

# EXPERIMENT 1

## 1) To Develop A Website And Host it on a VM.

Steps:

Create An Instance:

The screenshot shows the AWS EC2 Dashboard. On the left, a sidebar lists various EC2 services: Global View, Events, Instances (with sub-options like Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations), Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots). The main area is titled 'Resources' and displays a summary of Amazon EC2 resources in the US East (N. Virginia) Region. It includes tables for Instances (running: 2), Auto Scaling Groups (0), Capacity Reservations (0), Dedicated Hosts (0), Elastic IPs (0), Instances (2), Key pairs (1), Load balancers (0), Placement groups (0), Security groups (1), Snapshots (0), and Volumes (2). To the right, there's an 'Account attributes' section showing the Default VPC (vpc-0004ba2bb735a928f) and a 'Settings' section with links for Data protection and security, Zones, EC2 Serial Console, Default credit specification, and EC2 console preferences. Below these are sections for 'Launch instance' (with 'Launch instance' and 'Migrate a server' buttons) and 'Service health' (showing AWS Health Dashboard, Region: US East (N. Virginia), Status: This service is operating normally, and Zones). A 'Explore AWS' sidebar on the right promotes better price performance and spot instances, and a footer at the bottom provides copyright information and navigation links.

The screenshot shows the 'Launch an instance' wizard. The top navigation bar includes tabs for New Tab, Launch AWS Academy Learner, Host your personal site on AWS, and Launch an instance | EC2 | us-east-1. The main content area has a breadcrumb trail: EC2 > Instances > Launch an instance. The first step, 'Launch an instance' (Info), describes creating virtual machines and provides simple steps. The second step, 'Name and tags' (Info), shows a 'Name' field with 'Web-Server' and a 'Add additional tags' button. The third step, 'Application and OS Images (Amazon Machine Image)' (Info), includes a search bar for AMIs and a 'Recent' tab. The fourth step, 'Summary' (Info), shows settings: Number of instances (1), Software Image (AMI) set to Amazon Linux 2023 AMI 2023.5.2... (read more), Virtual server type (Instance type) set to t2.micro, Firewall (security group) set to New security group, and Storage (volumes) set to 1 volume(s) - 8 GiB. A tooltip for the Free tier indicates it includes 750 hours of t2.micro (or t3.micro in regions where t2.micro is unavailable) instance. At the bottom are 'Cancel', 'Launch instance' (highlighted in orange), and 'Review commands' buttons. The footer at the bottom is identical to the one in the previous screenshot.

**Application and OS Images (Amazon Machine Image)**

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Recents Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux

Browse more AMIs Including AMIs from AWS, Marketplace and the Community

**Amazon Machine Image (AMI)**

Amazon Linux 2023 AMI  
ami-066784287e358dad1 (64-bit (x86), uefi-preferred) / ami-023508951a94f0c71 (64-bit (Arm), uefi)  
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.5.2...  
Virtual server type (instance type): t2.micro  
Firewall (security group): New security group  
Storage (volumes): 1 volume(s) - 8 GiB

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

Cancel Launch instance Review commands

**Instance type**

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true  
On-Demand Windows base pricing: 0.0162 USD per Hour  
On-Demand SUSE base pricing: 0.0116 USD per Hour  
On-Demand RHEL base pricing: 0.026 USD per Hour  
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations Compare instance types

