferences. You can switch color themes, bind shortcut keys, enable programming language-specific syntax coloring and code formatting, and more.

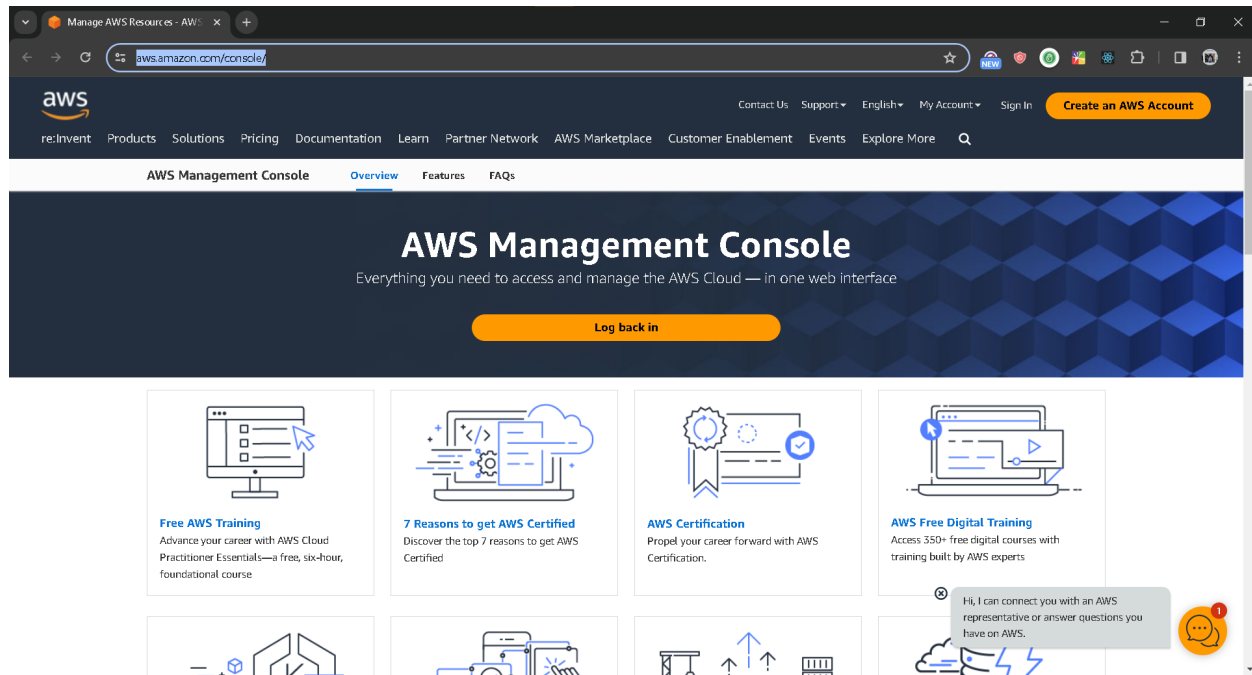
An *AWS Cloud9 environment* is a place where you store your project's files and where you run the tools to develop your applications.

Using the AWS Cloud9 IDE, you can:

- Store your project's files locally on the instance or server.
- Clone a remote code repository—such as a repo in AWS Code Commit—into your environment.
- Work with a combination of local and cloned files in the environment.

Steps to set up cloud9.

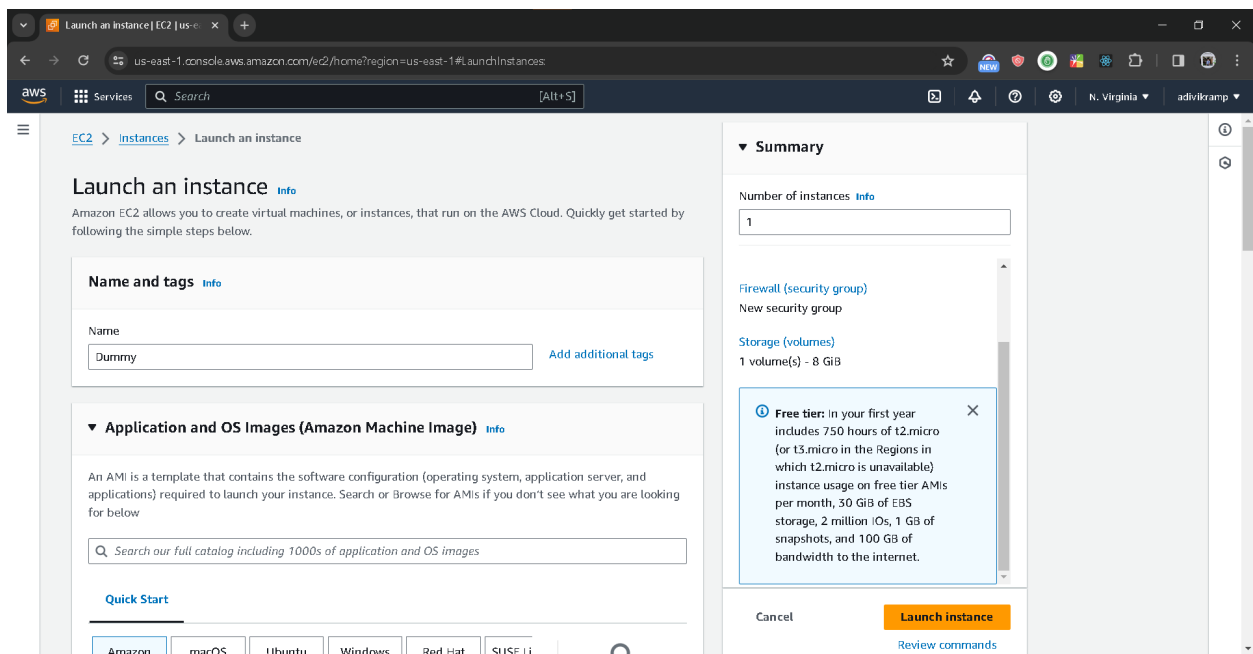
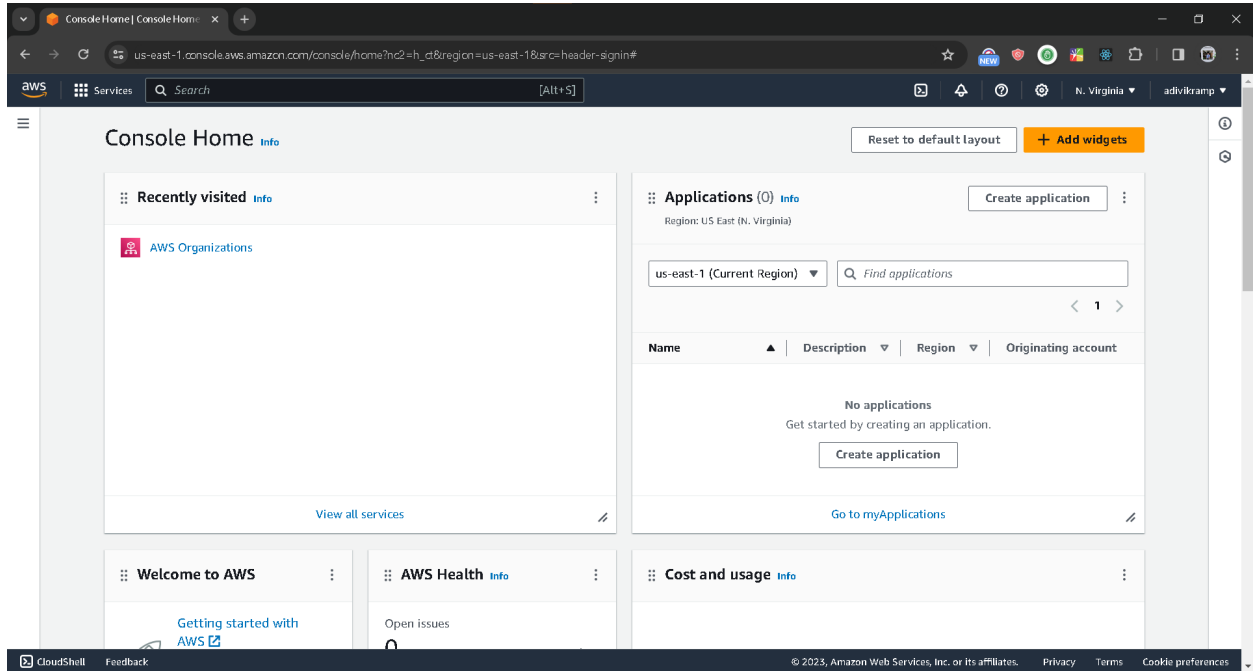
Step 1 - Sign into the Console <https://aws.amazon.com/console/>



