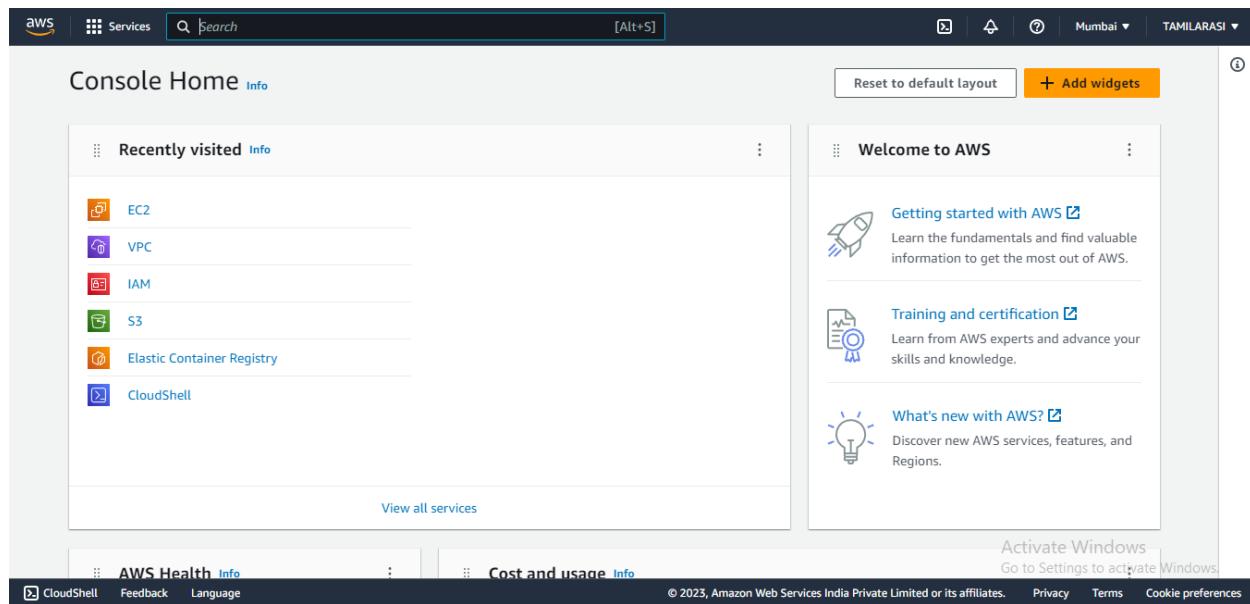


**NAME :TAMILARASI  
REG NO:727721EUCS163**

## **AMAZON CLOUD COMPUTING**

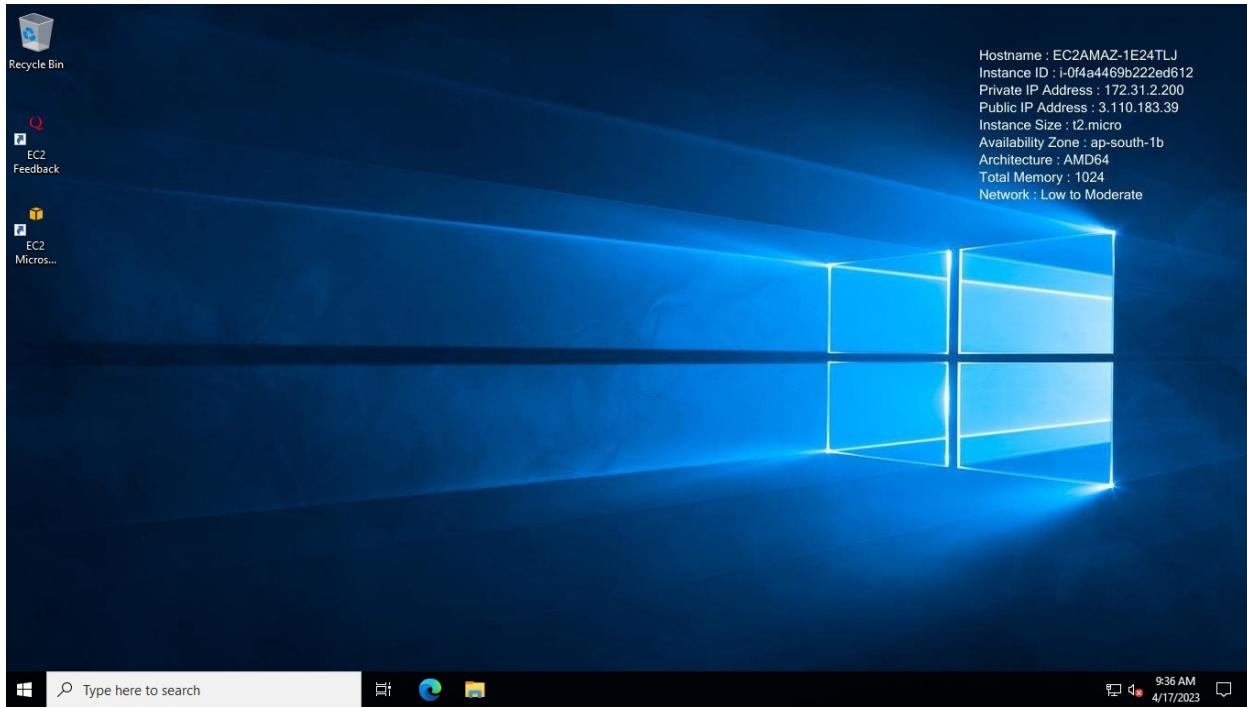