Additional costs apply for AMIs with pre-installed software

**Key pair (login)**

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

vockey

Create new key pair

**Network settings**

Network

Edit

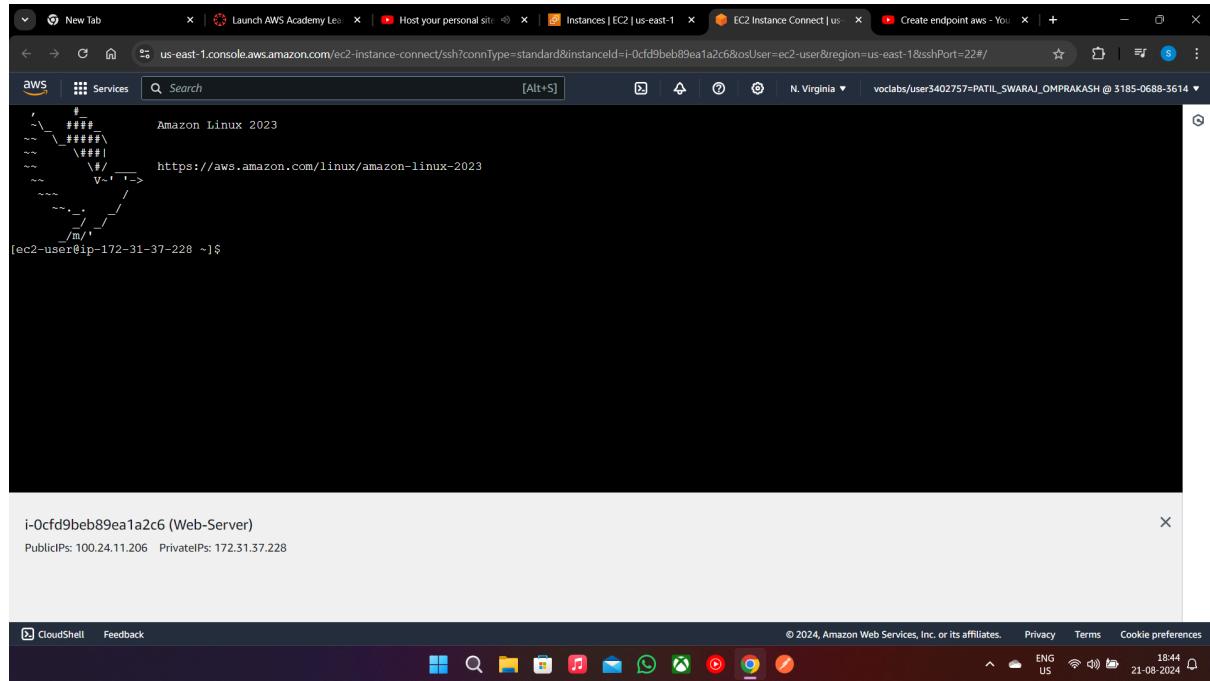
Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.5.2...  
Virtual server type (instance type): t2.micro  
Firewall (security group): New security group  
Storage (volumes): 1 volume(s) - 8 GiB

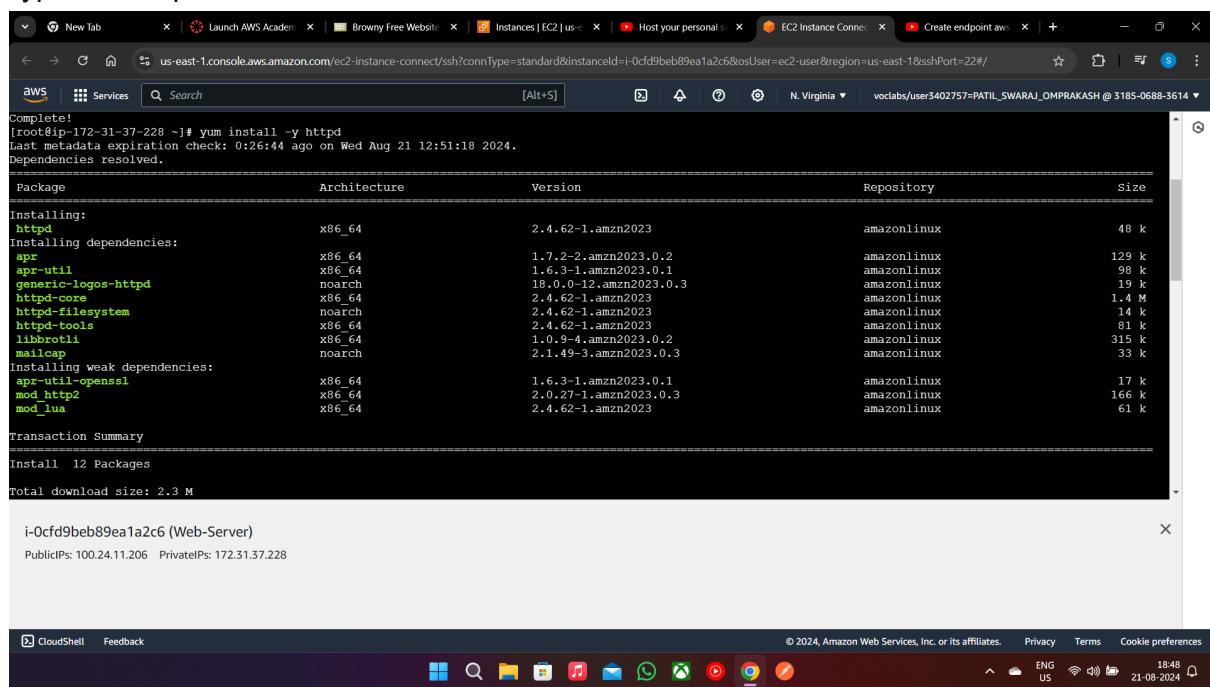
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance