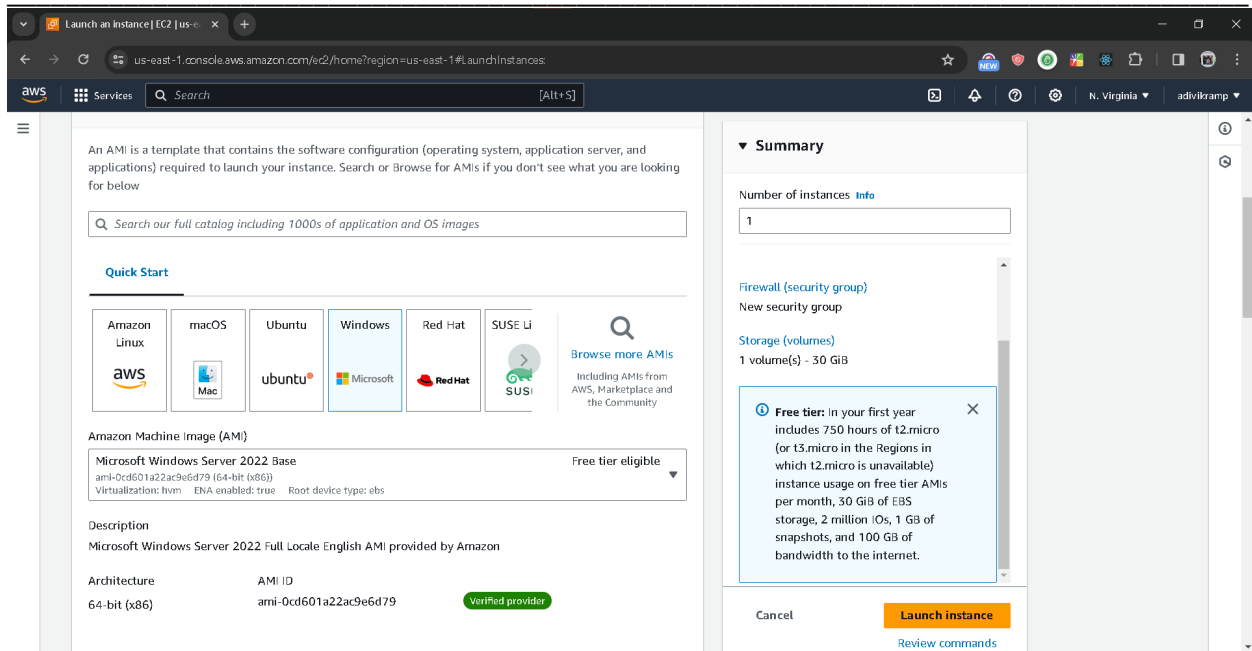
Step 2. Create EC2 instance and launch it.

- ✓ **Log in to your AWS Management Console.**
- ✓ **Go to the EC2 dashboard and click on the "Launch Instance" button.**
- ✓ **Choose an Amazon Machine Image (AMI) for your instance.**
- ✓ **Select an instance type and configure instance details like network settings, storage, and tags.**
- ✓ **Create a new key pair.**
- ✓ **Configure security groups to control inbound and outbound traffic.**
- ✓ **Review your instance settings and launch the EC2 instance.**
- ✓ **Create or use an existing key pair for secure access to your instance.**

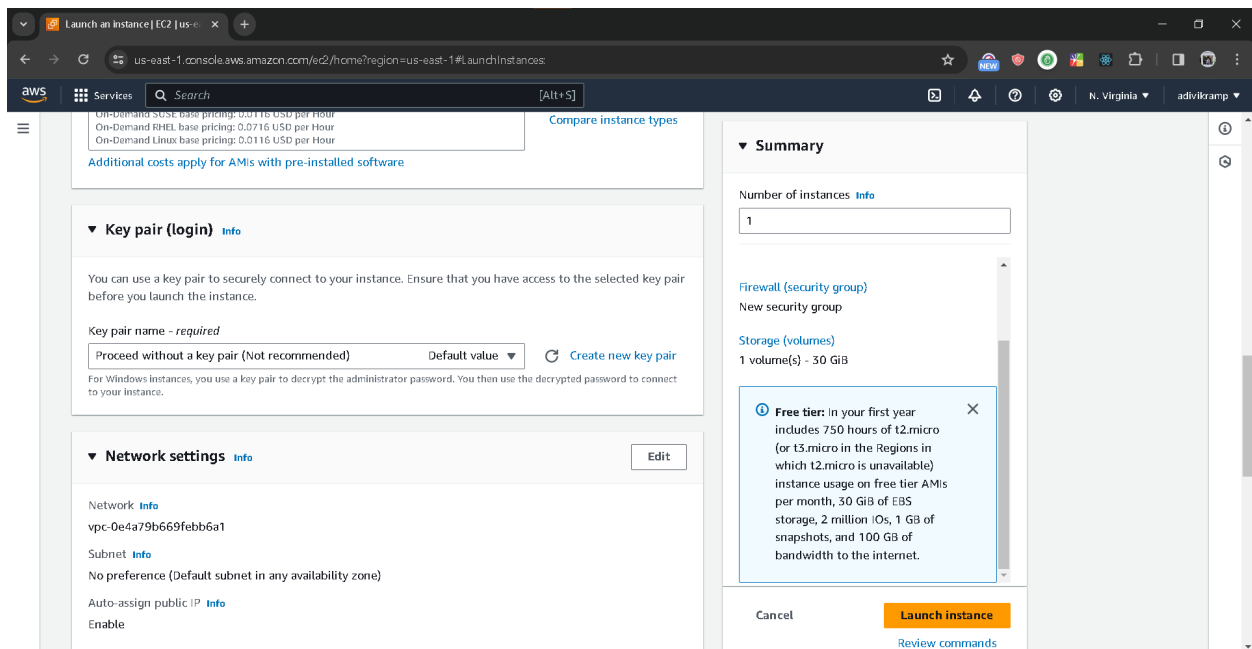
✓ **Access your EC2 instance using SSH (Secure Shell) or other remote access methods.**



NUTAN COLLEGE OF ENGINEERING & RESEARCH (NCER)
Department of Computer Science & Engineering (CSE)