### **Day1:**

#### **1.Aws account creation:**



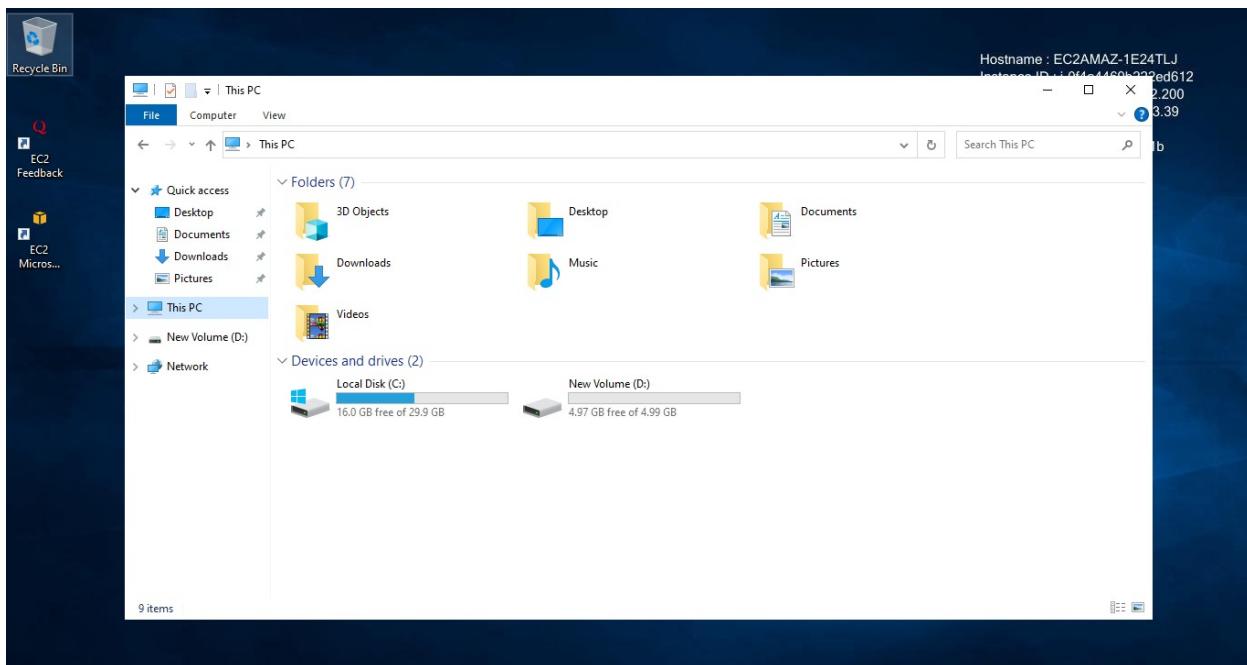
### **Day2:**

Create a Windows EC2 instance with t2.micro Instance and show the remote connection of that EC2 Instance.

