

Practical File

Submitted in partial fulfilment of the requirement for the award of

Degree of

BACHELOR OF COMPUTER APPLICATION



**CAREER POINT
UNIVERSITY**

Submitted By

Devendra Agarwal (K24511)

Submitted To

Dr. Abid Hussain

**SCHOOL OF COMPUTER APPLICATION AND TECHNOLOGY
CAREER POINT UNIVERSITY, KOTA**

AWS Cloud Essential: Concepts to Practice (CAL755)

2025-2026

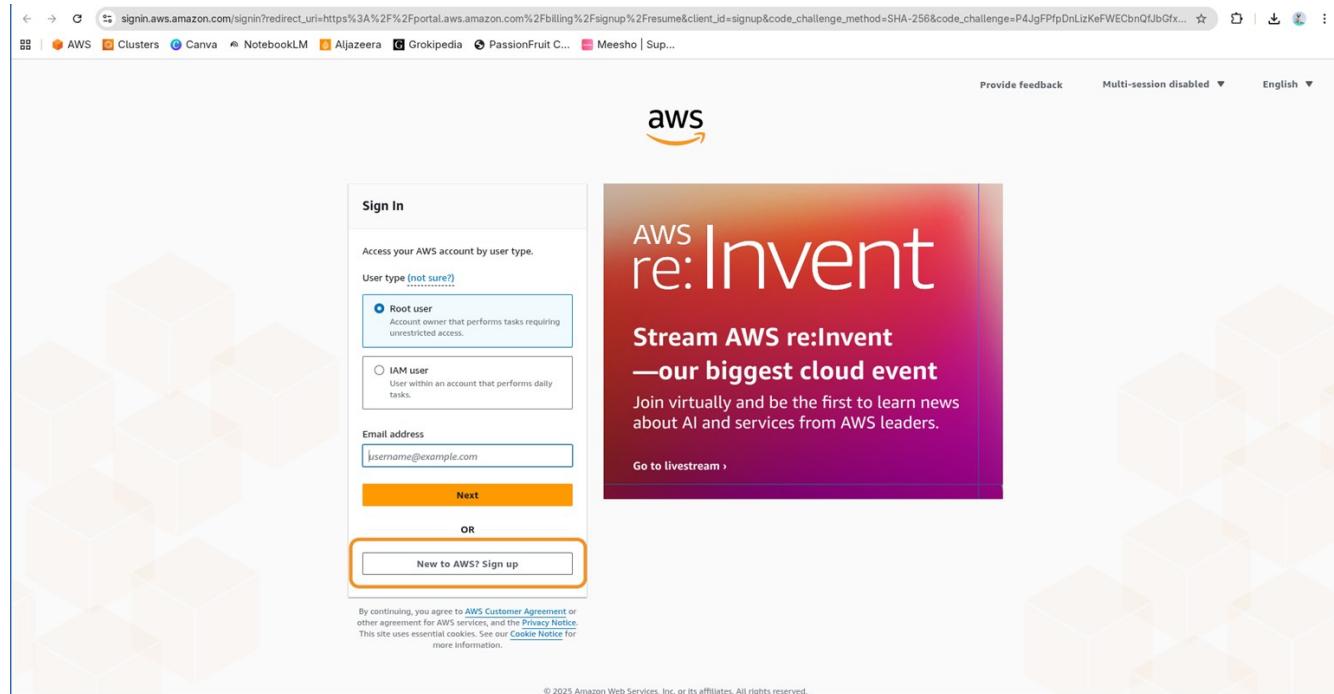
Step 1: First Open your web browser and search for AWS Login Console and click on the first link. As shown in the picture below

The screenshot shows a Google search results page with the query "aws console login". The top result is a link to the "AWS Management Console" on Amazon.com, which is highlighted with a yellow box. Below the search bar, there are filters for "AI Mode", "All", "Images", "Shopping", "Videos", "News", "Web", "More", and "Tools". The search results list several options under "AWS Management Console" and "AWS Activate Console". At the bottom, there's a link to "More results from amazon.com".

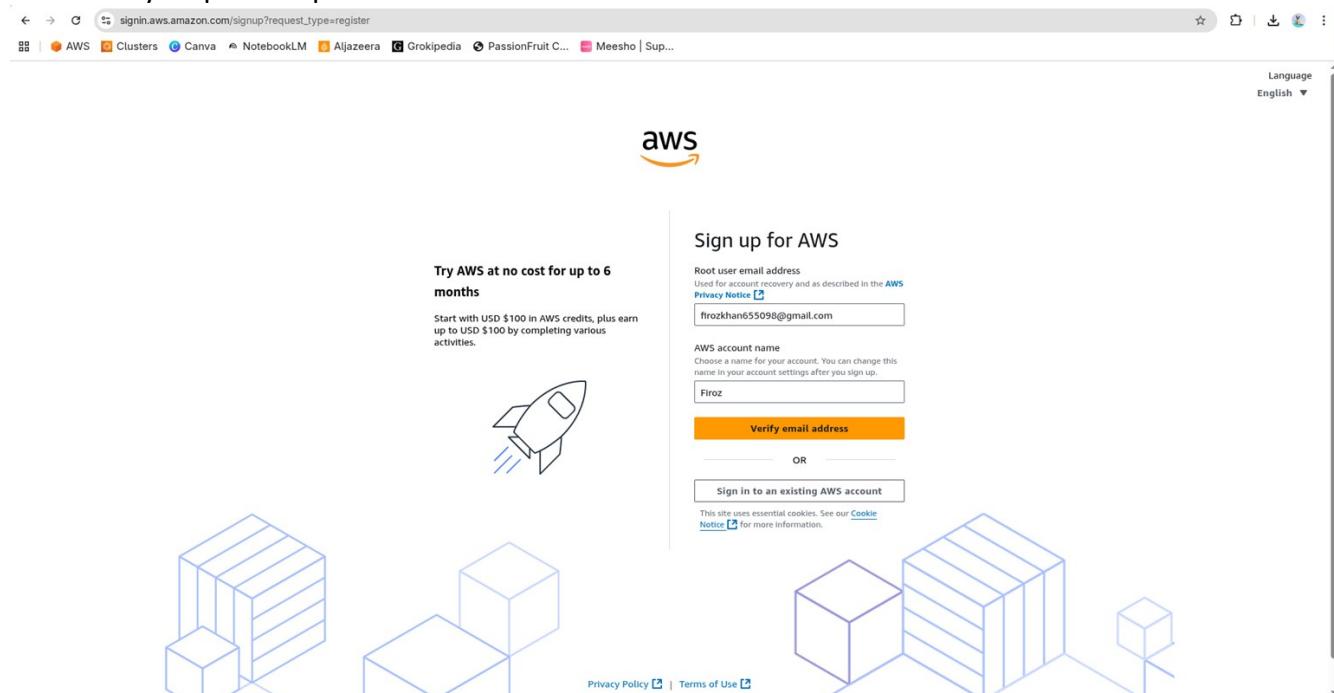
Step 2: An AWS Login Console page will open now click on Create an AWS Account.

The screenshot shows the AWS Management Console homepage. The "Create account" button in the top right corner is highlighted with a yellow box. The page features the AWS logo and navigation links for "Discover AWS", "Products", "Solutions", "Pricing", and "Resources". A search bar and a "Sign in to console" link are also present. The main content area is titled "AWS Management Console" and includes sections for "Trainings and certifications", "Free self-paced, online training", "AWS Live Training", and "Choose your AWS Certification path". A live chat window in the bottom right corner says, "Hi, I can connect you with an AWS representative or answer questions you have on AWS." There are also "Feedback" and "Star" buttons.

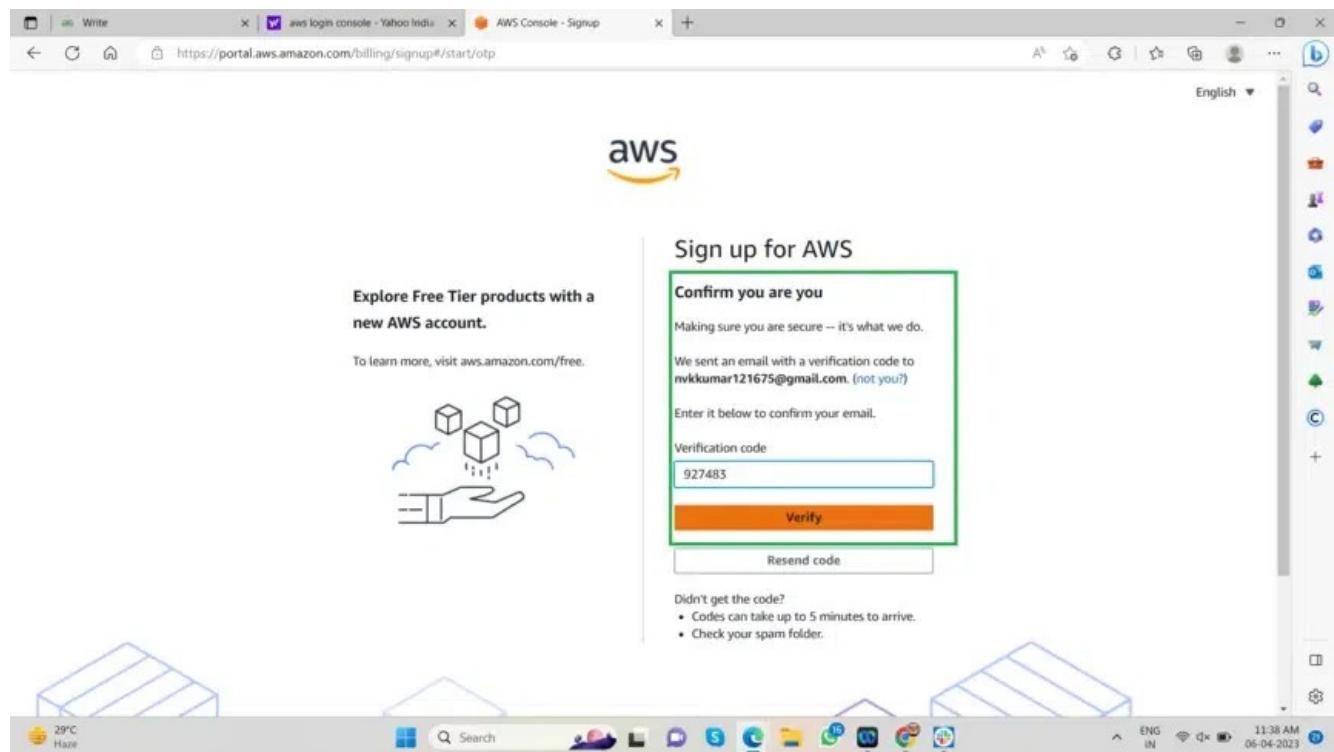
Step 3: A new AWS sign-in page will now open after selecting Create an AWS Account. Choose to Create a new AWS account. As shown in the image below