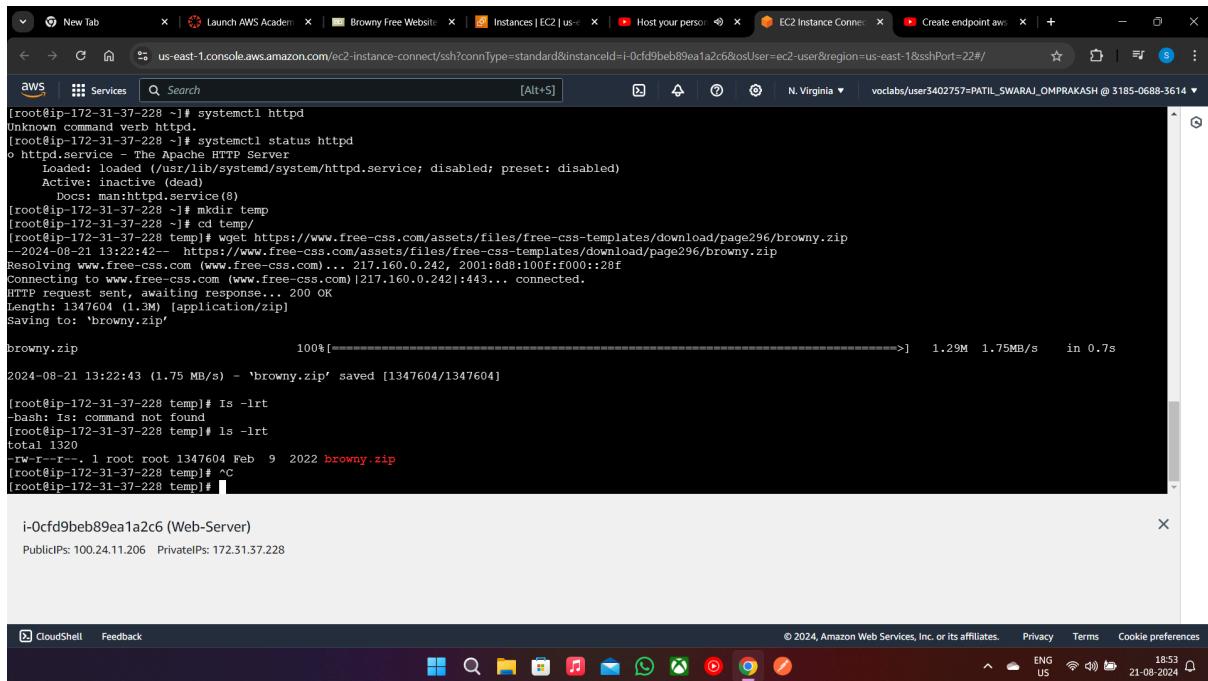
Cancel Launch instance Review commands

## Step 2: Connect To the Instance:



## Type the Required Commands:





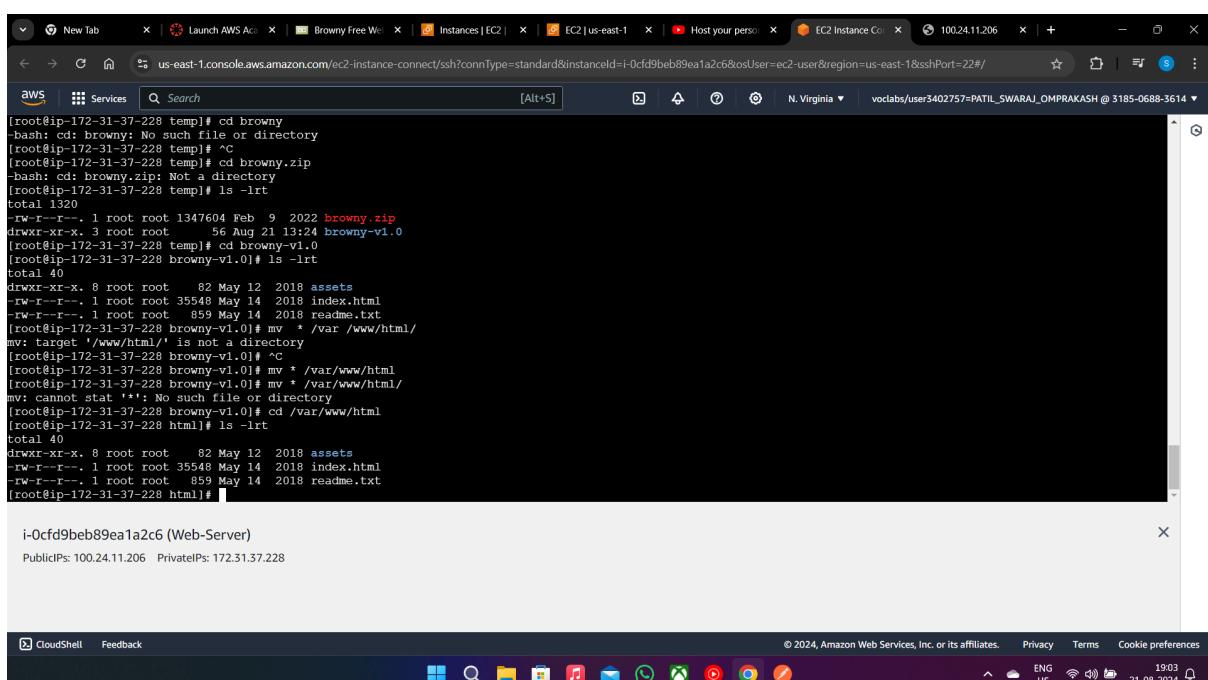
```
[root@ip-172-31-37-228 ~]# systemctl httpd
Unknown command verb httpd.
[root@ip-172-31-37-228 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
     Active: inactive (dead)
       Docs: man:httpd.service(8)
[root@ip-172-31-37-228 ~]# mkdir temp
[root@ip-172-31-37-228 temp]# wget https://www.free-css.com/assets/files/free-css-templates/download/page296/browny.zip
--2024-08-21 13:22:42-- https://www.free-css.com/assets/files/free-css-templates/download/page296/browny.zip
Resolving www.free-css.com (www.free-css.com)... 217.160.0.242, 2001:8d8:100f:f000::28E
Connecting to www.free-css.com (www.free-css.com)|217.160.0.242|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1347604 (1.3M) [application/zip]
Saving to: 'browny.zip'

browny.zip                                              100%[=====]  1.29M  1.75MB/s    in 0.7s

2024-08-21 13:22:43 (1.75 MB/s) - 'browny.zip' saved [1347604/1347604]

[root@ip-172-31-37-228 temp]# ls -lrt
-bash: ls: command not found
[root@ip-172-31-37-228 temp]# ls -lrt
total 1320
-rw-r--r--. 1 root root 1347604 Feb  9  2022 browny.zip
[root@ip-172-31-37-228 temp]# ^C
[root@ip-172-31-37-228 temp]# 
```