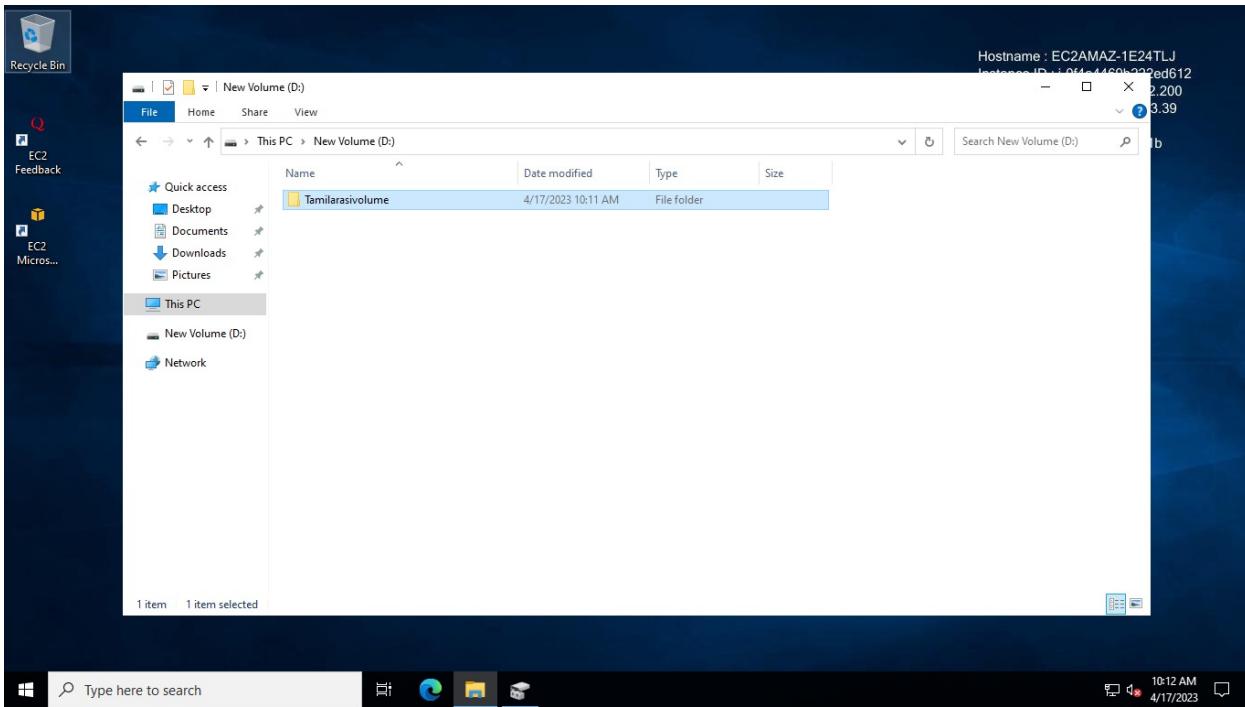


2.

Create an EBS volume of 5 GB and attach to a windows EC2 instance and make partition of that EBS volume.



3.Create some files and folders into 5 GB EBS volume of the previous exercise and take a snapshot of that EBS volume.