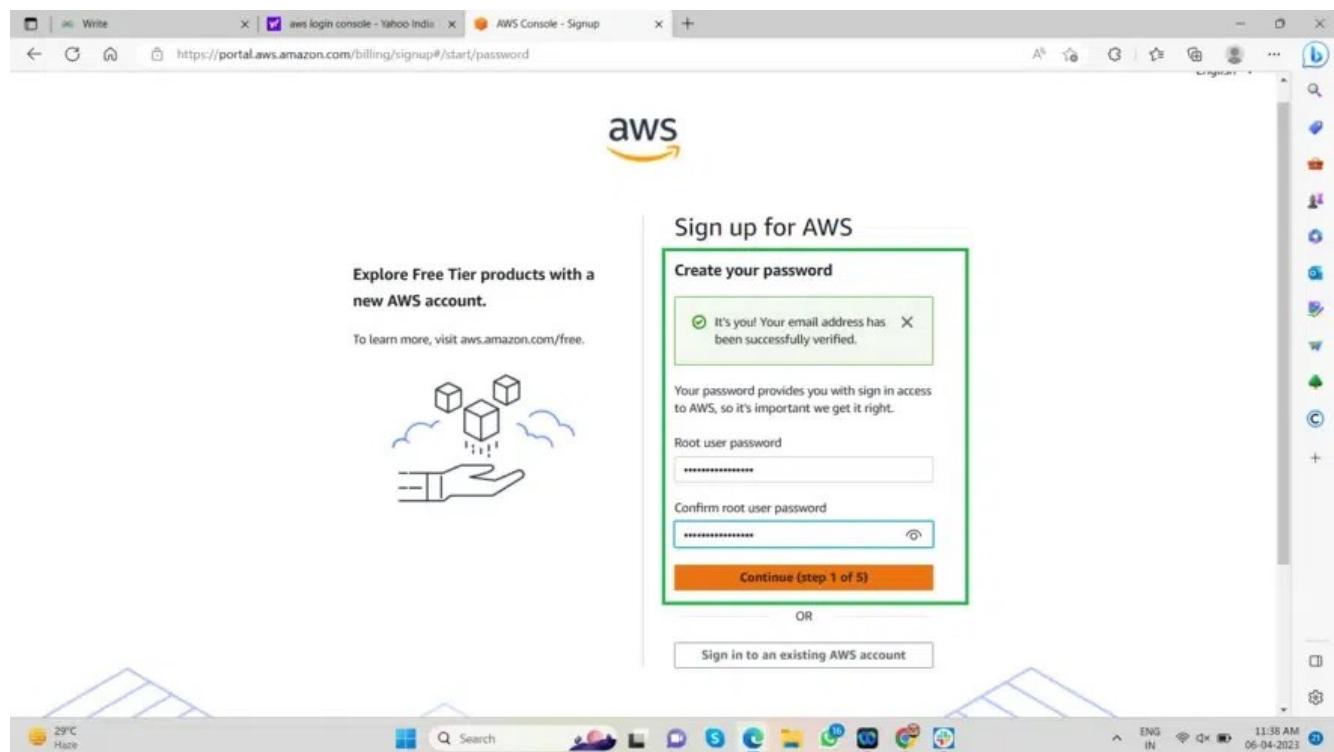
Step 4: In order to use the feature to log into an AWS Free Tire Account, we must validate the email address and have to provide the AWS account name in this stage. After clicking on "Verify Email Address," you will receive a verification code at the address you provided. Next, you must create a password for this account. Finally, click "Continue" to move on to the next stage. The pictures below show every step of the process.



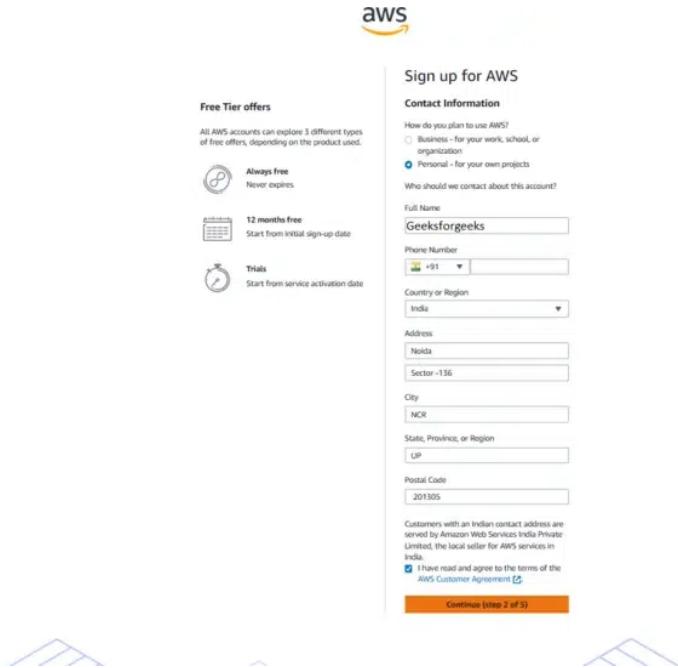
After clicking on "Verify Email Address," you will receive a verification code at the address you provided.



Next, you must create a password for this account. Finally, click "Continue" to move on to the next stage.

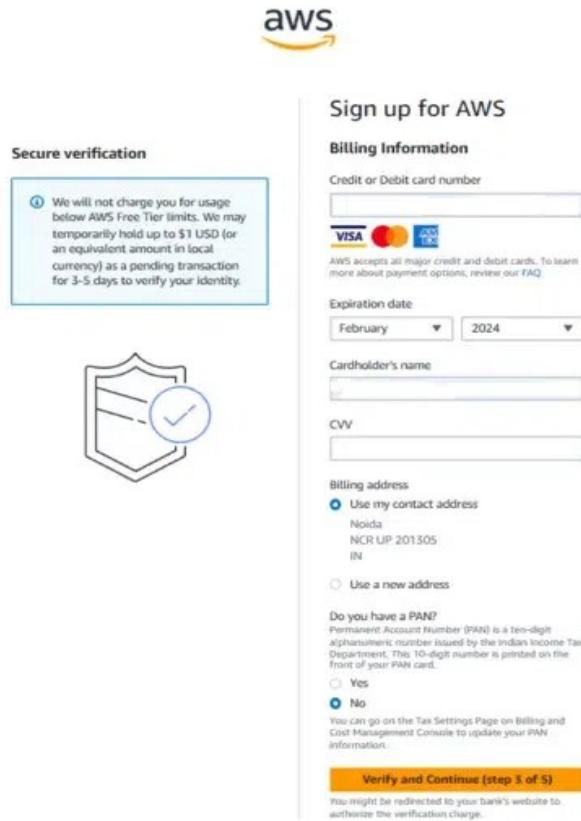


Step 5: We must include all of our contact information in this phase to make it easier for Amazon support personnel to get in touch with us about our AWS Account and any feature references. As shown in the image below.



The image shows the 'Sign up for AWS' contact information form. At the top, there's a section for 'Free Tier offers' with three options: 'Always free' (Never expires), '12 months free' (Start from initial sign-up date), and 'Trials' (Start from service activation date). Below this, the 'Contact Information' section asks 'How do you plan to use AWS?' with two radio button options: 'Business - for your work, school, or organization' (unchecked) and 'Personal - for your own projects' (checked). It also asks 'Who should we contact about this account?'. The form includes fields for 'Full Name' (Geeksforgeeks), 'Phone Number' (+91), 'Country or Region' (India), 'Address' (Noida), 'Sector -136', 'City' (NCR), 'State, Province, or Region' (UP), and 'Postal Code' (201305). A note at the bottom states: 'Customers with an Indian contact address are served by Amazon Web Services India Private Limited, the local seller for AWS services in India.' There is a checked checkbox for 'I have read and agree to the terms of the AWS Customer Agreement' with a link. A 'Continue (Step 2 of 3)' button is at the bottom right.

Step 6: We must provide the credit/debit card information in this step. There is no reason to panic at this time. AWS won't deduct any amount unless you pay it on your own. AWS may temporarily keep your identification that they will charge you only 2 Indian rupees.