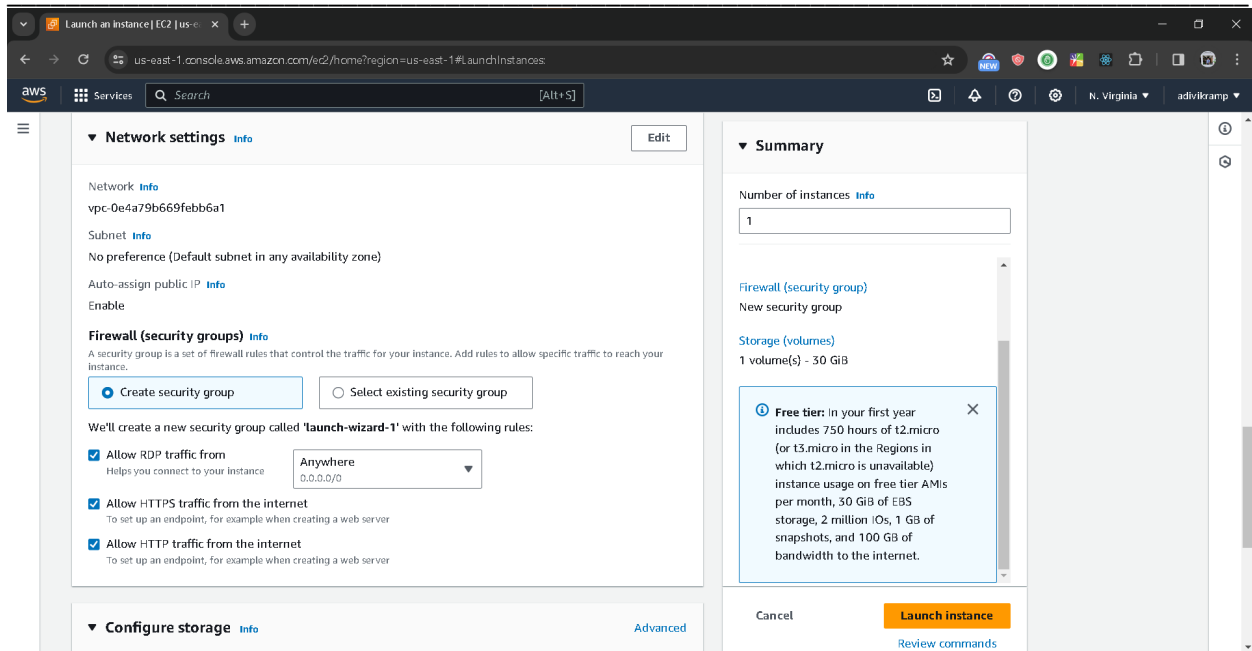


The screenshot shows the 'Launch an instance' page in the AWS Management Console. The left sidebar contains a 'Quick Start' section with buttons for Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux. Below this, the 'Amazon Machine Image (AMI)' section displays 'Microsoft Windows Server 2022 Base' with details like AMI ID, virtualization type, and root device type. A 'Description' section notes it's a full locale English AMI. The 'Architecture' is 64-bit (x86). The right sidebar shows a 'Summary' with 'Number of instances' set to 1, 'Firewall (security group)' set to 'New security group', and 'Storage (volumes)' set to '1 volume(s) - 30 GiB'. A 'Free tier' notification is visible, stating that in the first year, it includes 750 hours of t2.micro or t3.micro instance usage. At the bottom, there are 'Cancel' and 'Launch instance' buttons, along with a 'Review commands' link.



This screenshot shows the 'Launch an instance' page with the 'Key pair (login)' section expanded. It explains that a key pair is used to securely connect to the instance and provides a field for the 'Key pair name' with a 'Default value' dropdown and a 'Create new key pair' button. Below this, the 'Network settings' section is expanded, showing the 'Network' as 'vpc-0e4a79b669febb6a1', the 'Subnet' as 'No preference (Default subnet in any availability zone)', and 'Auto-assign public IP' as 'Enable'. The right sidebar remains the same as in the previous screenshot, showing the 'Summary' with instance count, security group, and storage settings. The 'Free tier' notification is also present here.

NUTAN COLLEGE OF ENGINEERING & RESEARCH (NCER)
Department of Computer Science & Engineering (CSE)



The screenshot shows the 'Launch an instance' wizard in the AWS Management Console, specifically the 'Network settings' page. The page is divided into two main sections: 'Network settings' on the left and 'Summary' on the right.

Network settings:

- Network:** vpc-0e4a79b669febb6a1
- Subnet:** No preference (Default subnet in any availability zone)
- Auto-assign public IP:** Enable
- Firewall (security groups):**
 - ☒ Create security group
 - ☐ Select existing security group

We'll create a new security group called 'launch-wizard-1' with the following rules:

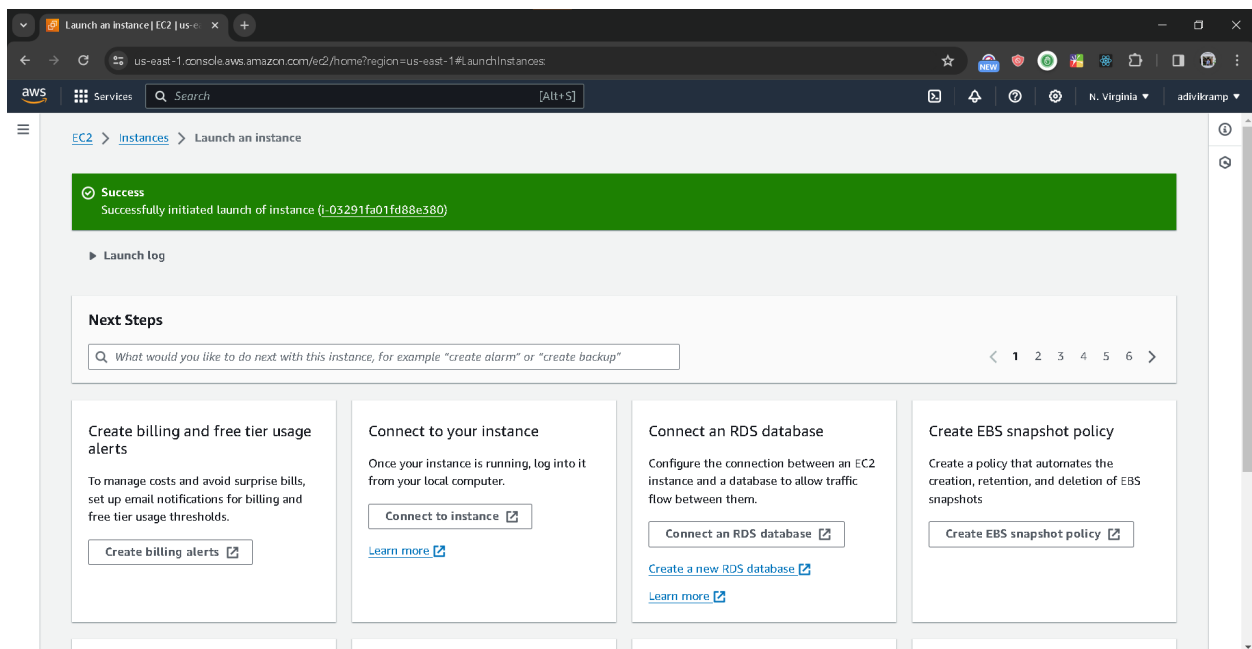
- ☒ Allow RDP traffic from **Anywhere** (0.0.0.0/0)
- ☒ Allow HTTPS traffic from the internet
- ☒ Allow HTTP traffic from the internet

Summary:

- Number of instances:** 1
- Firewall (security group):** New security group
- Storage (volumes):** 1 volume(s) - 30 GiB

A blue information box on the right states: "Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet."

At the bottom right, there are buttons for 'Cancel', 'Launch instance', and 'Review commands'.



The screenshot shows the 'Launch an instance' wizard in the AWS Management Console, specifically the 'Success' page. The page is divided into two main sections: 'Success' at the top and 'Next Steps' below.

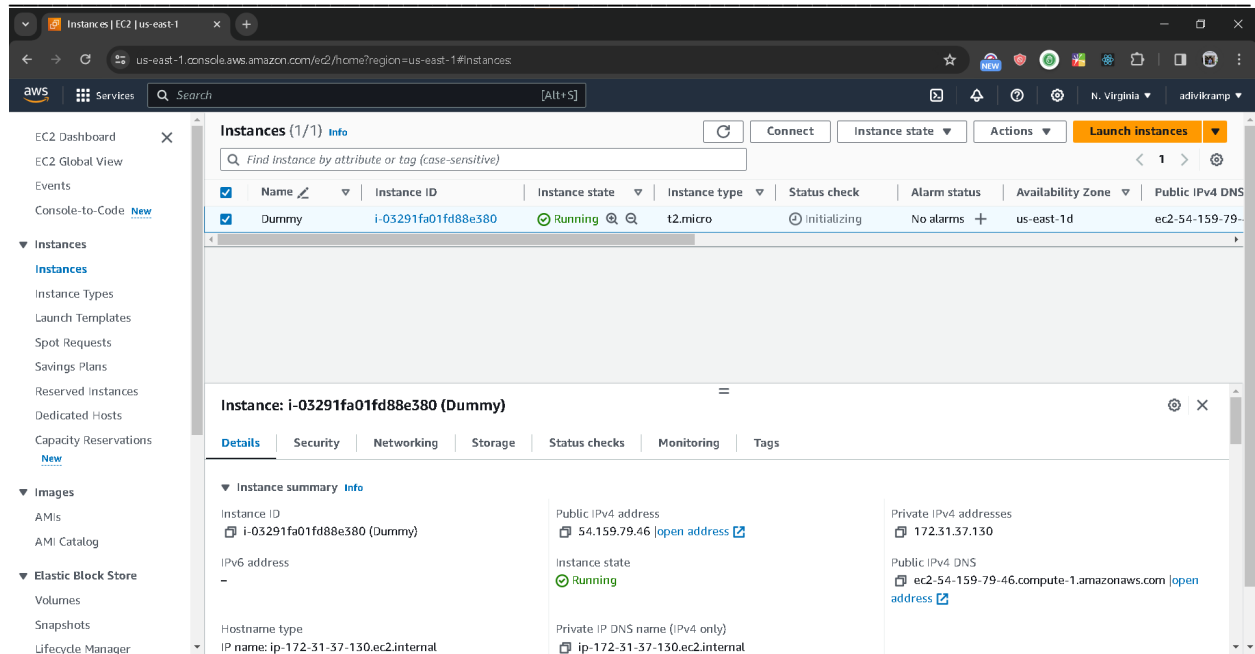
Success:

- Successfully initiated launch of instance (i-03291fa01fd88e380)**

Next Steps:

- Create billing and free tier usage alerts:** To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds. [Create billing alerts](#)
- Connect to your instance:** Once your instance is running, log into it from your local computer. [Connect to instance](#) [Learn more](#)
- Connect an RDS database:** Configure the connection between an EC2 instance and a database to allow traffic flow between them. [Connect an RDS database](#) [Create a new RDS database](#) [Learn more](#)
- Create EBS snapshot policy:** Create a policy that automates the creation, retention, and deletion of EBS snapshots. [Create EBS snapshot policy](#)

NUTAN COLLEGE OF ENGINEERING & RESEARCH (NCER)
Department of Computer Science & Engineering (CSE)



The screenshot displays the AWS Management Console for the 'us-east-1' region. The left sidebar shows navigation options like EC2 Dashboard, EC2 Global View, Events, and various instance management tools. The main content area shows a list of EC2 instances with columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. A single instance named 'Dummy' with ID 'i-03291fa01fd88e380' is shown in a 'Running' state. Below the list, the 'Instance summary' for 'Dummy' is expanded, showing details like Instance ID, Public IPv4 address (54.159.79.46), Private IPv4 addresses (172.31.37.130), Instance state (Running), IPv6 address, Hostname type, and IP name (ip-172-31-37-130.ec2.internal).

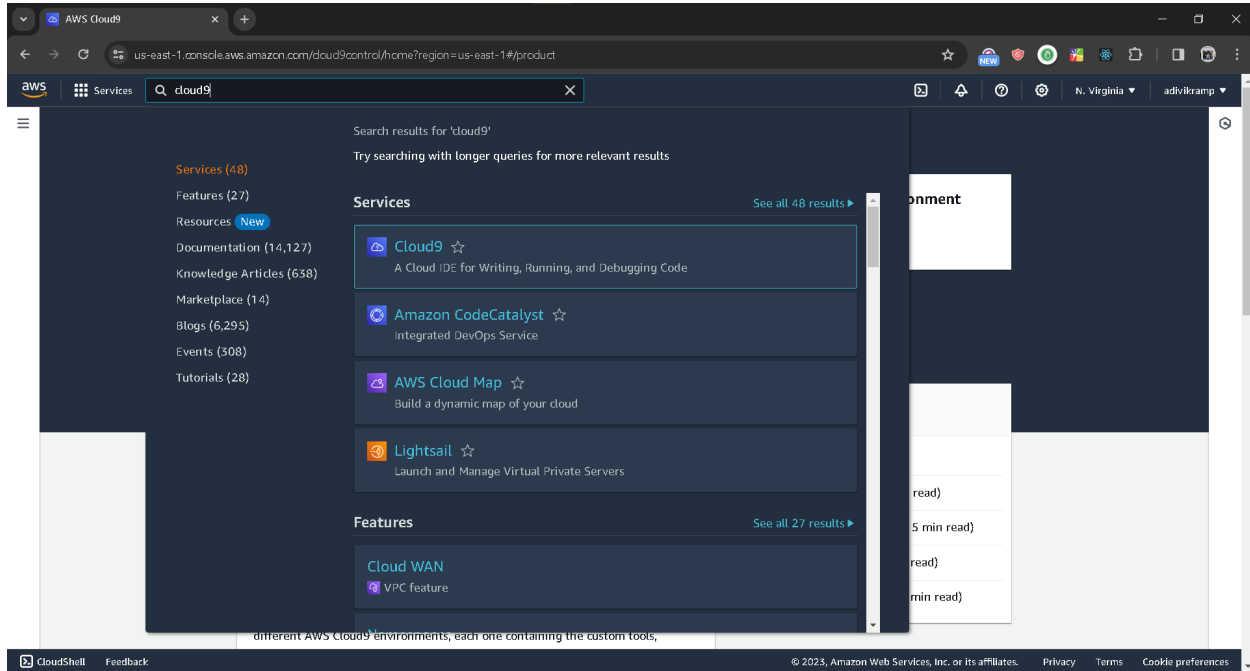
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
Dummy	i-03291fa01fd88e380	Running	t2.micro	Initializing	No alarms	us-east-1d	ec2-54-159-79-...

Instance: i-03291fa01fd88e380 (Dummy)