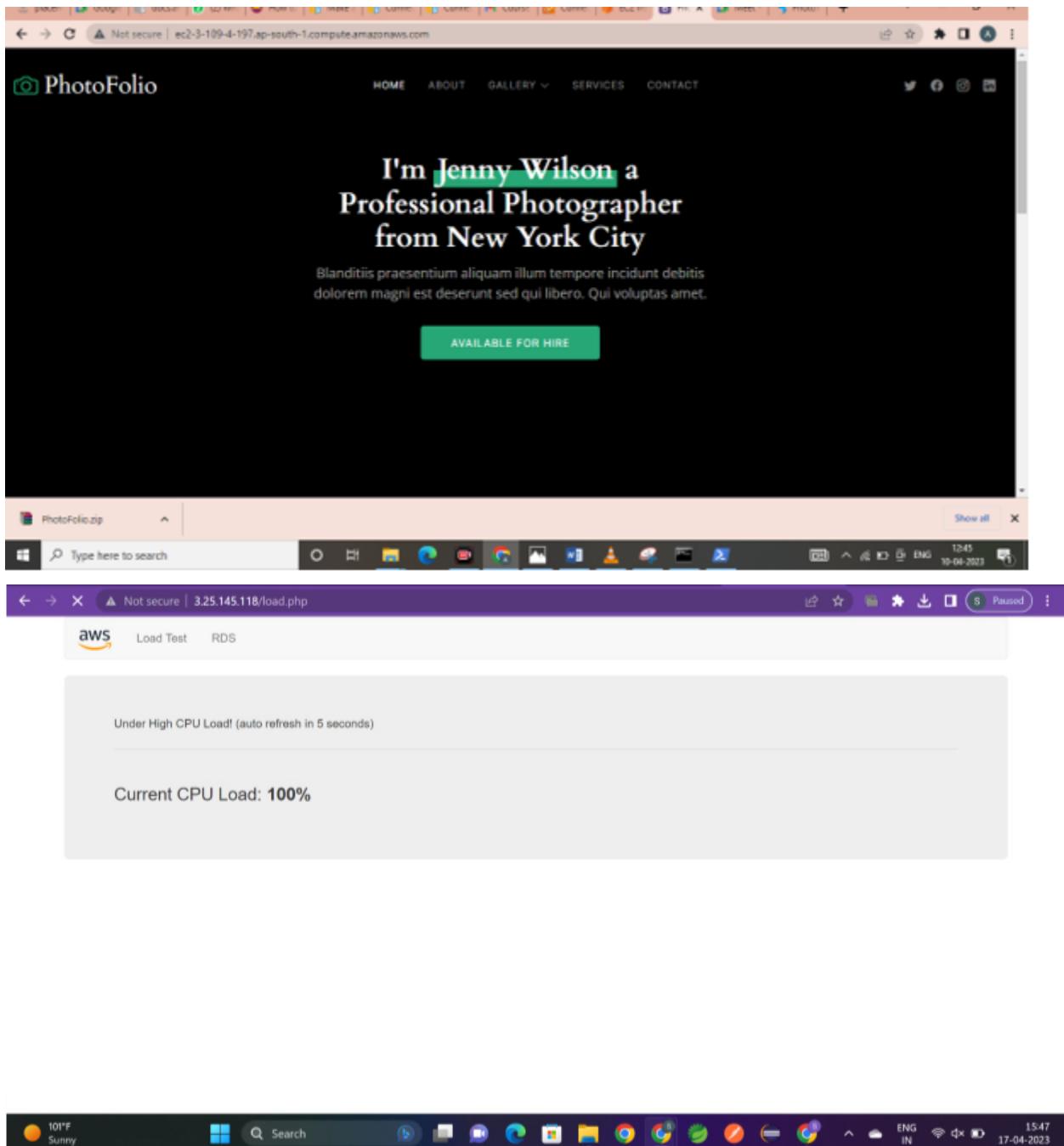


4.

Create a Linux EC2 instance with t2.micro Instance and show the remote connection of that EC2 Instance.



5. Install, Start and Enable the httpd webservice in that Linux EC2 Instance, then host a static website in EC2.



6.

Create Image(MyAMI) of the linux Webserver(from the previous exercise) and launch new EC2 instance from the created Image(MyAMI)

The screenshot shows the AWS EC2 Instances page. A green banner at the top indicates that an AMI is currently being created from instance i-02da9ffa06ead1b6e. Below this, a table lists one instance: 'webstatic' (Instance ID: i-02da9ffa06ead1b6e, State: Running, Type: t2.micro). The sidebar on the left includes sections for Instances, Images, and Elastic Block Store.

## Day3

1.Create a S3 Bucket and create a folder in the bucket and upload a file in the folder.

The screenshot shows the AWS S3 Buckets page. A bucket named 'tamilbucket19' is selected, which contains one object named 'forms.html'. The object is listed with details: Name: forms.html, Type: html, Last modified: April 6, 2023, 11:29:46 (UTC+05:30), Size: 1.1 KB, Storage class: Standard. The sidebar on the left includes sections for Buckets, Storage Lens, and AWS Marketplace for S3.

2.Disable "Block Public Access" for the bucket and enable public read access for a file.

The screenshot shows the AWS S3 console for a bucket named "tamilbucket19". The left sidebar has "Amazon S3" selected under "Buckets". The main tab is "Permissions". A callout box highlights the message: "This bucket has the bucket owner enforced setting applied for Object Ownership". The "Grantee" table lists three entries:

Grantee	Object	Object ACL
Object owner (your AWS account) Canonical ID: c1a00edf579403d6c7137da979ce7b13858a1d6732894e7b0d5d4f0744f25086	Read	Read, Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	-	-
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	-	-

At the bottom, there are links for "Feedback", "Language", "© 2023, Amazon Web Services India Private Limited or its affiliates.", "Privacy", "Terms", and "Cookie preferences".