The image shows the 'Sign up for AWS' billing information page. At the top right is the AWS logo. Below it, the heading 'Sign up for AWS' is followed by 'Billing Information'. On the left, a 'Secure verification' section contains a note: 'We will not charge you for usage below AWS Free Tier limits. We may temporarily hold up to \$1 USD (or an equivalent amount in local currency) as a pending transaction for 3-5 days to verify your identity.' Below this note is a shield icon with a checkmark inside. The main form area includes fields for 'Credit or Debit card number' (with a placeholder for VISA, MasterCard, and American Express), 'Expiration date' (set to February 2024), 'Cardholder's name' (placeholder 'John Doe'), 'CVV' (placeholder '123'), and 'Billing address' (selected 'Use my contact address' with 'Noida NCR UP 201305 IN'). A 'Do you have a PAN?' section is present with 'Yes' and 'No' radio buttons, both unselected. A note states: 'Permanent Account Number (PAN) is a ten-digit alphanumeric number issued by the India's Income Tax Department. This 10-digit number is printed on the front of your PAN card.' At the bottom is a yellow 'Verify and Continue (step 3 of 5)' button with the note: 'You might be redirected to your bank's website to authorize the verification charge.'

Step 7: We have to verify our phone number in this phase. As seen in the image below, select "TEXT or Voice call" as the method for receiving your verification number, then complete the captcha by clicking on "Send SMS." You will be sent to a screen where you must confirm the verification code you have received and click continue to proceed to the following stage. As seen in the pictures below

aws

Sign up for AWS

Confirm your identity

Before you can use your AWS account, you must verify your phone number. When you continue, the AWS automated system will contact you with a verification code.

How should we send you the verification code?

Text message (SMS)
 Voice call

Country or region code
 india (+91)

Mobile phone number

Security check



Type the characters as shown above
 b5c8x6

Send SMS (step 4 of 5)

Step 8: Enter the verification code you received on your mobile device, validate it, and then click Continue to move on to the following stage

aws

Sign up for AWS

Confirm your identity

Verify code

1

Continue (step 4 of 5)

Having trouble? Sometimes it takes up to 10 minutes to retrieve a verification code. If it's been longer than that, return to the previous page and try again.



Step 9: Choose the support strategy you want to use. We are setting up an AWS Free Tier Account so select the Basic Support option, which is cost-free and which AWS also suggests for new customers. The Basic Support Plan includes following

- 24x7 self-service access to AWS resources
- Can access personal health dashboard
- It is free of cost



Sign up for AWS

Select a support plan

Choose a support plan for your business or personal account. [Compare plans and pricing examples](#). You can change your plan anytime in the AWS Management Console.

Basic support - Free

- Recommended for new users just getting started with AWS
- 24x7 self-service access to AWS resources
- For account and billing issues only
- Access to Personal Health Dashboard & Trusted Advisor



Developer support - From \$29/month

- Recommended for developers experimenting with AWS
- Email access to AWS Support during business hours
- 12 (business)-hour response times



Business support - From \$100/month

- Recommended for running production workloads on AWS
- 24x7 tech support via email, phone, and chat
- 1-hour response times
- Full set of Trusted Advisor best-practice recommendations



Need Enterprise level support?

From \$15,000 a month you will receive 15-minute response times and concierge-style experience with an assigned Technical Account Manager. [Learn more](#)

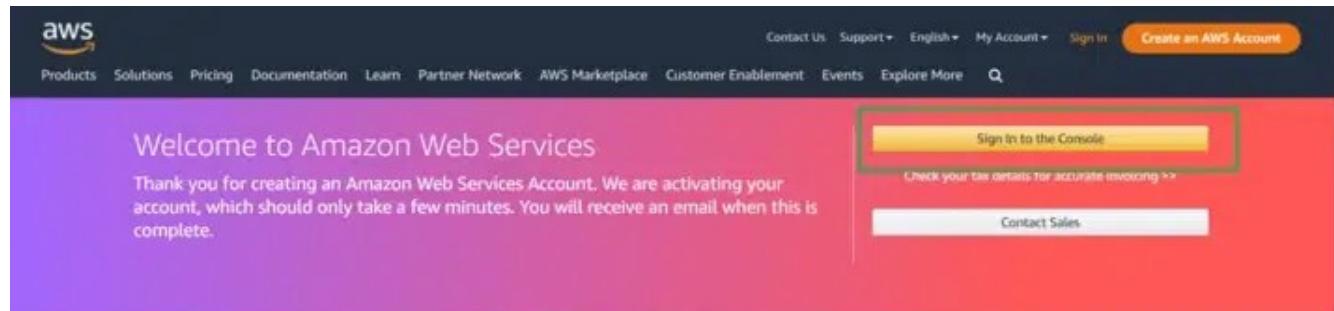
[Complete sign up](#)

After selecting a plan, click "Complete the sign up" as shown in the image.

Step 10: "Congratulation" Upon the creation of your AWS account, you can sign in by clicking Click Sign into the console once more, input the email address that you provided, your password, and then click Sign in as shown in the accompanying image, where you can see AWS Management Console's home page for certain of its offering services.

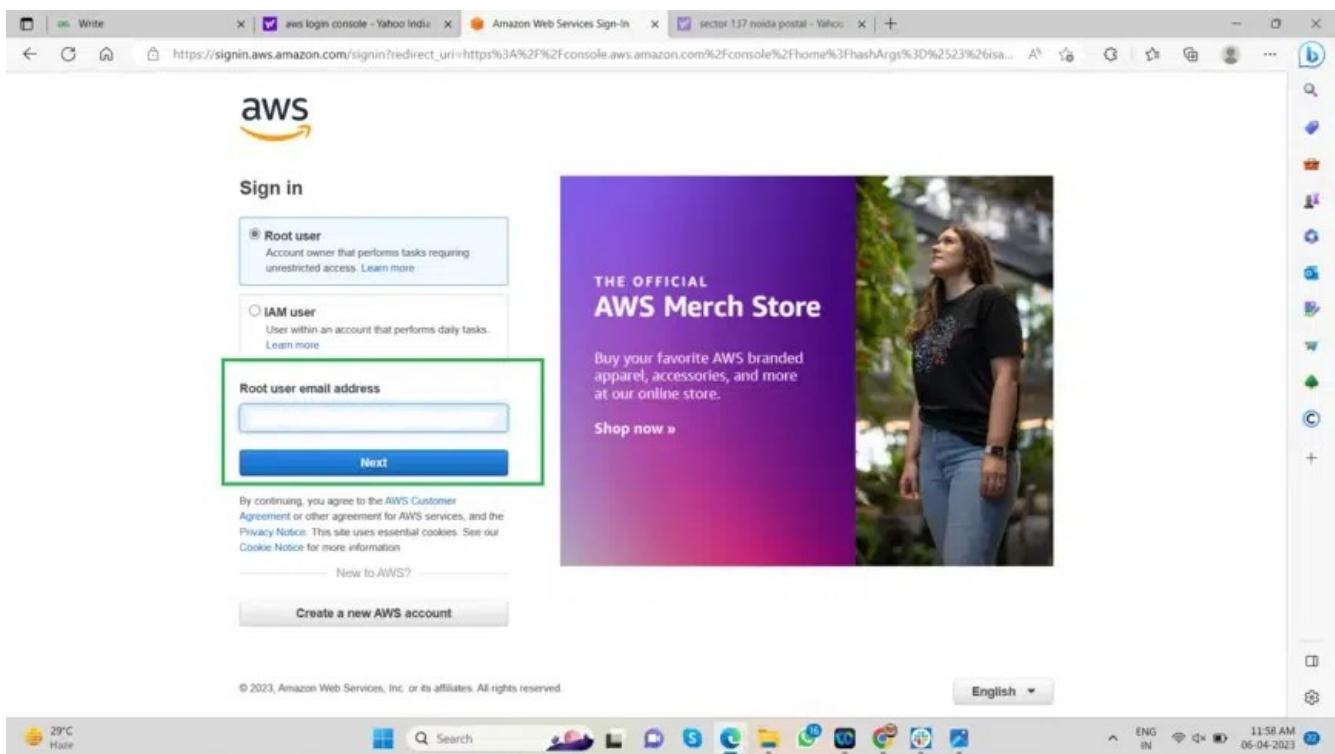


The screenshot shows the AWS Management Console home page. At the top center is the AWS logo. Below it is a graphic of a rocket launching from a cloud. The main heading is "Congratulations". Below that, a message says "Thank you for signing up for AWS. We are activating your account, which should only take a few minutes. You will receive an email when this is complete." A green-bordered button labeled "Go to the AWS Management Console" is centered. Below the button, smaller text says "Sign up for another account or contact sales." At the bottom, there are three large 3D wireframe cube icons representing different AWS services. In the center between them is a link to "Privacy Policy" and "Terms of Use". Below the cubes, the text "Amazon Web Services, Inc. or its affiliates. All rights reserved." is visible. The top navigation bar includes links for Contact Us, Support, English, My Account, Sign In, and Create an AWS Account.