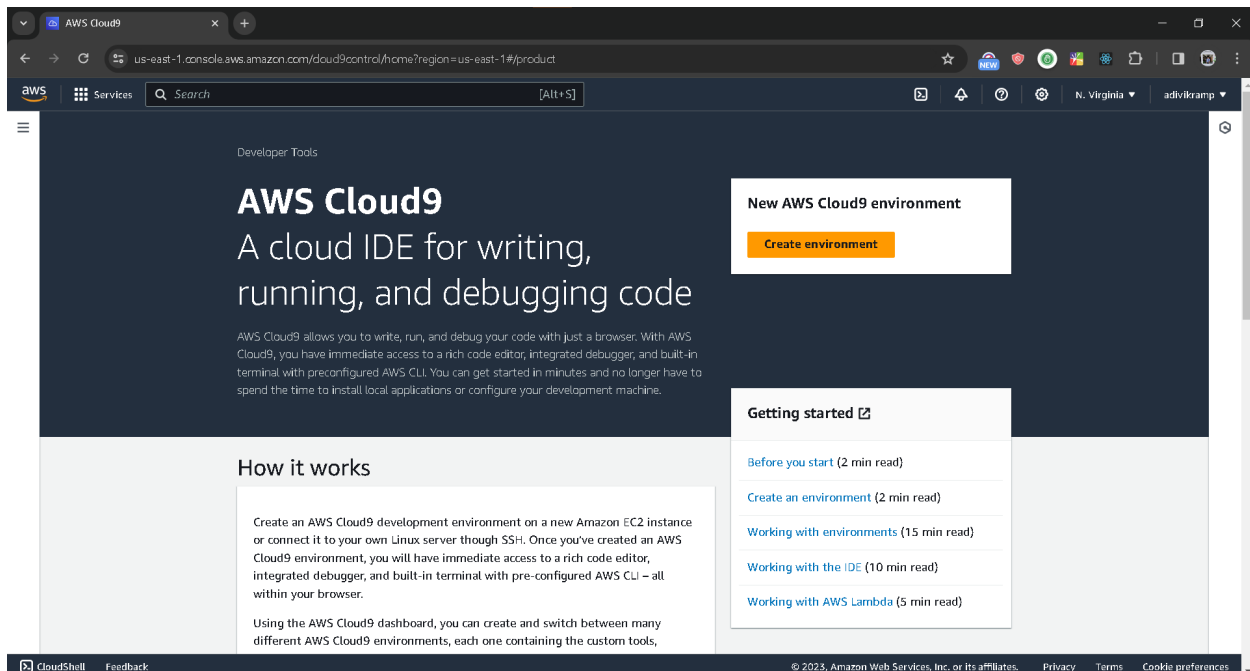
Instance summary

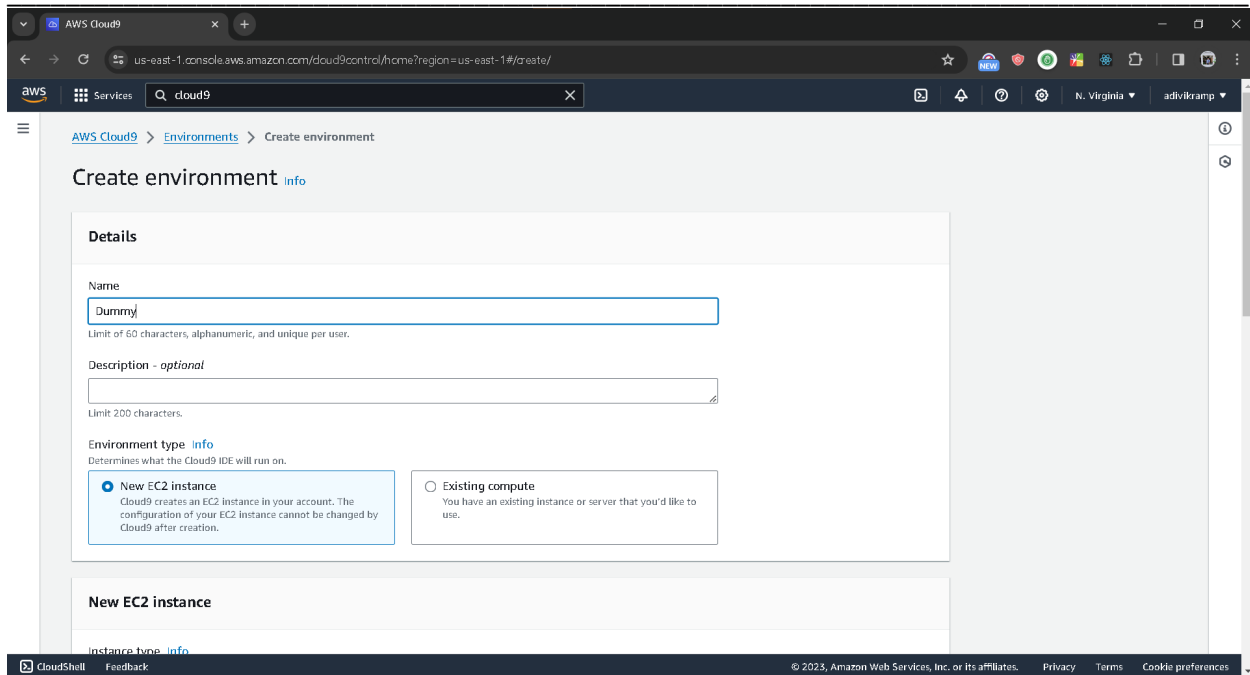
- Instance ID: i-03291fa01fd88e380 (Dummy)
- Public IPv4 address: 54.159.79.46 [open address](#)
- Private IPv4 addresses: 172.31.37.130
- Instance state: Running
- IPv6 address: -
- Public IPv4 DNS: ec2-54-159-79-46.compute-1.amazonaws.com [open address](#)
- Hostname type: -
- Private IP DNS name (IPv4 only): ip-172-31-37-130.ec2.internal
- IP name: ip-172-31-37-130.ec2.internal

Step 3 – Now we create instance and launch it, now it's time for Cloud9's setup



Create New Environment

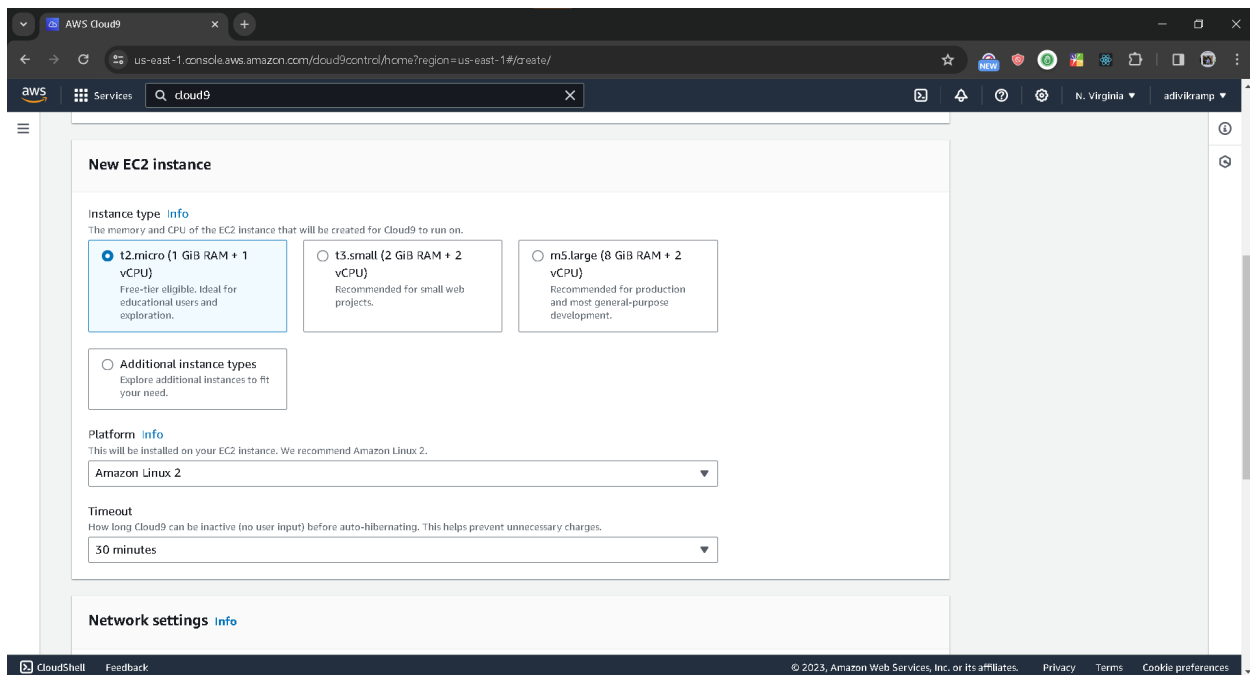




The screenshot shows the 'Create environment' page in the AWS Cloud9 console. The browser address bar shows the URL: `us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/create/`. The page title is 'Create environment info'. The 'Details' section contains the following fields:

- Name:** A text input field containing 'Dummy'. Below it, a note states: 'Limit of 60 characters, alphanumeric, and unique per user.'
- Description - optional:** A text input field. Below it, a note states: 'Limit 200 characters.'
- Environment type:** A section with two radio buttons:
 - ☒ **New EC2 instance**
Cloud9 creates an EC2 instance in your account. The configuration of your EC2 instance cannot be changed by Cloud9 after creation.
 - ☐ **Existing compute**
You have an existing instance or server that you'd like to use.

Below the 'Details' section is a section titled 'New EC2 instance' with a link for 'Instance type info'.

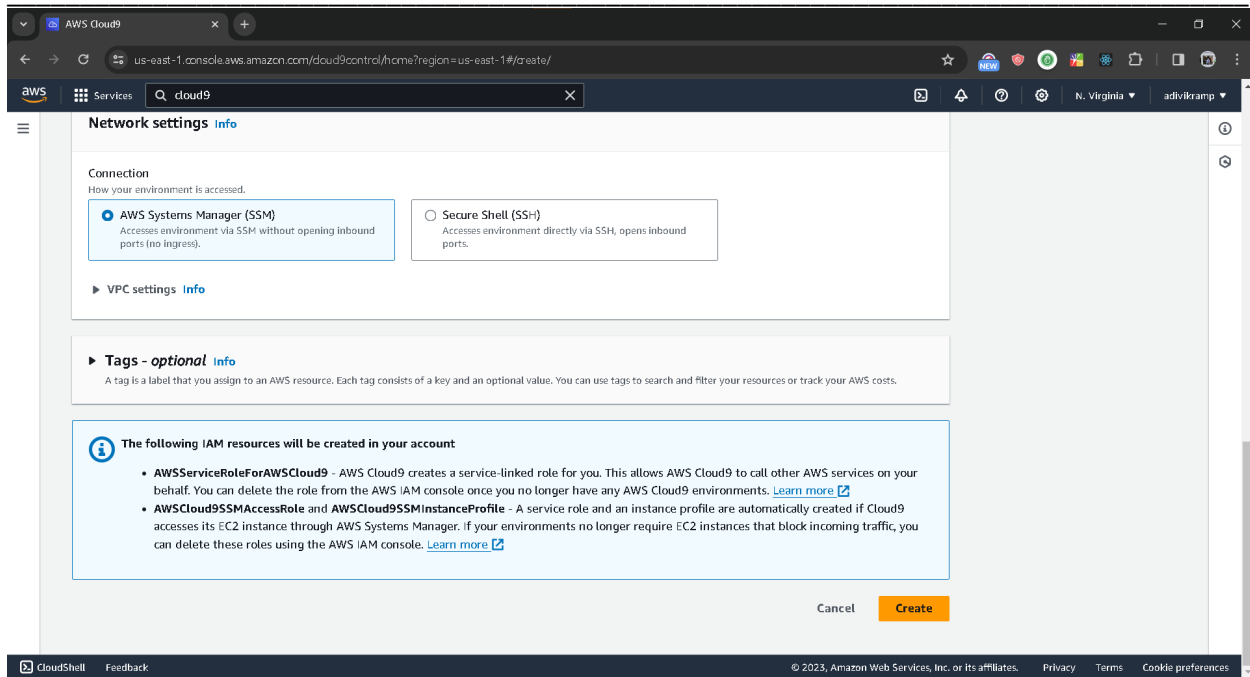


The screenshot shows the 'New EC2 instance' page in the AWS Cloud9 console. The browser address bar shows the URL: `us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/create/`. The page title is 'New EC2 instance'. The 'Instance type' section contains the following options:

- ☒ **t2.micro (1 GiB RAM + 1 vCPU)**
Free-tier eligible. Ideal for educational users and exploration.
- ☐ **t3.small (2 GiB RAM + 2 vCPU)**
Recommended for small web projects.
- ☐ **m5.large (8 GiB RAM + 2 vCPU)**
Recommended for production and most general-purpose development.
- ☐ **Additional instance types**
Explore additional instances to fit your need.