i-0cf9beb89ea1a2c6 (Web-Server)  
PublicIPs: 100.24.11.206 PrivateIPs: 172.31.37.228



```
[root@ip-172-31-37-228 temp]# cd browny
-bash: cd: browny: No such file or directory
[root@ip-172-31-37-228 temp]# ^C
[root@ip-172-31-37-228 temp]# cd browny.zip
-bash: cd: browny.zip: Not a directory
[root@ip-172-31-37-228 temp]# ls -lrt
total 1320
-rw-r--r--. 1 root root 1347604 Feb  9  2022 browny.zip
drwxr-xr-x. 3 root root      56 Aug 21 13:24 browny-v1.0
[root@ip-172-31-37-228 temp]# cd browny-v1.0
[root@ip-172-31-37-228 browny-v1.0]# ls -lrt
total 40
drwxr-xr-x. 8 root root     82 May 12  2018 assets
-rw-r--r--. 1 root root 35548 May 14  2018 index.html
-rw-r--r--. 1 root root    859 May 14  2018 readme.txt
[root@ip-172-31-37-228 browny-v1.0]# mv * /var/www/html/
mv: target '/var/www/html/' is not a directory
[root@ip-172-31-37-228 browny-v1.0]# ^C
[root@ip-172-31-37-228 browny-v1.0]# mv * /var/www/html/
[root@ip-172-31-37-228 browny-v1.0]# mv * /var/www/html/
mv: cannot stat '*': No such file or directory
[root@ip-172-31-37-228 browny-v1.0]# cd /var/www/html
[root@ip-172-31-37-228 html]# ls -lrt
total 40
drwxr-xr-x. 8 root root     82 May 12  2018 assets
-rw-r--r--. 1 root root 35548 May 14  2018 index.html
-rw-r--r--. 1 root root    859 May 14  2018 readme.txt
[root@ip-172-31-37-228 html]# 
```

i-0cf9beb89ea1a2c6 (Web-Server)  
PublicIPs: 100.24.11.206 PrivateIPs: 172.31.37.228