The screenshot shows the AWS Management Console home page with a purple header. The header contains the AWS logo, a search bar, and links for Products, Solutions, Pricing, Documentation, Learn, Partner Network, AWS Marketplace, Customer Enablement, Events, Explore More, and Contact Us. The main content area has a purple background and displays the message "Welcome to Amazon Web Services" and "Thank you for creating an Amazon Web Services Account. We are activating your account, which should only take a few minutes. You will receive an email when this is complete." To the right, there is a yellow button labeled "Sign In to the Console" and a link to "Check your tax details for accurate invoicing >>". Below the main content, there is a section titled "Personalize Your Experience" with a sub-instruction "Fill in the blanks below to receive recommendations catered to your role and interests." It includes fields for "My role is:" and "I am interested in:", both with dropdown menus. There is also a checkbox for receiving news updates and a "Submit" button at the bottom.

Enter your email address and previously-configured password.



And this is the Amazon Console Home page, where you may access some of the most popular AWS services, including EC2, VPC, AUTOSCALING, etc.

Console Home [Info](#)

[Reset to default layout](#) [+ Add widgets](#)

Recently visited [Info](#)

- FSx
- Storage Gateway
- Simple Notification Service
- Elastic Kubernetes Service
- AWS Auto Scaling
- Support
- VPC

[View all services](#)

Welcome to AWS

- Getting started with AWS** [Info](#)
Learn the fundamentals and find valuable information to get the most out of AWS.
- Training and certification** [Info](#)
Learn from AWS experts and advance your skills and knowledge.
- What's new with AWS?** [Info](#)
Discover new AWS services, features, and Regions.

AWS Health [Info](#)

Open issues **0** Past 7 days

Cost and usage [Info](#)

© 2023, Amazon Web Services India Private Limited or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

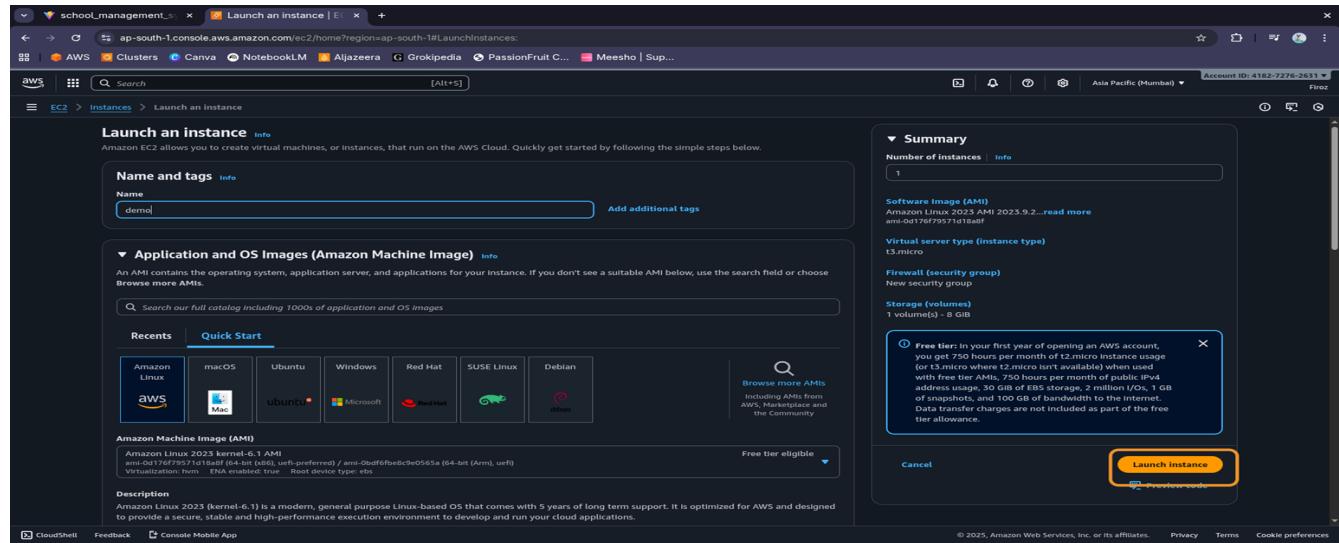
2. Identify and demonstrate one example service for each model (e.g., EC2, Elastic Beanstalk, AWS WorkSpaces).

EC2 (IaaS - Infrastructure as a Service)

Example Service: Virtual servers in the cloud

Steps:

1. Navigate to EC2 Dashboard
2. Click "Launch Instance"
3. Configure computing resources as needed



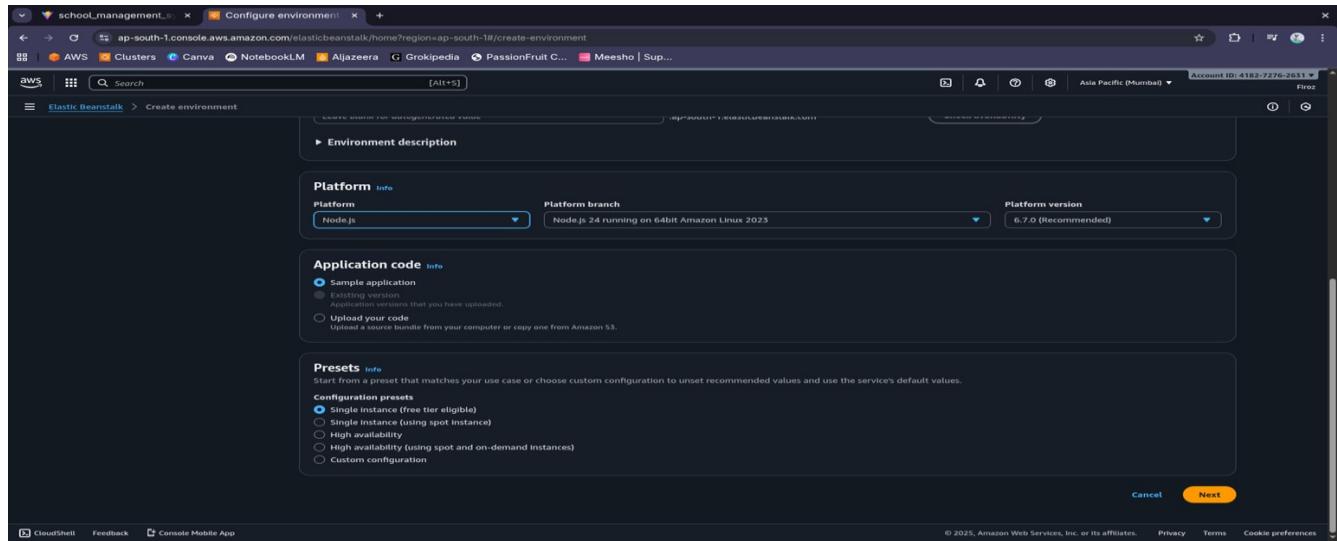
Elastic Beanstalk (PaaS - Platform as a Service)

Example Service: Web application deployment platform

Steps:

1. Go to Elastic Beanstalk console
2. Create new application environment

3. Upload application code

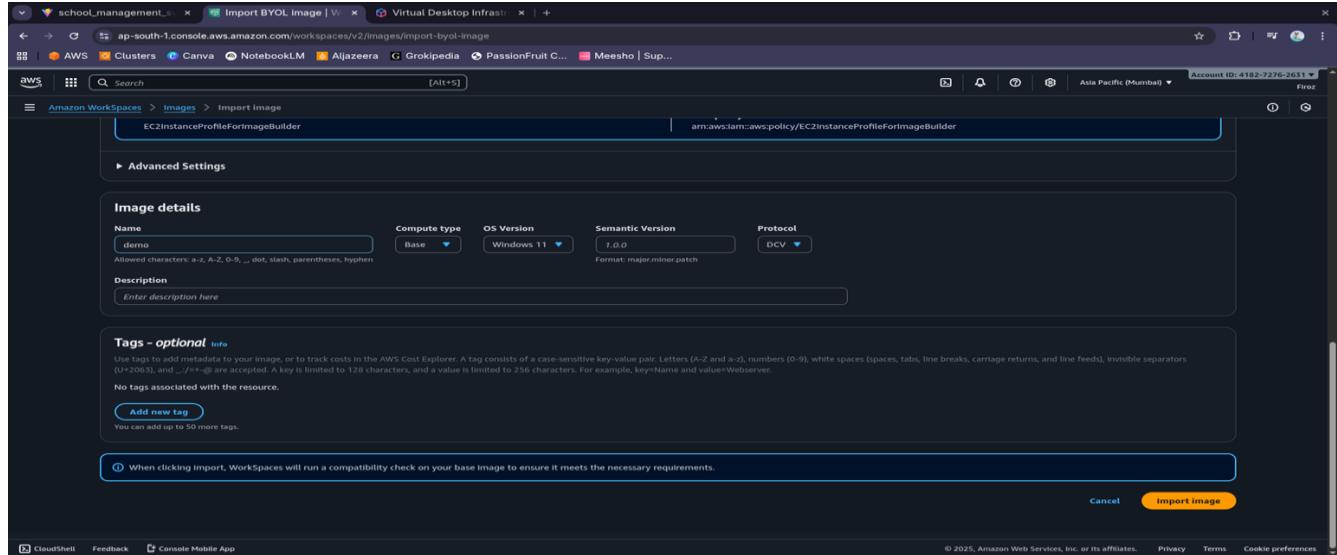


AWS WorkSpaces (DaaS - Desktop as a Service)

Example Service: Virtual desktop infrastructure

Steps:

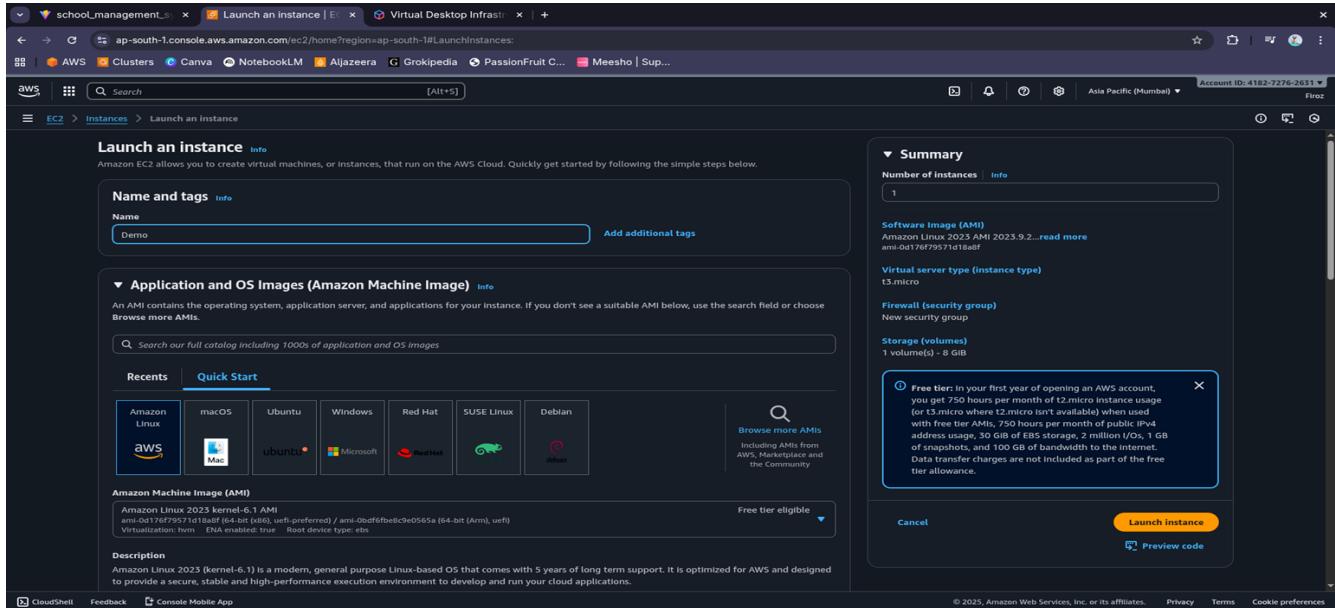
1. Access WorkSpaces console
2. Provision virtual desktops
3. Assign to users



3. Launch an instance, connect via SSH, check default network settings, and stop/terminate safely.

1. Navigate to EC2 Dashboard

- Go to AWS Management Console
- Search for "EC2" and select



2. Launch Instance

- Click "Launch Instance"
- Name: "test-instance"

The screenshot shows the AWS EC2 Dashboard for the Asia Pacific (Mumbai) Region. The left sidebar includes sections for Dashboard, Instances (with sub-options like Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Capacity Manager), Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), and Network & Security (Security Groups, Elastic IPs, Placement Groups). The main content area displays 'Resources' (Instances running: 0, Auto Scaling Groups: 0, Capacity Reservations: 0, Dedicated Hosts: 0, Elastic IPs: 0, Instances: 0, Key pairs: 15, Load balancers: 0, Placement groups: 0, Security groups: 17, Snapshots: 0, Volumes: 0), 'Launch instance' (Launch instance button, Migrate a server link), 'Service health' (Region: Asia Pacific (Mumbai), Status: This service is operating normally), 'Zones' (Zone name: ap-south-1a, ap-south-1b, ap-south-1c, Zone ID: aps1-az1, aps1-az3, aps1-az2), 'Instance alarms' (0 in alarm, 0 OK, 0 insufficient data), 'Scheduled events' (No scheduled events), and 'Account attributes' (Default VPC: vpc-015a6441cb23dd9a5, Settings, Explore AWS, Additional information). The URL in the address bar is https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Volumes:.

3. Choose AMI

- Select "Amazon Linux 2023 AMI"

The screenshot shows the 'Launch an instance' wizard. The first step, 'Name and tags', has a 'Name' field containing 'Demo'. The second step, 'Application and OS Images (Amazon Machine Image)', shows a search bar and a grid of recent and quick start AMIs. An orange box highlights the 'Amazon Linux' AMI, which is selected. Its details are shown in a modal: 'Amazon Linux 2023 kernel-6.1 AMI', 'ami-0d176f9571cf1afaf (64-bit (x86), uefi-preferred) / ami-0bdf6fb8c0e0565a (64-bit (Arm), uefi)', 'Virtualization: hvm', 'ENI enabled: true', 'Root device type: ebs'. The third step, 'Summary', shows 'Number of instances' (1), 'Software Image (AMI)' (Amazon Linux 2023 AMI 2023.9.2...), 'Virtual server type (instance type)' (t3.micro), 'Firewall (security group)' (New security group), 'Storage (volumes)' (1 volume(s) - 8 GiB), and a summary of free tier benefits. The 'Launch instance' button is at the bottom right. The URL in the address bar is https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:.

4. Instance Type

- Select "t2.micro" (Free tier eligible)

The screenshot shows the AWS EC2 'Launch an instance' wizard. In the 'Instance type' section, 't2.micro' is selected, which is highlighted as 'Free tier eligible'. Other options like 't3.micro' and 'On-Demand SUSE base pricing' are also listed. Below this, the 'Key pair (login)' section is shown, where a key pair must be selected or created. The 'Network settings' section shows the VPC and subnet configuration. On the right side, the 'Summary' panel indicates 1 instance is being launched, and the 'Software Image (AMI)' is set to 'Amazon Linux 2023 AMI 2023.9.2...'. A note about free tier usage is present. At the bottom right are 'Cancel', 'Launch instance', and 'Preview code' buttons.

5. Key Pair

- Create new key pair: "assignment-key"
- Download .pem file

This screenshot continues the 'Launch an instance' wizard. In the 'Key pair (login)' section, a new key pair 'assignment-key' is selected. The 'Network settings' section includes a 'Firewall (security groups)' section where a new security group 'launch-wizard-17' is being created. Under this group, three rules are defined: 'Allow SSH traffic from Anywhere', 'Allow HTTPS traffic from the internet', and 'Allow HTTP traffic from the internet'. The 'Summary' panel on the right shows the same configuration as the previous screenshot, including the 'Amazon Linux 2023 AMI 2023.9.2...' AMI. The 'Launch instance' button is visible at the bottom right.

6. Network Settings

- Allow SSH traffic
- Create security group

The screenshot shows the AWS EC2 'Launch an instance' wizard. In the 'Network settings' section, under 'Firewall (security group)', there is a checkbox labeled 'Allow SSH traffic from' which is checked. This checkbox has a yellow border around it. Below this, there are other options like 'Allow HTTPS traffic from the internet' and 'Allow HTTP traffic from the internet'. In the 'Summary' section on the right, there is a note about the free tier for t3.micro instances.

7. Launch

- Click "Launch Instance"

The screenshot shows the final configuration screen of the AWS EC2 'Launch an instance' wizard. It includes sections for 'Name and tags', 'Application and OS Images (Amazon Machine Image)', and 'Description'. The 'Launch instance' button at the bottom right is highlighted with a yellow box. A note about the free tier for t3.micro instances is also present.

Connect via SSH

```
user1@Ubuntu18b:~/Downloads$ ssh -i blog02-temp.pem ubuntu@52.55.222.44
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1045-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 System information as of Wed Nov  3 09:20:33 UTC 2021

 System load:  0.0          Processes:           113
 Usage of /:   26.4% of 7.69GB   Users logged in:    1
 Memory usage: 24%
 Swap usage:  0%          IPv4 address for eth0: 172.11.0.200

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

Last login: Wed Nov  3 06:17:43 2021 from 176.77.177.137
ubuntu@ip-172-11-0-200:~$ █
```

Stop/Terminate Safely

1. Stop Instance

- Select instance in EC2 console
- Instance State → Stop

2. Terminate Instance

- Select stopped instance
- Instance State → Terminate
- Confirm termination

EC2 Management Console

https://ap-south-1.console.aws.amazon.com/ec2/v2/home?region=ap-south-1#Instances:v=3;sort=desc\$instanceState

Mumbai Manu Manjunatha

New EC2 Experience Tell us what you think

EC2 Dashboard

EC2 Global View

Events

Tags

Limits

Instances

- Instances New
- Launch Types
- Launch Templates
- Spot Requests
- Savings Plans
- Reserved Instances New
- Dedicated Hosts
- Capacity Reservations

Images

- AMIs New
- AMI Catalog

Elastic Block Store

- Volumes New

Instances (1/1) Info

Search

Connect

Stop instance (highlighted)

Start instance

Reboot instance

Hibernate instance

Terminate instance

Name	Instance ID	Instance state	Instance type	Status
testmachine	i-08bfd70493f5a756a	Running	t2.micro	2/2

Availability Zone Public IPv4 DNS

ap-south-1a ec2-15-207-108-

Instance: i-08bfd70493f5a756a (testmachine)

Details Security Networking Storage Status checks Monitoring Tags

Instance summary Info

Instance ID i-08bfd70493f5a756a (testmachine)	Public IPv4 address 15.207.108.247 open address	Private IPv4 addresses 172.31.44.139
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-15-207-108-247.ap-south-1.compute.amazonaws.com open address
Hostname type	Private IP DNS name (IPv4 only)	Answer private resource DNS name

4. Demonstrate the use of key pairs by creating a new key pair and connecting to EC2 using a terminal tool (PuTTY or VS Code).