### 3.Create a bucket policy which should deny to read objects under a folder of a bucket.

The screenshot shows the AWS S3 console for the same bucket "tamilbucket19". The left sidebar has "Amazon S3" selected under "Buckets". The main tab is "Permissions". The "Permissions overview" section shows "Access: Only authorized users of this account". The "Block public access (bucket settings)" section shows "Block all public access: On". There is also a link to "Individual Block Public Access settings for this bucket".

### 4.Enable versioning objects for a bucket and upload objects with multiple versions of it.

The screenshot shows the AWS S3 console. On the left, there's a sidebar with options like Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings for this account, Storage Lens, Dashboards, and AWS Organizations settings. A 'Feature spotlight' section is also present. The main area is titled 'tamilbucket19' with an 'Info' link. Below it, tabs for Objects, Properties (which is selected), Permissions, Metrics, Management, and Access Points are visible. A prominent green banner at the top says 'Successfully edited Bucket Versioning' with the sub-instruction 'To transition, archive, or delete older object versions, configure lifecycle rules for this bucket.' Below the banner, the 'Bucket overview' section displays the AWS Region as 'Asia Pacific (Mumbai) ap-south-1', the Amazon Resource Name (ARN) as 'arn:aws:s3:::tamilbucket19', and the Creation date as 'April 6, 2023, 11:24:51 (UTC+05:30)'. The 'Bucket Versioning' section contains a note about versioning and a 'Edit' button. At the bottom right of the main content area, there's an 'Activate Windows' message with a link to 'Go to Settings to activate Windows.'

5.Host a static webpage in a bucket itself by using static website hosting feature of it.

The screenshot shows a web browser window with the URL 'https://sownd-bucket.s3.ap-south-1.amazonaws.com/CSS1st.html?response-content-disposition=inline&X-Amz-Security-Token=IQoJb3JpZ2luX2VjEHOaCmFwLXNvdXlloL...' in the address bar. The page content is titled 'Smooth Scroll' and includes a section titled 'Section 1' with the text 'Click on the link to see the "smooth" scrolling effect.' and 'Click Me to Smooth Scroll to Section 2 Below'. It also has a note 'Note: Remove the scroll-behavior property to remove smooth scrolling.' At the bottom right of the page, there's an 'Activate Windows' message with a link to 'Go to PC settings to activate Windows.'

6.Enable a lifecycle management rule between various storage classes for a S3 bucket.

The screenshot shows the AWS S3 Management console. On the left, there's a sidebar with options like Buckets, Access Points, Storage Lens, and Feature spotlight. The main area shows the 'Management' tab selected under 'Lifecycle rules (1)'. A table lists one rule named 'ruletamil' with status 'Enabled' and scope 'Filtered'. The rule specifies transitioning objects to 'Standard-IA'. There are buttons for Actions (View details, Edit, Delete, Create lifecycle rule) and a link to 'View lifecycle configuration'.

## Day:4

1.Create an IAM group called as 'S3-Admins' with 'AmazonS3FullAccess'.

The screenshot shows the AWS IAM Management console. The left sidebar includes 'User groups' under 'Access management'. In the main area, a success message 'S3-Admins user group created.' is displayed. Below it, the 'User groups (1)' section shows a table with one entry: 'S3-Admins' (Group name), 'Defined' (Permissions), and 'Now' (Creation time). There are buttons for 'View group' and 'Create group'.

2.Create an IAM user called as 'S3Admin1' and add it to the 'S3-Admins' group.

The screenshot shows the AWS Management Console with the IAM service selected. A green banner at the top indicates that a user has been created successfully. Below the banner, the 'Create user' wizard is shown, currently on Step 4: 'Retrieve password'. The 'Console sign-in details' section displays the sign-in URL (https://015066583083.siginin.aws.amazon.com/console), the user name (S3-Admin1), and the console password (represented by a series of asterisks). A 'Show' link is available to view the full password.

3. Attach an IAM custom policy to the 'S3-Admins' group which should deny to delete objects.

The screenshot shows the AWS Management Console with the IAM service selected. A green banner at the top indicates that a policy named 'Tamilpolicy' has been created. Below the banner, the 'Policies' page is displayed, showing a list of existing policies. The newly created 'Tamilpolicy' is listed as a Customer managed policy. Other policies shown include AWS managed policies like 'AWSMarketplaceFullAccess' and 'ClientVPNServiceRolePolicy', and other customer managed policies like 'AWSDirectConnectReadOnlyAccess' and 'AmazonGlacierReadOnlyAccess'.