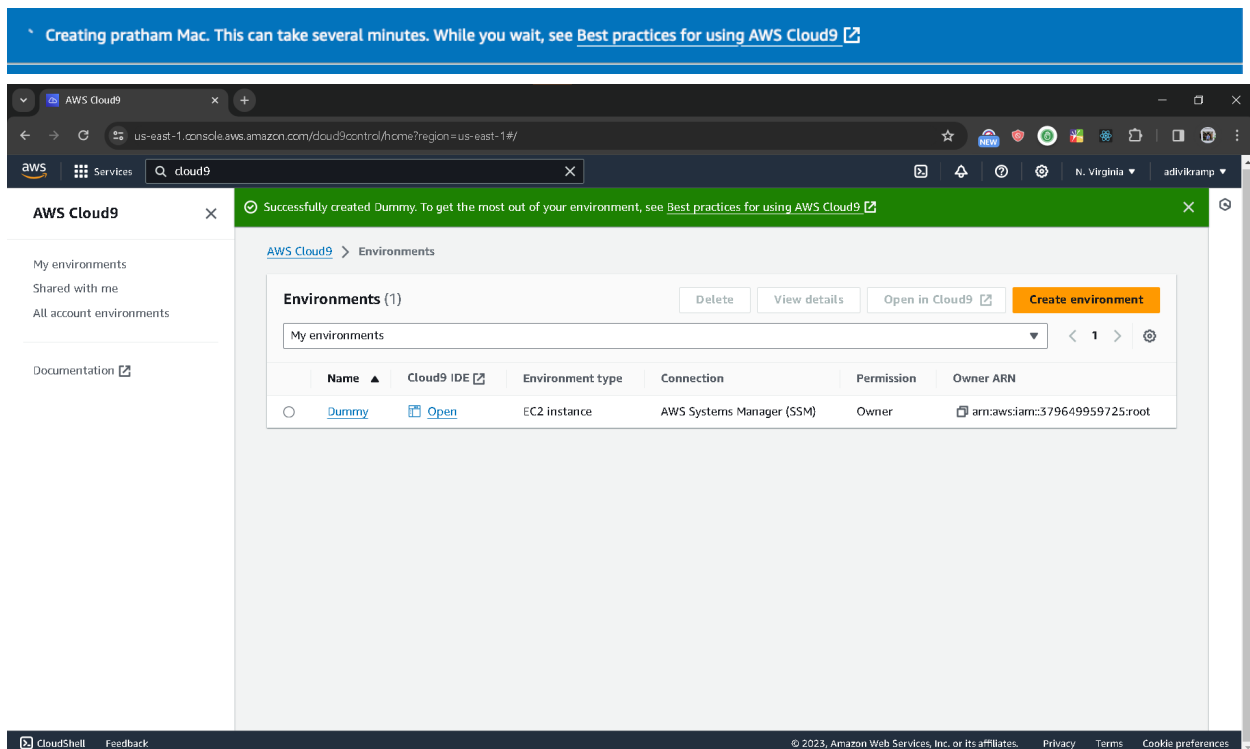
Below the 'Instance type' section is a section titled 'Platform' with a link for 'info'. It contains a dropdown menu set to 'Amazon Linux 2'. Below that is a section titled 'Timeout' with a dropdown menu set to '30 minutes'. Below the 'Platform' and 'Timeout' sections is a section titled 'Network settings' with a link for 'info'.

NUTAN COLLEGE OF ENGINEERING & RESEARCH (NCER)
Department of Computer Science & Engineering (CSE)



The screenshot shows the AWS Cloud9 console's 'Network settings' page. The 'Connection' section has two options: 'AWS Systems Manager (SSM)' (selected) and 'Secure Shell (SSH)'. Below this is a 'Tags - optional' section. A blue box contains information about IAM resources to be created: 'AWSServiceRoleForAWSCloud9', 'AWSCloud9SSMAccessRole', and 'AWSCloud9SSMInstanceProfile'. At the bottom right are 'Cancel' and 'Create' buttons.

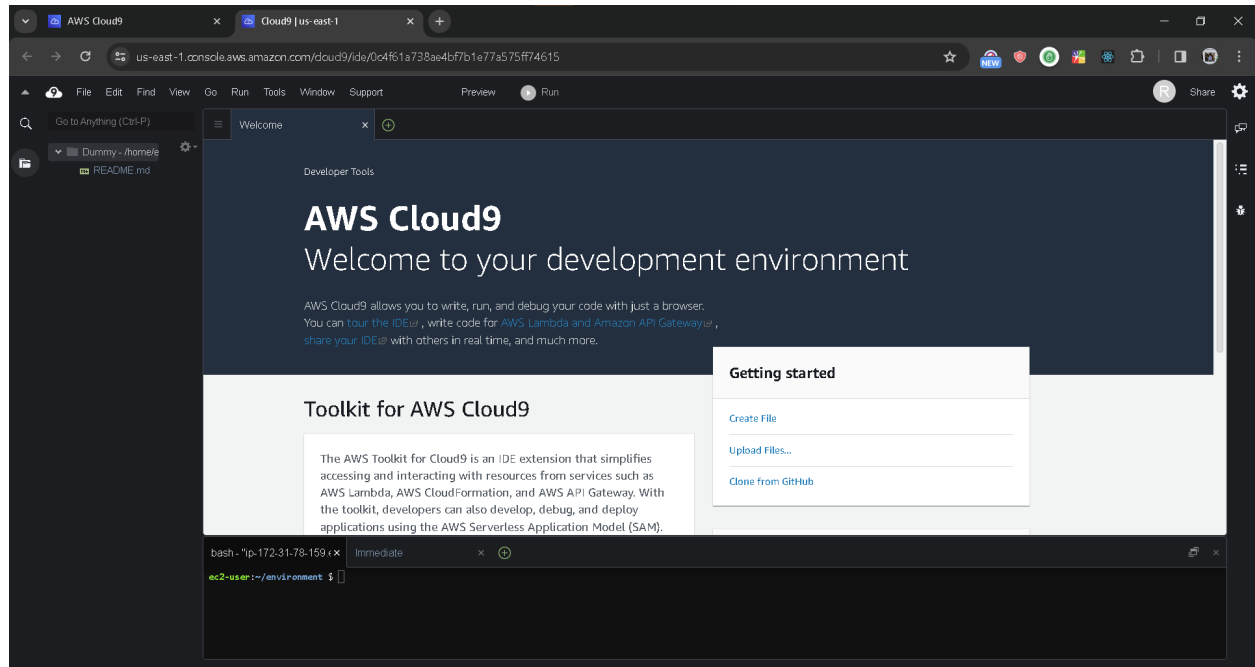
Step 4 – Successfully Account is created