Create New Key Pair

1. EC2 Console

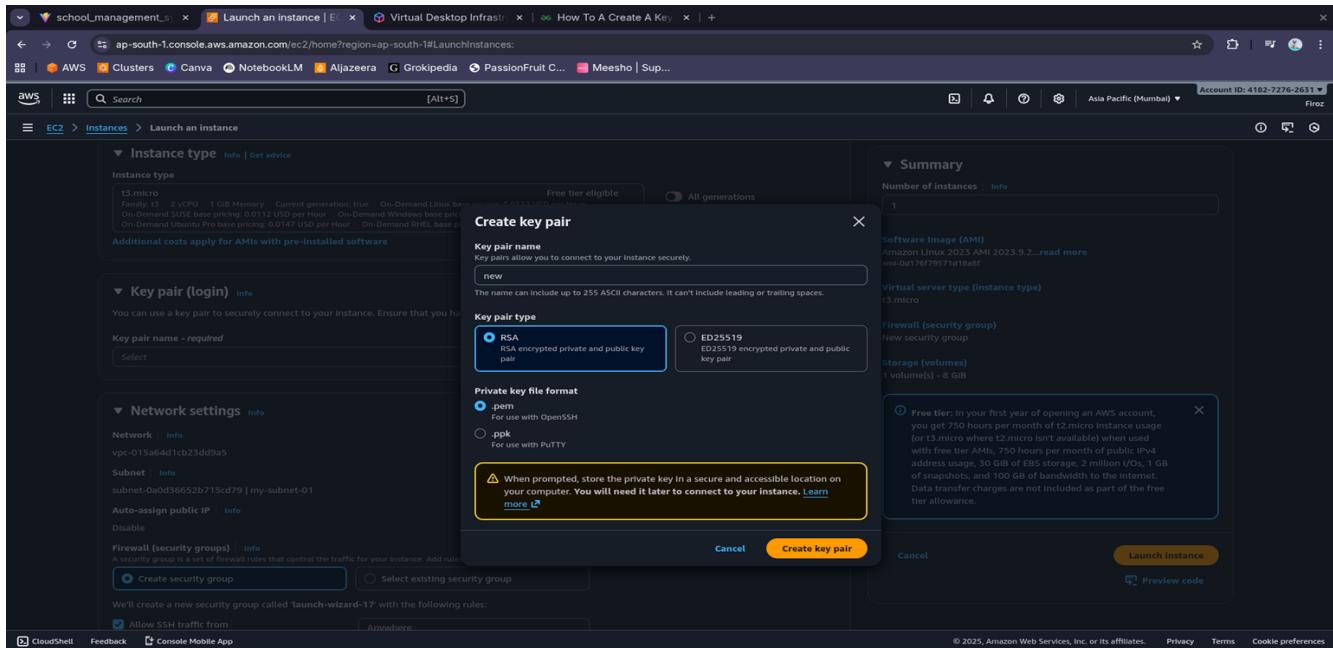
- Navigate to "Key Pairs" under Network & Security
- Click "Create key pair"

The screenshot shows the AWS EC2 Instances Launch an instance wizard. The process is at Step 1: Set instance type and key pair. The 'Instance type' section shows a t3.micro selection with details: Family: t3, 2 vCPUs, 1 GiB Memory, Current generation: true, On-Demand Linux base pricing: 0.0112 USD per Hour, On-Demand SUSE base pricing: 0.0112 USD per Hour, On-Demand Windows base pricing: 0.0112 USD per Hour, On-Demand Ubuntu base pricing: 0.0147 USD per Hour, On-Demand RHEL base pricing: 0.04 USD per Hour. It also mentions 'Additional costs apply for AMIs with pre-installed software'. The 'Key pair (login)' section has a dropdown for 'Key pair name - required' with 'Select' and a button 'Create new key pair' highlighted with a yellow box. The 'Network settings' section includes 'Network' (vpc-015a64d1cb23dd9a5), 'Subnet' (subnet-0a0d36652b715cd79), 'Auto-assign public IP' (disabled), and 'Firewall (security groups)' (Create security group selected). The 'Summary' section shows 1 instance, AMI: Amazon Linux 2025.0.2023.9.2..., Virtual server type: t3.micro, Firewall: New security group, Storage: 1 volume(s) - 8 GiB. A note about free tier usage is present. At the bottom are 'Cancel', 'Launch instance', and 'Preview code' buttons.

2. Configure Key

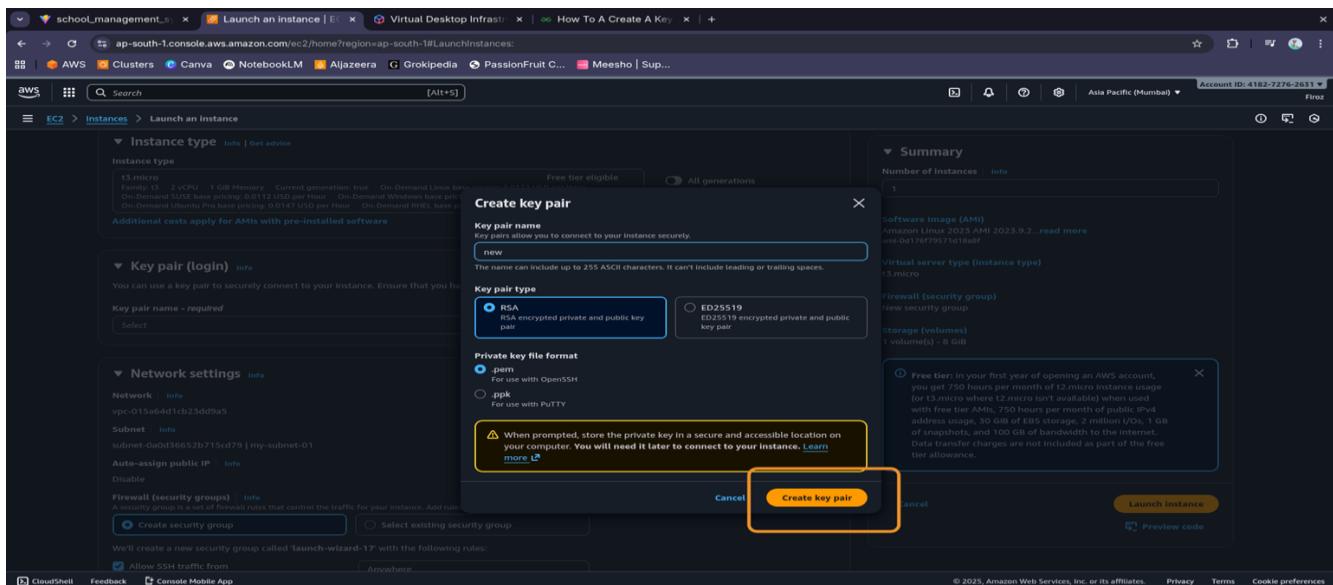
- Name: "new-assignment-key"
- Type: RSA

- Format: .pem



3. Download

- Save .pem file securely



Connect Using VS Code

1. Install Remote-SSH Extension
2. Configure SSH Config

..

Host aws-ec2

HostName <public-ip>

```
User ec2-user  
IdentityFile ~/.ssh/assignment-key.pem  
``
```

3. Connect

- Open Command Palette
- "Remote-SSH: Connect to Host"

5. Add inbound/outbound rules, test connectivity (ping, SSH, HTTP), and verify network restrictions in terms of the 5. Configure EC2 Security Group rules.

Add Inbound Rules

1. Navigate to Security Groups

- EC2 Console → Security Groups
- Select your instance's security group

2. Edit Inbound Rules

- Add Rule: SSH - Port 22 - My IP
- Add Rule: HTTP - Port 80 - 0.0.0.0/0
- Add Rule: All ICMP - IPv4 - 0.0.0.0/0

Add Outbound Rules

1. Edit Outbound Rules

- Allow All Traffic (for testing)
- Or restrict to specific ports

Test Connectivity

```
```bash
```

#### Test SSH

```
ssh -i key.pem ec2-user@<ip>
```

#### Test HTTP

```
curl http://<ip>
```

#### Test Ping

```
ping <ip>
```

```
```
```

Verify Network Restrictions

1. Remove HTTP Rule
2. Test HTTP Access (should fail)
3. Restore HTTP Rule
4. Verify Access Restored

6. Create an S3 Bucket and perform object operations with the help of Upload, download, version, and delete files; enable bucket public access settings and analyze permissions.

Create S3 Bucket

1. S3 Console

- Click "Create bucket"
- Bucket name: "assignment-bucket-unique-name"
- Region: us-east-1

The screenshot shows the 'Create bucket' wizard on the AWS S3 console. The first step, 'General configuration', is completed with the bucket name 'amzn-s3-demo-bucket' and the region set to 'Asia Pacific (Mumbai) ap-south-1'. The 'General purpose' bucket type is selected. The second step, 'Object Ownership', shows that 'ACLs disabled (recommended)' is selected. The third step, 'Optional settings', includes sections for 'Bucket Versioning' (disabled), 'Tags - optional' (empty), and 'Copy settings from existing bucket - optional' (disabled). The bottom of the page shows standard AWS navigation links and copyright information.

2. Configure Options

- Versioning: Enable
- Block Public Access: Initially keep enabled

The screenshot shows the 'Bucket Properties' page for the 'amzn-s3-demo-bucket'. The 'Versioning' section is highlighted with an orange border, showing that 'Versioning' is disabled. The 'Tags' section is also highlighted with an orange border, showing an empty tag list. Other sections like 'Public access' and 'Encryption' are visible but not highlighted.