## Give it Permission For HTTP And Https:

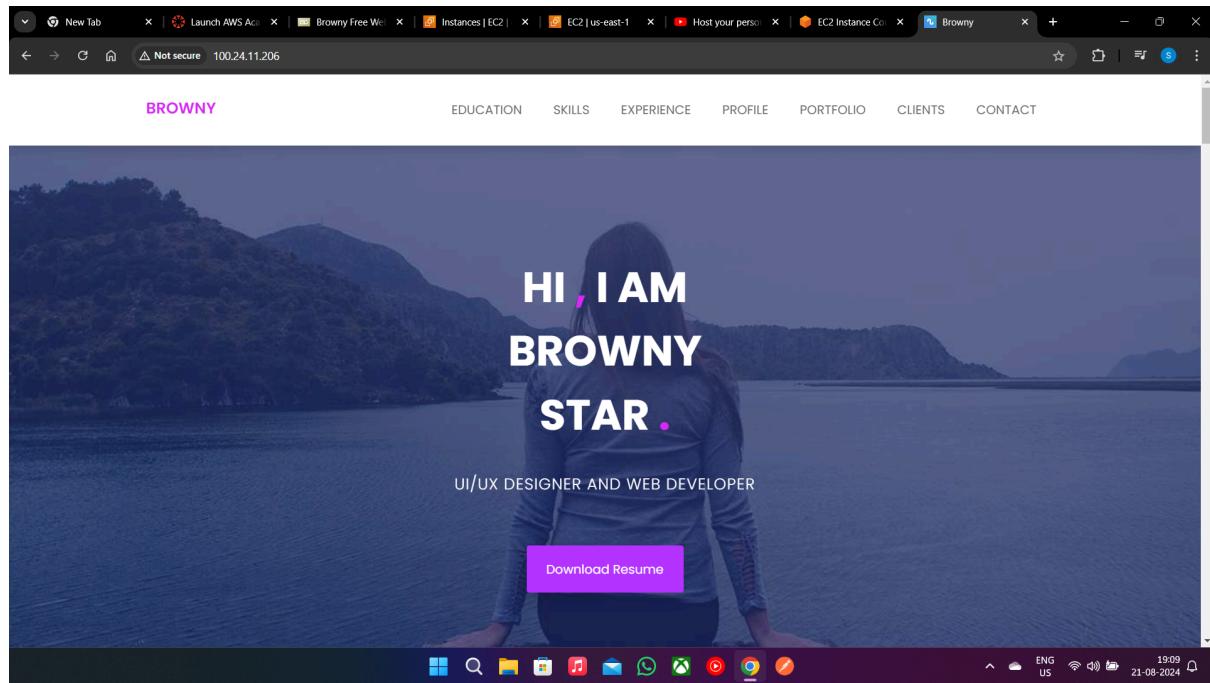
The screenshot shows the AWS CloudShell interface. The user is navigating to the AWS Management Console and selecting the EC2 service. They are viewing the details of a security group named "sg-0f1d7bd3441ae56c8". In the "Inbound rules" tab, they have added a rule allowing SSH traffic (Protocol TCP, Port 22) from their local IP address (IPv4). The browser status bar indicates the session is active on port 22.

The screenshot shows the AWS CloudShell interface with a terminal window open. The user is running the `systemctl` command to manage the Apache HTTPD service. They are listing the available options for the service. The terminal output shows the service is currently inactive (dead). The browser status bar indicates the session is active on port 22.

```
--image=PATH           Edit/enable/disable/mask unit files in the specified image
--n --lines=INTEGER    Number of journal entries to show
--o --output=STRING    Change journal output mode (short, short-precise,
                      short-iso, short-iso-precise, short-full,
                      short-monotonic, short-unix, short-delta,
                      verbose, export, json, json-pretty, json-sse, cat)
--firmware-setup       Tell the firmware to show the setup menu on next boot
--boot-loader-menu=TIME
                      Boot into boot loader menu on next boot
--boot-loader-entry=NAME
                      Boot into a specific boot loader entry on next boot
--plain               Print unit dependencies as a list instead of a tree
--timestamp=FORMAT    Change format of printed timestamps (pretty, unix,
                      us, utc, us+utc)
--read-only            Create read-only bind mount
--mkdir                Create directory before mounting, if missing
--marked               Restart/reload previously marked units

See the systemctl(1) man page for details.
log file: /var/log/systemctl.log
C
[root@ip-172-31-37-228 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
  Active: inactive (dead)
    Docs: man:httpd.service(8)
[root@ip-172-31-37-228 ~]#
```

## THE HOSTED WEBSITE:



## 2) HOSTING THROUGH CLOUD9:

### Step 1: CREATE A CLOUD9 ENVIRONMENT:

A screenshot of the AWS Cloud9 'Create environment' interface. The browser address bar shows 'us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/create/'. The main area is titled 'Create environment' with a 'Details' tab selected. It asks for a 'Name' (with a note about 60 characters) and a 'Description - optional' (with a note about 200 characters). Below this is a section for 'Environment type' with two options: 'New EC2 instance' (selected) and 'Existing compute'. A note says 'Cloud9 creates an EC2 instance in your account. The configuration of your EC2 instance cannot be changed by Cloud9 after creation.' Another note for 'Existing compute' says 'You have an existing instance or server that you'd like to use.' The bottom of the screen shows the AWS footer and a taskbar.