Policy name	Type	Used as	Description
Tamilpolicy	Customer managed	None	
AWSMarketplaceFullAccess	AWS managed	None	Provides the ability to access the AWS Marketplace
ClientVPNServiceRolePolicy	AWS managed	None	Policy to enable AWS Client VPN access
AWSGlacierReadOnlyAccess	AWS managed	None	Provides read-only access to Amazon Glacier
AWSMarketplaceFullAccess	AWS managed	None	Provides the ability to access the AWS Marketplace
AWSDirectConnectReadOnlyAccess	AWS managed	None	Provides read-only access to AWS Direct Connect

4.

Create an Inline policy for an IAM user and set some permission boundary for that user.

The screenshot shows the AWS Identity and Access Management (IAM) service. In the left sidebar, under 'Access management', 'Users' is selected. The main content area displays 'Permissions policies (2)' attached to a user. One policy is AWS managed ('IAMUserChangePassword'), and the other is Customer inline ('mypolicy'). A section for 'Permissions boundary (not set)' is present, along with a note about delegating permission management. At the bottom, there's a link to 'Generate policy based on CloudTrail events'.

## 5.Create an IAM role with 'AmazonS3FullAccess' and attach the role to an EC2 instance.

The screenshot shows the AWS IAM service. In the left sidebar, under 'Access management', 'Roles' is selected. The main content area displays 'Roles (3) Info', indicating that an IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust. The table lists three roles: 'AWSServiceRoleForSupport', 'AWSServiceRoleForTrustedAdvisor', and 'Tamilrole'. Below the table, there's a section for 'Roles Anywhere' with a 'Manage' button.

## 6.Activate MFA for an IAM user and Set some Password Policies such as 1 uppercase, 1 lowercase etc

**Multi-factor authentication (MFA) (1)**

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. [Learn more](#)

**Device type** Identifier Created on

Virtual arn:aws:iam::226327626315:mfa/vaish 1 minute ago

**Create access key**

**No access keys**

As a best practice, avoid using long-term credentials like access keys. Instead, use tools which provide short term credentials. [Learn more](#)

<https://us-east-1.console.aws.amazon.com/iam/home#/home>

## Day 5:

1.Create a launch template with a custom AMI and t2.micro instance type

**Success**  
Successfully initiated launch of instance (i-08a938c26628b2b61)

[Launch log](#)

**Next Steps**

Q. What would you like to do next with this instance, for example "create alarm" or "create backup"

1 2 3 4 5 6

Create billing and free tier usage alerts  To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage.	Connect to your instance  Once your instance is running, log into it from your local computer. <a href="#">Connect to instance</a>	Connect an RDS database  Configure the connection between an EC2 instance and a database to allow traffic flow between them.	Create EBS snapshot policy  Create a policy that automates the creation, retention, and deletion of EBS snapshots.
--	---	--	--

[CloudShell](#) [Feedback](#) [Language](#)

2.Create an autoscaling group with the above-created launch template

## Day:6

Qst:1

Create a vpc with multiple subnets(atleast 1 subnet in each zone)

The screenshot shows the 'Create VPC workflow' page in the AWS Management Console. At the top, there's a navigation bar with 'VPC' selected. Below it, the breadcrumb trail reads: 'VPC > Your VPCs > Create VPC > Create VPC resources'. The main area displays a 'Success' message with a green checkmark icon. A 'Details' section lists 14 successful steps, each with a green checkmark and a blue link icon. The steps include creating the VPC, disabling DNS hostnames and resolution, verifying VPC creation, creating an S3 endpoint, and creating three subnets. On the right side of the page, there's a promotional message for activating Windows.

Success

Details

- Create VPC: vpc-0468260030cf613a7
- Disable DNS hostnames
- Disable DNS resolution
- Verifying VPC creation: vpc-0468260030cf613a7
- Create S3 endpoint: vpce-0ceb4a20618b01805
- Create subnet: subnet-099eb3e89ddd418db
- Create subnet: subnet-037b092349aa74d8e
- Create subnet: subnet-06295a5046f134a5f
- Create subnet: subnet-0aa0d6c54f0a36778
- Create internet gateway: igw-016bf1d143a9fc03
- Attach internet gateway to the VPC
- Create route table: rtb-08351608d438c6894

Activate Windows  
Go to Settings to activate Windows.