Object Operations

1. Upload Files

- Click "Upload"

- Add multiple files
- Upload

The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with tabs like 'Clusters', 'Canvas', 'NotebookLM', 'AppStream', 'Grokpedia', 'PassionFruit C...', 'Meesho | Sup...', and 'Amazon S3'. Below the navigation bar, it says 'Amazon S3 > Buckets > demo234-4'. The main area is titled 'demo234-4' with an 'Info' tab. There are tabs for 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. Under the 'Objects' tab, it says '(0)' and 'No objects'. There's a 'Find objects by prefix' input field and a 'Show versions' toggle. A large orange box highlights the 'Upload' button at the bottom right of the object list.

2. Download Files

- Select object
- Click "Download"

This screenshot shows the same AWS S3 console as the previous one, but now it lists a single object: 'Untitled 1.ods'. The 'Download' button for this object is highlighted with an orange box. The rest of the interface is identical to the first screenshot, including the navigation bar and the 'demo234-4' bucket title.

3. Version Management

- Upload same file with changes
- View versions in object details

This screenshot shows the 'Versions' tab for the 'Untitled 1.ods' file. It displays two versions of the file. The first version is the current one, uploaded on November 25, 2025, at 17:46:41 (UTC+05:30). The second version was uploaded on November 25, 2025, at 17:45:27 (UTC+05:30). Both are 18.4 KB in size and have a Standard storage class. At the top right of the 'Versions' table, there's an 'Actions' dropdown menu with several options: 'Copy S3 URI', 'Download', 'Open L...', 'Object actions', and a 'More' option. An orange box highlights the 'Actions' button in the top right corner of the table header.

4. Delete Files

- Select object
- Click "Delete"
- Confirm deletion

The screenshot shows the AWS S3 console interface. At the top, the URL is ap-south-1.console.aws.amazon.com/s3/buckets/demo234-4?region=ap-south-1&tab=objects. The browser tabs include AWS, Clusters, Carva, NotebookLM, Aljazeera, Grokipedia, PassionFruit C..., Meesho, and Sup... The header also shows the account ID: 4182-7276-2631 and the region: Asia Pacific (Mumbai). The main content area shows a bucket named 'demo234-4'. The 'Objects' tab is active, showing one object: 'Untitled 1.ods' (Type: ods, Last modified: November 25, 2025, 17:45:27 (UTC+05:30), Size: 18.4 KB, Storage class: Standard). Below the object list are buttons for Copy S3 URI, Copy URL, Download, Open, Delete (which is highlighted with a yellow box), Actions, Create folder, and Upload.

Public Access Configuration

1. Edit Block Public Access

- Uncheck "Block all public access"
- Confirm

The screenshot shows the AWS S3 console with the 'demo234-4' bucket selected. The 'Permissions' tab is active. In the 'Block public access (bucket settings)' section, there is a note about public access being granted through access control lists (ACLs), bucket policies, and access point policies. It also notes that turning on 'Block all public access' will block public access to all objects in the bucket. A note says 'AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases.' The 'Block all public access' switch is currently set to 'On'. Below this section is a note about 'Individual Block Public Access settings for this bucket'. The 'Bucket policy' section is also visible at the bottom.

2. Bucket Policy

- Permissions → Bucket Policy
- Add public read policy:

```
```json
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Principal": "*",
 "Action": "s3:GetObject",
 "Resource": "arn:aws:s3:::assignment-bucket-unique-name/*"
 }
]
}
```
The JSON code defines a bucket policy with a single statement. The statement allows public access ('Effect: Allow') to all objects ('Resource: arn:aws:s3:::assignment-bucket-unique-name/*') for all principals ('Principal: *'). The policy is versioned to '2012-10-17'.
```

Successfully edited Block Public Access settings for this bucket.

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

No policy to display.

Object Ownership

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

[Edit](#)

CloudShell Feedback Console Mobile App © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

7. Host a simple HTML webpage on S3 and test the public URL.

Configure Static Website

1. Properties Tab

- Scroll to "Static website hosting"
- Click "Edit"

[Edit](#)

Transfer acceleration

Use an accelerated endpoint for faster data transfers. [Learn more](#)

Transfer acceleration

Disabled

[Edit](#)

Object Lock

Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely. Object Lock works only in versioned buckets. [Learn more](#)

Object Lock

Disabled

[Edit](#)

Requester pays

When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled. [Learn more](#)

Requester pays

Disabled

[Edit](#)

Static website hosting

Use this bucket to host a website or redirect requests. [Learn more](#)

We recommend using AWS Amplify Hosting for static website hosting
Deploy a fast, secure, and reliable website quickly with AWS Amplify Hosting. Learn more about [Amplify Hosting](#) or [View your existing Amplify apps](#)

[Create Amplify app](#)

S3 static website hosting

Disabled

CloudShell Feedback Console Mobile App © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

2. Enable Hosting

- Select "Enable"

- Index document: index.html
- Error document: error.html

The screenshot shows the AWS S3 Bucket static website hosting configuration page. At the top, there are tabs for 'Edit static website host' and 'Virtual Desktop Infrastr...'. Below the tabs, the URL is ap-south-1.console.aws.amazon.com/s3/bucket/demo234-4/property/website/edit?region=ap-south-1. The browser header shows 'AWS Clusters Carva NotebookLM Aljazeera Grokypedia PassionFruit C... Meesho Sup...' and 'Account ID: 4102-7276-2631'. The main content area is titled 'Static website hosting' with a sub-instruction: 'Use this bucket to host a website or redirect requests. [Learn more](#)'. A radio button group for 'Static website hosting' has 'Enable' selected. Under 'Hosting type', 'Host a static website' is selected, with a note: 'Use the bucket endpoint as the web address. [Learn more](#)'. There is also an option for 'Redirect requests for an object' with a note: 'Redirect requests to another bucket or domain. [Learn more](#)'. A callout box provides information about public access: 'For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see [Using Amazon S3 Block Public Access](#)'. Below this, there are fields for 'Index document' (set to 'index.html') and 'Error document - optional' (set to 'error.html').

3. Upload HTML Files

- Create index.html:

```
```html
<!DOCTYPE html>
<html>
<head>
<title>Assignment Website</title>
</head>
<body>
<h1>Hello from S3!</h1>
<p>This is a static website hosted on Amazon S3</p>
</body>
</html>
````
```

- Upload to bucket

The screenshot shows the AWS S3 console with a green success message at the top: "Upload succeeded. For more information, see the Files and folders table." Below it, a summary table shows one succeeded file (index.html) and zero failed files. The "Files and folders" tab is selected, displaying a table with one row for index.html. The table columns include Name, Folder, Type, Size, Status, and Error. The "index.html" row is highlighted with a yellow border.

| Name | Folder | Type | Size | Status | Error |
|------------|--------|-----------|---------|-----------|-------|
| index.html | - | text/html | 206.0 B | Succeeded | - |

4. Test Public URL

- Access via: <https://demo234-4.s3.ap-south-1.amazonaws.com/index.html>

The screenshot shows a browser window with the URL <https://demo234-4.s3.ap-south-1.amazonaws.com/index.html>. The page displays the content "Hello from S3!" and the footer "This is a static website hosted on Amazon S3".

8. Create an Elastic IP and attach it to an EC2 instance with Observe behavior after stopping/starting the instance and validate EIP persistence.

Create Elastic IP

1. EC2 Console

- Elastic IPs under Network & Security
- Click "Allocate Elastic IP address"

2. Allocate EIP

- Amazon's pool of IPv4 addresses
- Click "Allocate"

Associate with EC2 Instance

1. Associate EIP

- Select allocated EIP
- Actions → Associate Elastic IP address
- Choose instance and private IP

2. Verify Association

- Check instance details
- Public IP should show EIP

Test Persistence

1. Stop Instance

- Instance retains EIP association

2. Start Instance

- Public IP remains the same EIP

3. Disassociate EIP

- Actions → Disassociate Elastic IP address
- Observe IP change after restart

9. Create an RDS MySQL instance and connect using MySQL Workbench and Configure security groups, test connectivity, create a database, and run basic SQL queries.