## STEP 2:

The screenshot shows the 'Network settings' step of the AWS Cloud9 environment creation wizard. It includes options for 'AWS Systems Manager (SSM)' (selected) and 'Secure Shell (SSH)'. A note states that SSM allows access via SSM without opening inbound ports. A 'Tags - optional' section is also present. A callout box highlights the creation of IAM resources: 'AWSServiceRoleForAWSCloud9' and 'AWSCloud9SSMAccessRole'.

**Connection**  
How your environment is accessed.

AWS Systems Manager (SSM)  
Accesses environment via SSM without opening inbound ports (no ingress).

Secure Shell (SSH)  
Accesses environment directly via SSH, opens inbound ports.

▶ VPC settings [Info](#)

▶ Tags - optional [Info](#)  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

**The following IAM resources will be created in your account**

- AWSServiceRoleForAWSCloud9 - AWS Cloud9 creates a service-linked role for you. This allows AWS Cloud9 to call other AWS services on your behalf. You can delete the role from the AWS IAM console once you no longer have any AWS Cloud9 environments. [Learn more](#)
- AWSCloud9SSMAccessRole and AWSCloud9SSMInstanceProfile - A service role and an instance profile are automatically created if Cloud9 accesses its EC2 instance through AWS Systems Manager. If your environments no longer require EC2 instances that block incoming traffic, you can delete these roles using the AWS IAM console. [Learn more](#)

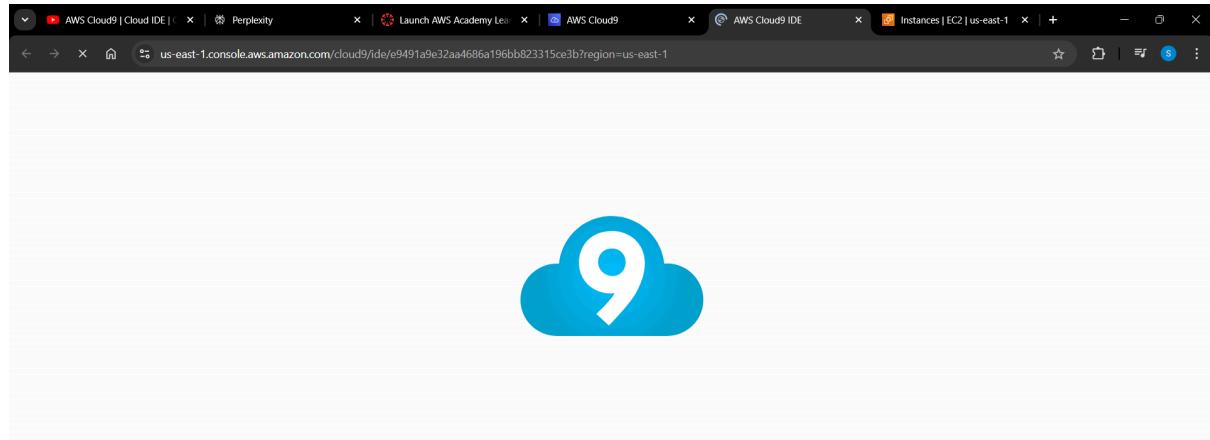
Cancel [Create](#)

The screenshot shows the 'Environments' list in the AWS Cloud9 console. A success message indicates the environment was created. The table lists the single environment 'AWS Cloud9' with details like Cloud9 IDE (Open), Environment type (EC2 instance), Connection (Secure Shell (SSH)), and Permissions (Owner). The ARN is also listed.

Name	Cloud9 IDE	Environment type	Connection	Permissions	Owner ARN
AWS Cloud9	<a href="#">Open</a>	EC2 instance	Secure Shell (SSH)	Owner	arn:aws:sts::318506883614:assumed-role/voclabs/user3402757=PATIL_SWARAJ_OMPRAKASH

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG US 19:46 22-08-2024

### STEP 3: OPEN IN THE CLOUD9 IDE:



To rename a variable, highlight it then press Ctrl-Alt-R.