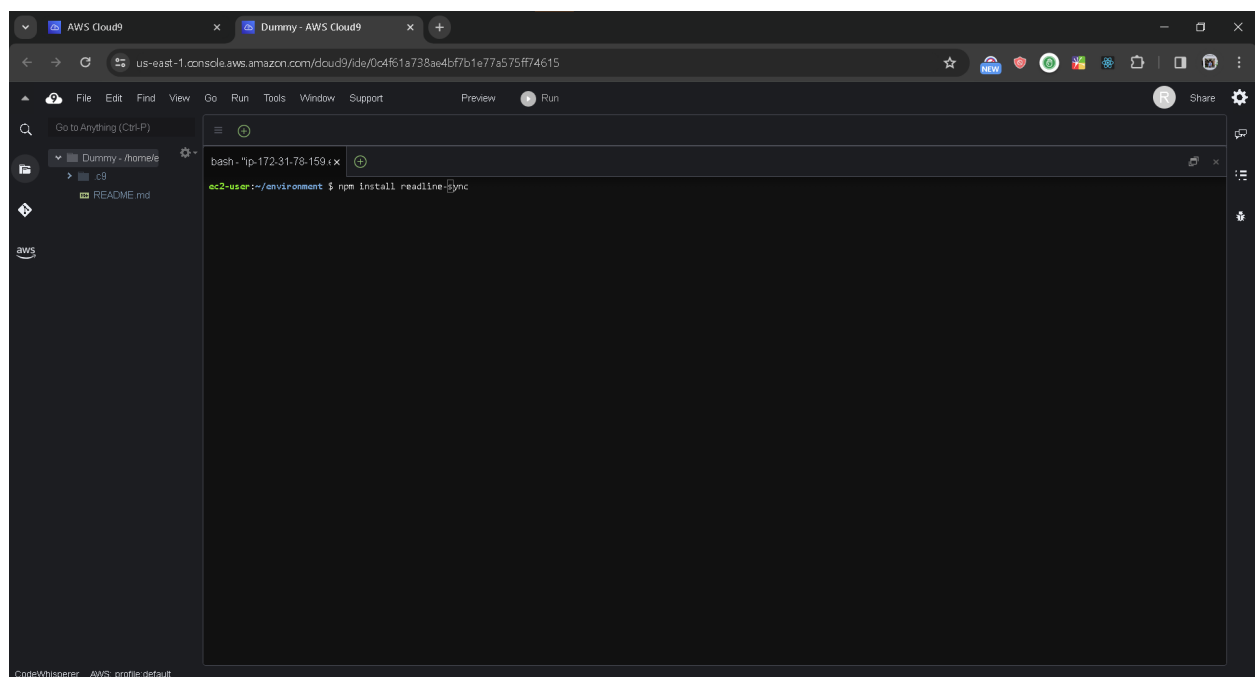
The screenshot shows the AWS Cloud9 console's 'Environments' page. A green banner at the top says 'Successfully created Dummy. To get the most out of your environment, see Best practices for using AWS Cloud9'. Below this is a table of environments. The table has columns: Name, Cloud9 IDE, Environment type, Connection, Permission, and Owner ARN. One environment named 'Dummy' is listed with type 'EC2 instance' and connection 'AWS Systems Manager (SSM)'. Buttons for 'Delete', 'View details', 'Open in Cloud9', and 'Create environment' are visible.

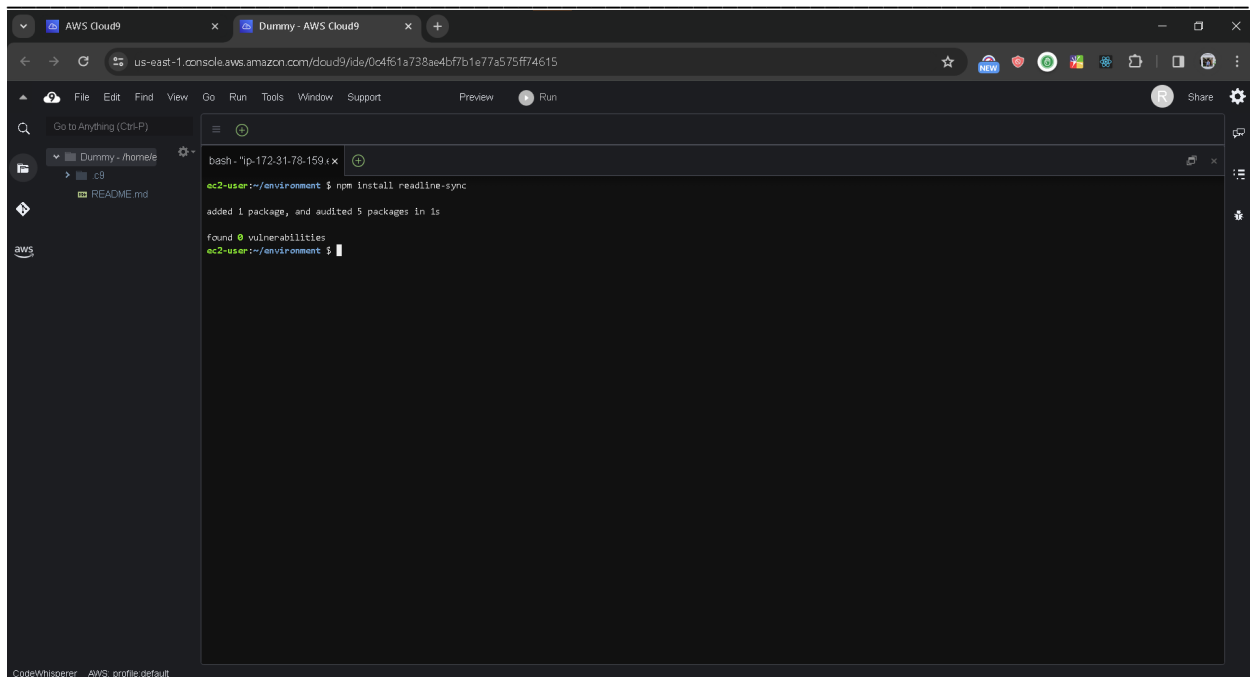
Name	Cloud9 IDE	Environment type	Connection	Permission	Owner ARN
Dummy	Open	EC2 instance	AWS Systems Manager (SSM)	Owner	arn:aws:iam::379649959725:root

Step 5 – After creating the cloud9 environment now we can start code in the environment.



`npm install readline-sync`





The screenshot shows the AWS Cloud9 IDE interface. The top bar indicates the current session is 'Dummy - AWS Cloud9'. The browser address bar shows the URL 'us-east-1.console.aws.amazon.com/cloud9/ide/0c4f61a738ae4b77a575ff74615'. The left sidebar shows a file explorer with a directory structure: 'Dummy - /home/c9' containing 'README.md'. The main editor area displays a terminal window with the following content:

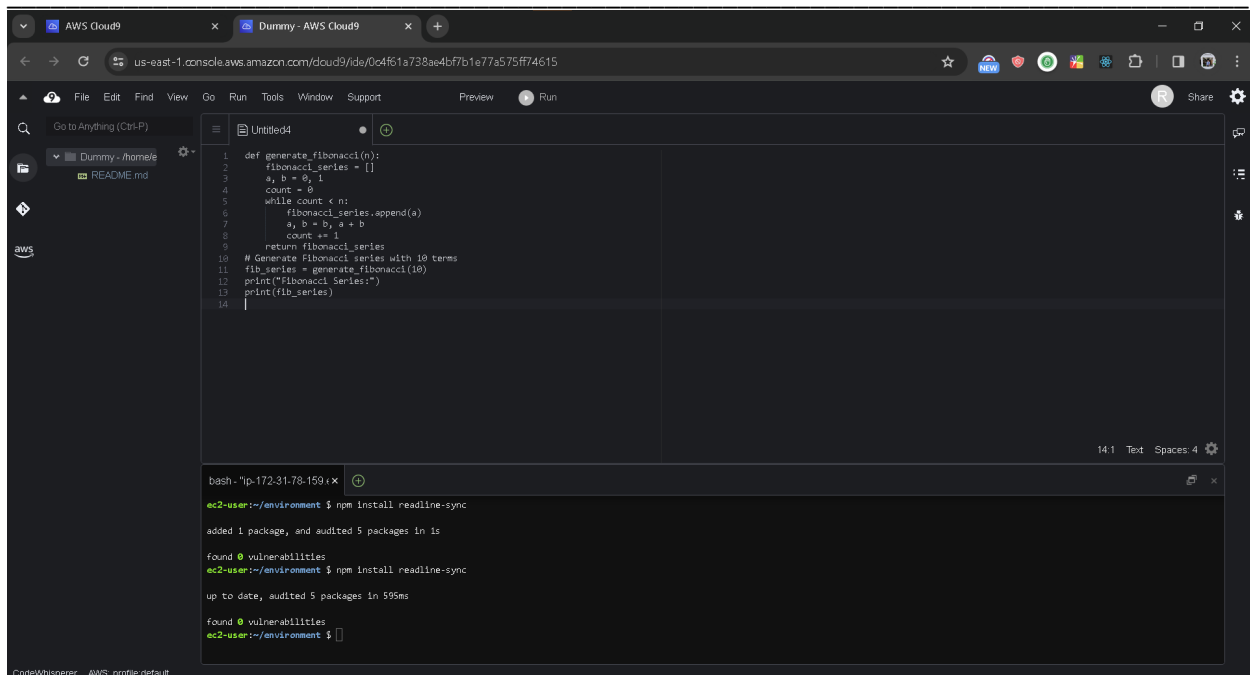
```
bash - "ip-172-31-78-159" x
ec2-user:~/environment $ npm install readline-sync
added 1 package, and audited 5 packages in 1s
found 0 vulnerabilities
ec2-user:~/environment $
```