Create RDS Instance

1. RDS Console

- Click "Create database"
- Choose "Standard create"

2. Engine Options

- Engine type: MySQL
- Version: Latest minor version

3. Templates

- Free tier

4. Settings

- DB instance identifier: assignment-db
- Master username: admin
- Master password: [secure-password]

5. Instance Configuration

- DB instance class: db.t3.micro

6. Storage

- Default settings (20GB)

7. Connectivity

- Public access: Yes
- VPC security group: Create new
- Additional configuration: Initial database name: assignmentdb

Configure Security Group

1. Add Inbound Rule

- Type: MySQL/Aurora
- Port: 3306
- Source: My IP

Connect with MySQL Workbench

1. Setup Connection

- Hostname: [RDS endpoint]
- Port: 3306
- Username: admin
- SSL: Require

2. Test Connection

Database Operations

```sql

-- Create table

CREATE TABLE users (

  id INT AUTO\_INCREMENT PRIMARY KEY,  
  name VARCHAR(100),  
  email VARCHAR(100)

);

-- Insert data

INSERT INTO users (name, email) VALUES  
(‘John Doe’, ‘john@example.com’),  
(‘Jane Smith’, ‘jane@example.com’);

-- Query data

SELECT FROM users;

-- Update data

UPDATE users SET email = ‘john.doe@example.com’ WHERE id = 1;

-- Delete data

DELETE FROM users WHERE id = 2;

```

10. Create an EBS volume, attach to an EC2 instance, format, mount, store data, and detach safely.

Create EBS Volume

1. EC2 Console → Volumes

- Click "Create volume"
- Size: 1 GB
- Availability Zone: Same as EC2 instance
- Volume type: gp3

Attach to EC2 Instance

1. Select Volume

- Actions → Attach volume
- Choose instance
- Device name: /dev/sdf

Format and Mount

```bash

List available disks

lsblk

Check if volume has file system

sudo file -s /dev/xvdf

Format volume (if no file system)

sudo mkfs -t xfs /dev/xvdf

Create mount point

sudo mkdir /mnt/assignment-volume

Mount volume

sudo mount /dev/xvdf /mnt/assignment-volume

Verify mount

df -h

```

Store Data

```bash

Create test file

sudo touch /mnt/assignment-volume/test-file.txt

echo "Hello from EBS volume" | sudo tee /mnt/assignment-volume/test-file.txt

Verify file

sudo cat /mnt/assignment-volume/test-file.txt

```

Detach Safely

```bash

Unmount volume

sudo umount /mnt/assignment-volume

Detach from AWS console

EC2 Console → Volumes → Actions → Detach volume

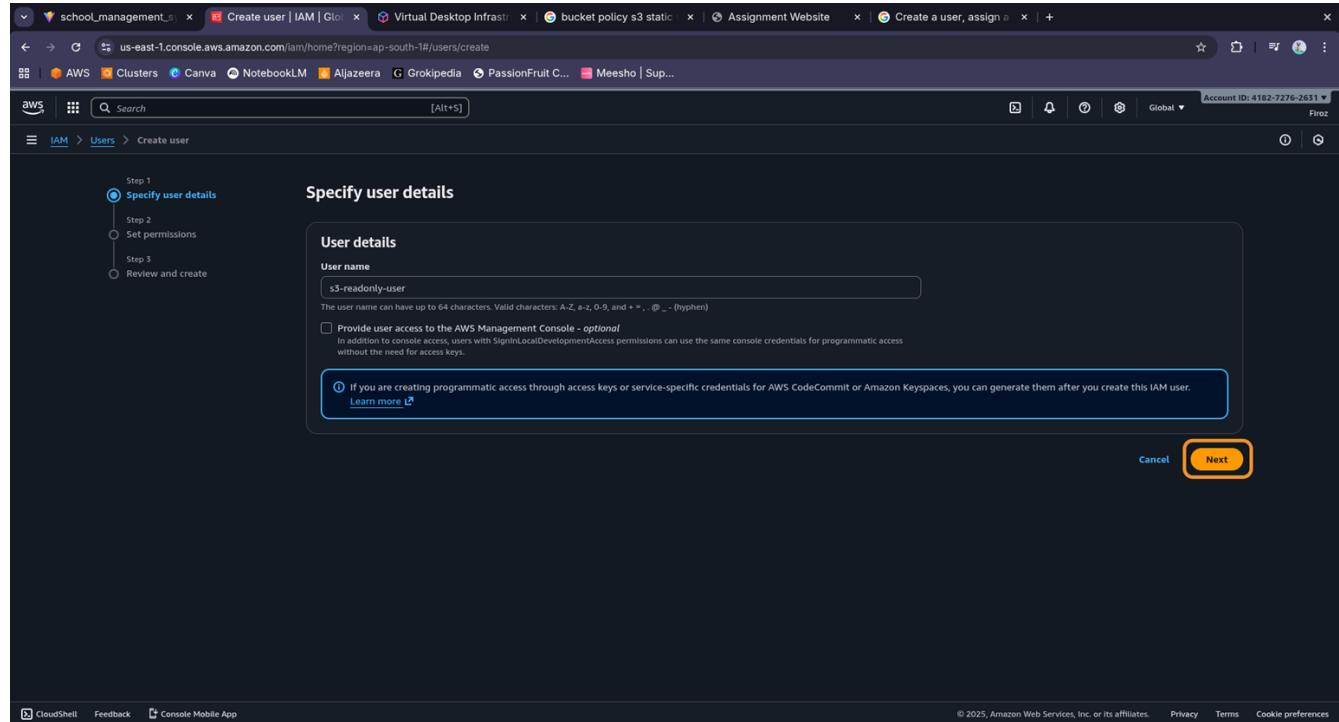
```

11. Create a user, assign a custom policy (S3-read only), generate access keys, and test via AWS CLI.

Create IAM User

1. IAM Console

- Users → Add users
- Username: s3-readonly-user
- Access type: Programmatic access



Assign Custom Policy

1. Attach Policy to User

- Attach existing policies directly
- Select created policy

The screenshot shows the AWS IAM 'Create user' wizard at Step 2: Set permissions. The 'Attach policies directly' option is selected. A search bar at the top shows 's3'. Below it, a table lists 17 matches for AWS managed policies. The 'AmazonS3FullAccess' policy is highlighted with a blue border.

Generate Access Keys

1. Security Credentials

- Save Access Key ID and Secret Access Key

The screenshot shows the AWS IAM 'Create access key' wizard at Step 1: Access key best practices & alternatives. The 'Command Line Interface (CLI)' option is selected. A yellow box highlights the 'Alternatives recommended' section, which lists 'aws login' and 'aws CloudShell'.

Test via AWS CLI

```
```bash
```

### Configure AWS CLI

```
aws configure
```

```
Enter Access Key, Secret Key, Region: us-east-1
```

### Test S3 access

```
aws s3 ls s3://assignment-bucket-unique-name
```

```
aws s3 cp s3://assignment-bucket-unique-name/test-file.txt .
```

Test write permission (should fail)

```
aws s3 cp test-file.txt s3://assignment-bucket-unique-name/
```
```

12. Configure route tables, attach an Internet Gateway, launch EC2 instances, and verify accessibility.

Create VPC

1. VPC Console

- Create VPC
- Name: assignment-vpc
- CIDR: 10.0.0.0/16

Create Internet Gateway

1. Internet Gateways

- Create internet gateway
- Name: assignment-igw
- Attach to VPC

Create Subnet

1. Subnets

- Create subnet
- VPC: assignment-vpc
- CIDR: 10.0.1.0/24
- Enable auto-assign public IP

Configure Route Table

1. Route Tables

- Select main route table
- Edit routes
- Add: 0.0.0.0/0 → Internet Gateway

Launch EC2 in VPC

1. Launch Instance

- Network: assignment-vpc
- Subnet: created subnet
- Auto-assign Public IP: Enable

Verify Accessibility

```bash

SSH to instance

ssh -i key.pem ec2-user@<public-ip>

Test internet connectivity

ping google.com

curl http://example.com

```


13. Deploy two EC2 web servers, attach them to an ELB, and verify automatic traffic distribution.

Create Load Balancer

1. EC2 Console → Load Balancers

- Create Application Load Balancer
- Name: assignment-alb

2. Configure

- Scheme: Internet-facing
- IP address type: IPv4

3. Listeners

- HTTP : 80

4. Availability Zones

- Select at least 2 AZs

Create Target Group

1. Target Groups

- Create target group
- Name: assignment-tg
- Protocol: HTTP
- Health check path: /

Launch Two Web Servers

1. Launch Two Instances

- User data script:
```bash  
#!/bin/bash  
yum update -y  
yum install -y httpd  
systemctl start httpd  
systemctl enable httpd  
echo "<h1>Server \$(hostname)</h1>" > /var/www/html/index.html  
```

2. Register with Target Group

- Add both instances to target group

Test Load Distribution

1. Access ALB DNS

- Refresh multiple times
- Observe different server responses

14. Enable CloudWatch Logs for an EC2 instance and visualize system logs in the console.

Install CloudWatch Agent

1. Create IAM Role

- Attach CloudWatchAgentServerPolicy

2. Attach Role to EC2

- Instance → Actions → Security → Modify IAM role

3. Install Agent

```
```bash
sudo yum install -y amazon-cloudwatch-agent
```
```

Configure Logs

1. Create Config File

```
```json
{
 "logs": {
 "logs_collected": {
 "files": {
 "collect_list": [
 {
 "file_path": "/var/log/messages",
 "log_group_name": "/aws/ec2/assignment-instance",
 "log_stream_name": "system-messages"
 }
]
 }
 }
 }
}
```
```

2. Start Agent

```
```bash
sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -s -c file:config.json
```
```

View Logs in Console

1. CloudWatch Console

- Logs → Log groups
- Select "/aws/ec2/assignment-instance"

- View real-time logs

15. Application Deployment & Scaling

Upload Application

1. Create PHP/Python App

```
```php
<!-- index.php -->
<?php
echo "<h1>Welcome to Assignment App</h1>";
echo "<p>Server: " . gethostname() . "</p>";
echo "<p>Timestamp: " . date('Y-m-d H:i:s') . "</p>";
?>
````
```

2. Upload to Web Server

- Place in /var/www/html/

Configure Environment

1. Install Web Server

```
```bash
For PHP
sudo yum install -y httpd php
sudo systemctl start httpd
sudo systemctl enable httpd
````
```

Auto Scaling Setup

1. Create Launch Template

- Include user data for automatic setup

2. Create Auto Scaling Group

- Desired capacity: 2
- Minimum: 1
- Maximum: 4

Test Deployment

1. Access Application

- Use instance public IP or load balancer DNS

2. Test Scaling

- Configure scaling policy based on CPU
- Generate load to trigger scaling