Qst:2

Make 1 public subnet and 2 private subnets in the created VPC

The screenshot shows the 'Subnets (4)' list in the AWS Management Console. The left sidebar has 'Your VPCs' selected under 'Virtual private cloud'. The main table lists four subnets with columns for Name, Subnet ID, State, VPC, and IPv4 CIDR. Three subnets are public (available) and one is private (available). The public subnet has an IPv4 CIDR of 182.0.0.0/20, and the private subnets have CIDRs of 182.0.16.0/20 and 182.0.144.0/20 respectively. A search bar at the top is set to 'search: vpc-035d2c3c794106a62'. A modal window titled 'Select a subnet' is open at the bottom.

Name	Subnet ID	State	VPC	IPv4 CIDR
uthika-subnet-publ...	subnet-09ea4a8c27ee81bd8	Available	vpc-035d2c3c794106a62   uth...	182.0.0.0/20
uthika-subnet-publ...	subnet-0d577073501372e01	Available	vpc-035d2c3c794106a62   uth...	182.0.16.0/20
uthika-subnet-driva...	subnet-02da8b2012cd57c7c	Available	vpc-035d2c3c794106a62   uth...	182.0.144.0/20

Qst3

Make internet connection using NAT gateway for the 2 private subnets.

VPC > NAT gateways > Create NAT gateway

### Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

**NAT gateway settings**

Name - optional  
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Subnet  
Select a subnet in which to create the NAT gateway.

Connectivity type  
Select a connectivity type for the NAT gateway.  
 Public  
 Private

Elastic IP allocation ID Info  
Assign an Elastic IP address to the NAT gateway.

Activate Windows  
Go to PC settings to activate Windows

Qst:4

Create a VPC peering connection between 2 different VPCs from 2 different regions.

The screenshot shows the AWS VPC Peering Connections page. On the left, there's a sidebar for 'Virtual private cloud' with options like 'Your VPCs', 'Subnets', 'Route tables', etc. The main area displays a table of peering connections:

Name	Peering connection ID	Status	Requester VPC	Acceptor VPC
tamilpeering	pcx-031f540188ebc74a2	Active	vpc-000f3a62a6ac87779 / ta...	vpc-035d2c3c794106a62 / i...
-	pcx-08e84ab82aae60399	Active	vpc-02b875482f536539a	vpc-000f3a62a6ac87779 / t...

Below the table, there are tabs for 'Details', 'DNS', 'Route tables', and 'Tags'. The 'Details' tab is selected, showing fields for Requester owner ID (015066583083), Acceptor owner ID (015066583083), and VPC Peering connection ARN (arn:aws:ec2:ca-central-1:015066583083:vpc-peering-connection/pcx-031f540188ebc74a2).

Qst5: Create VPC peering connections for 3 different VPCs from the same region

The screenshot shows the AWS VPC Subnets page. The left sidebar lists various VPC-related services. The main content area displays a table of subnets with the following data:

Name	Subnet ID	State	VPC	IPv4 CIDR
uthika-subnet-public	subnet-09ea4a8c27ee81bd8	Available	vpc-035d2c3c794106a62   uthika-vpc	182.0.0.0/20
uthika-subnet-public	subnet-0d577073501372e01	Available	vpc-035d2c3c794106a62   uthika-vpc	182.0.16.0/20
uthika-subnet-private	subnet-02da8b2012cd57c7c	Available	vpc-035d2c3c794106a62   uthika-vpc	182.0.144.0/20

A search bar at the top is set to "search: vpc-035d2c3c794106a62". A "Create subnet" button is located in the top right corner.

## QSt6

Add security rules in the VPC's NACL which should deny RDP, SSH from the public network

1

The screenshot shows the AWS Network ACL (NACL) details page for the NACL ID "acl-02b77ef4959a0f477". A green success message at the top states: "You have successfully updated inbound rules for acl-02b77ef4959a0f477". The main content area displays the following details:

Network ACL ID	Associated with	Default	VPC ID
acl-02b77ef4959a0f477	4 Subnets	Yes	vpc-035d2c3c794106a62 / uthika-vpc
Owner	015066583083		

Below the details, there are tabs for "Inbound rules", "Outbound rules", "Subnet associations", and "Tags". A message at the bottom encourages users to "Run Reachability Analyzer".