Step 6. – Write the code in editor

python code for fibonacci series

```
def generate_fibonacci(n):  
    fibonacci_series = []  
    a, b = 0, 1  
    count = 0  
    while count < n:  
        fibonacci_series.append(a)  
        a, b = b, a + b  
        count += 1  
    return fibonacci_series  
  
# Generate Fibonacci series with 10 terms  
fib_series = generate_fibonacci(10)  
print("Fibonacci Series:")  
print(fib_series)
```

NUTAN COLLEGE OF ENGINEERING & RESEARCH (NCER)
Department of Computer Science & Engineering (CSE)



The screenshot shows the AWS Cloud9 IDE interface. The editor displays a Python script named 'Untitled4' with the following code:

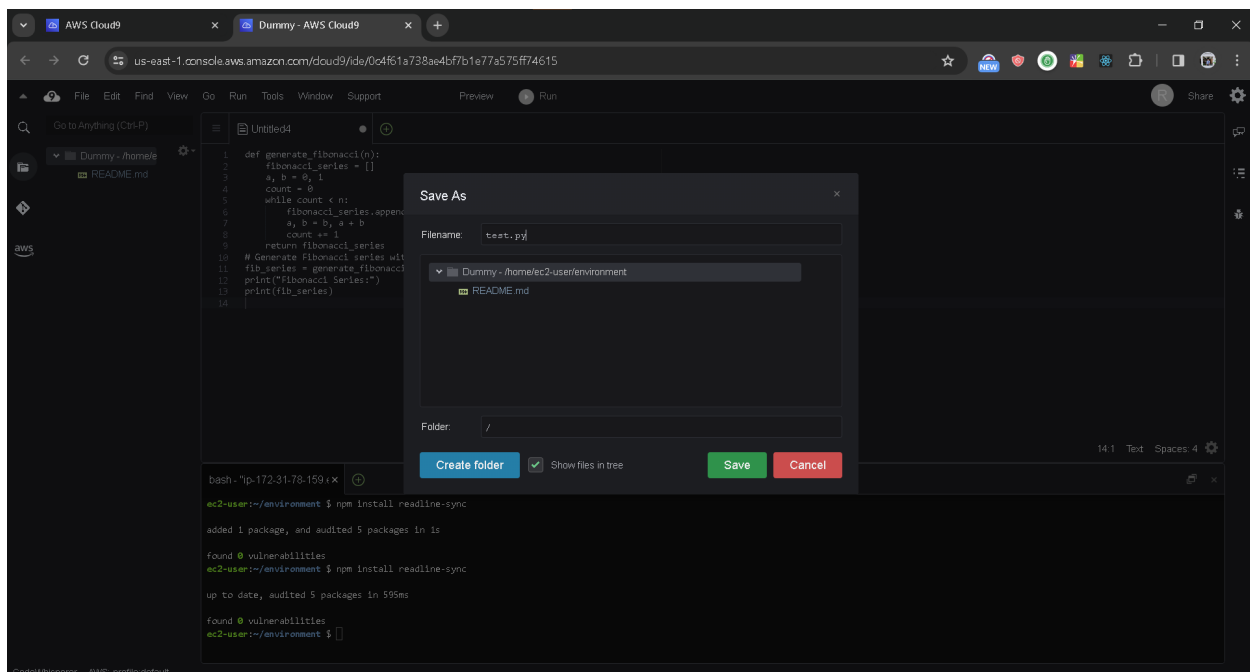
```
1 def generate_fibonacci(n):
2     fibonacci_series = []
3     a, b = 0, 1
4     count = 0
5     while count < n:
6         fibonacci_series.append(a)
7         a, b = b, a + b
8         count += 1
9     return fibonacci_series
10 # Generate fibonacci series with 10 terms
11 fib_series = generate_fibonacci(10)
12 print("Fibonacci Series:")
13 print(fib_series)
14
```

The terminal at the bottom shows the execution of the script:

```
bash-!ip-172-31-78-159~x
ec2-user:~/environment $ rpm install readline-sync
added 1 package, and audited 5 packages in 1s
found 0 vulnerabilities
ec2-user:~/environment $ rpm install readline-sync
up to date, audited 5 packages in 59ms
found 0 vulnerabilities
ec2-user:~/environment $
```

Step 7 – Save the file and run the code.

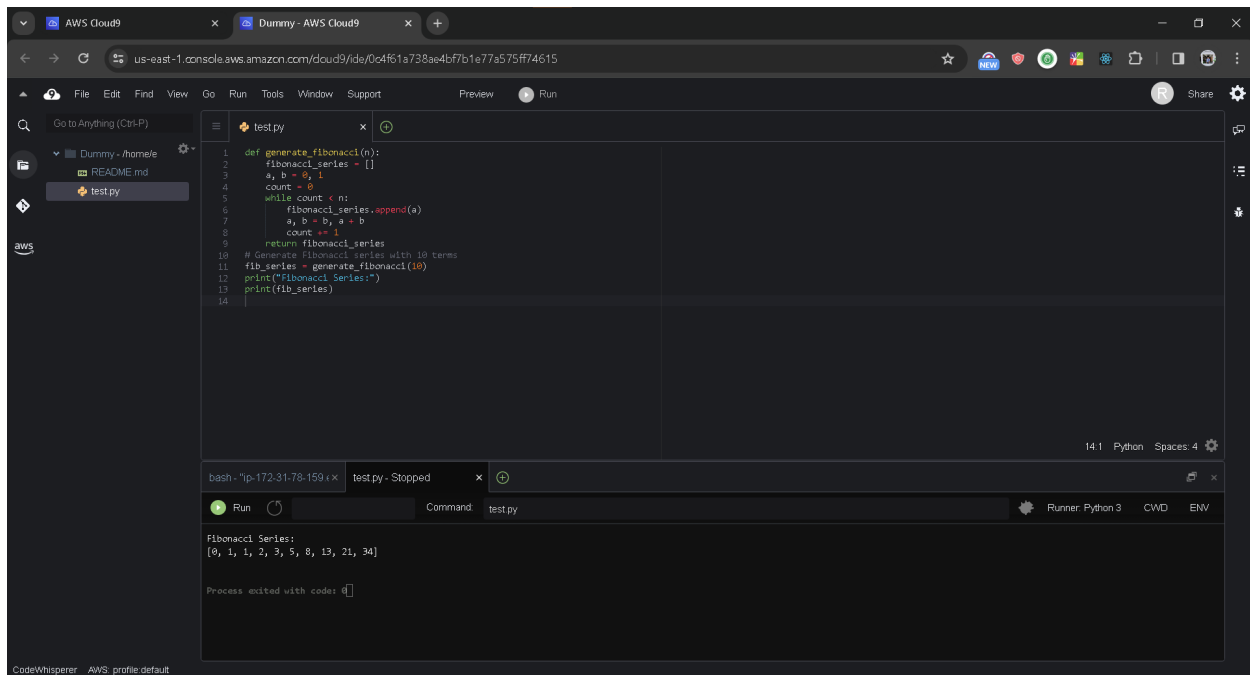
- 1. Click on Run**
- 2. Run Configuration**
- 3. New Configuration**



The screenshot shows the AWS Cloud9 IDE interface with a 'Save As' dialog box open. The dialog box has the following fields:

- Filename: test.py
- Folder: /
- Buttons: Create folder, Show files in tree, Save, Cancel

The background shows the same Python script and terminal output as the previous screenshot.



The screenshot displays the AWS Cloud9 IDE interface. The top toolbar includes options for File, Edit, Find, View, Go, Run, Tools, Window, and Support. The left sidebar shows the file explorer with a project named 'Dummy' containing a 'README.md' file and a 'test.py' file. The main editor area shows the code in 'test.py':

```
1 def generate_fibonacci(n):
2     fibonacci_series = []
3     a, b = 0, 1
4     count = 0
5     while count < n:
6         fibonacci_series.append(a)
7         a, b = b, a + b
8         count += 1
9     return fibonacci_series
10 # Generate Fibonacci series with 10 terms
11 fib_series = generate_fibonacci(10)
12 print("Fibonacci Series:")
13 print(fib_series)
14
```

Below the editor, the 'Run' tab is active, showing the command 'test.py' and the output:

```
bash - "ip-172-31-78-159" x test.py - Stopped
Run Command: test.py Runner: Python 3 CWD: ENV
Fibonacci Series:
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
Process exited with code: 0
```

In cloud9 you can write code in any languages like C++, C, python and many more.

Conclusion-

Successfully learnt Working in Cloud9 to demonstrate different language.