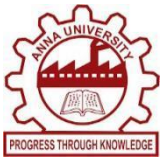
Dr. N.G.P INSTITUTE OF TECHNOLOGY, COIMBATORE - 641048

AN AUTONOMOUS INSTITUTION



Reg. No.	**:**	710722104005
Name	**:**	ANANTHA SUDEESH SD
Class	**:**	II Year CSE A
Git URL	**:**	**https://github.com/ananthasudeesh**
Course Name	**:**	Microsoft azure Fundamentals
Company	**:**	Pinesphere Solution, Coimbatore.
Start Date	**:**	06.08.2024
End Date	**:**	10.08.2024

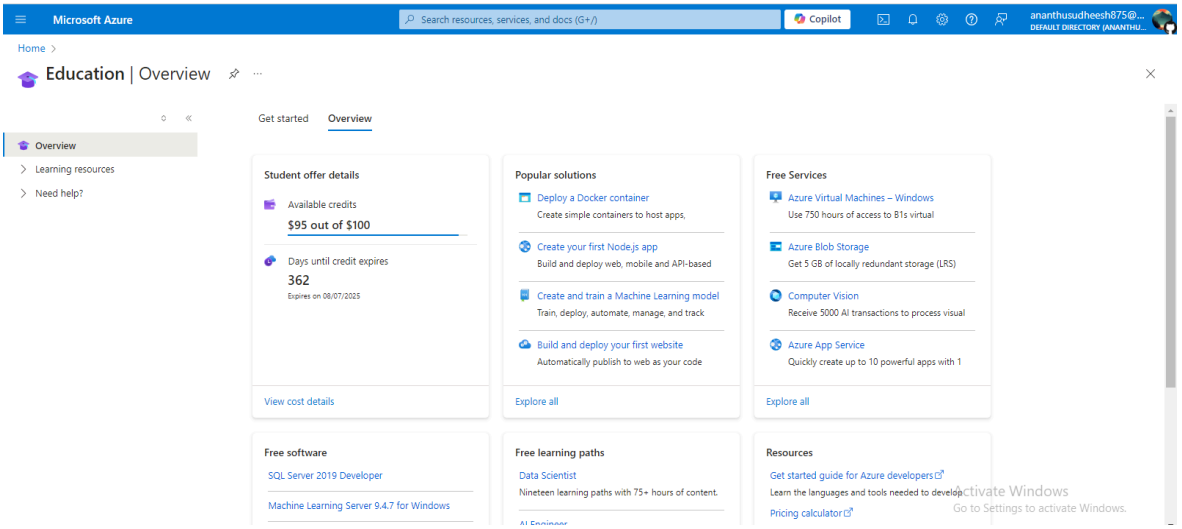
INDEX

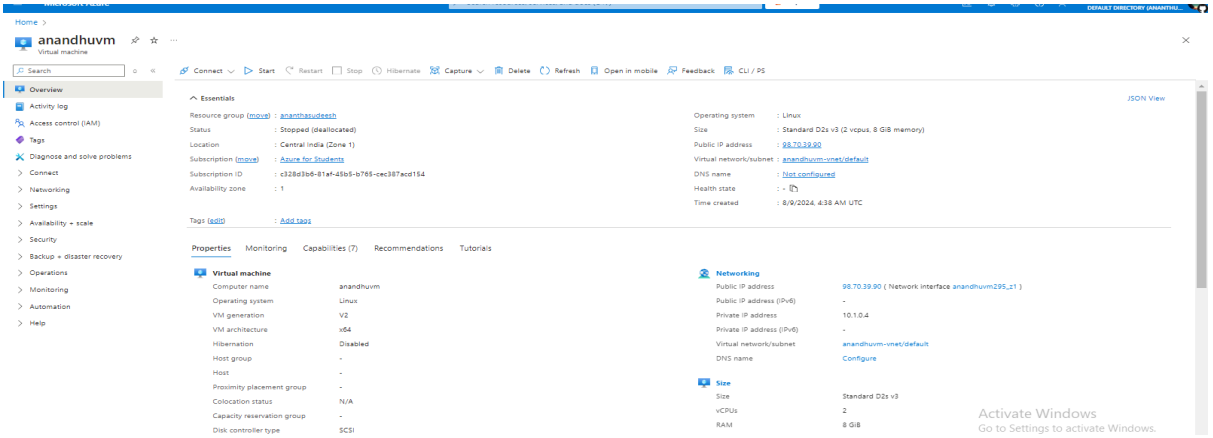
CREATING A VIRTUAL MACHINE (VM) IN MICROSOFT AZURE:

Creating A Virtual Machine (Vm) In Microsoft Azure Involves The Following Steps:

1. Sign in to the Azure portal.
2. Navigate to "Create a resource" and select "Virtual Machine."
3. Choose a subscription, resource group, and region.
4. Configure VM settings, including size, OS, and storage.
5. Set up networking, security, and management options.
6. Review and create the VM, then monitor its deployment.

The VM will be ready to use after deployment.





HOST A WEBSITE FROM GITHUB ON A VIRTUAL MACHINE (VM) IN MICROSOFT AZURE

1. Set Up the VM: Ensure your Azure VM is running and accessible via SSH or RDP. Install a web server like Apache or Nginx on the VM.

2. Clone the GitHub Repository: SSH into the VM and clone your website's repository from GitHub using git clone <repository-url>.

3. Deploy the Website: Move the cloned repository to the web server's root directory, typically /var/www/html for Apache or the appropriate directory for Nginx.

4. Configure the Web Server: Update the web server configuration files to serve your website. Restart the server to apply changes.

5. Open Ports: Ensure that the necessary ports (e.g., port 80 for HTTP) are open in the Azure network security group settings to allow web traffic.

6. Access the Website: Access your website by entering the VM's public IP address or domain name in a web browser.

COMMANDS:

Requesting a Cloud Shell.Succeeded.

Connecting terminal...

Your Cloud Shell session will be ephemeral so no files or system changes will persist beyond your current session.

anantha [~]$ ssh anandhu@4.240.82.27

The authenticity of host '4.240.82.27 (4.240.82.27)' can't be established.

ED25519 key fingerprint is SHA256:U+fwDQJ6SyYBN+Z/JcIKWMEvb2aH7Sp4w7jr82qlk8c.

This key is not known by any other names

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added '4.240.82.27' (ED25519) to the list of known hosts.

anandhu@4.240.82.27's password:

Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-azure x86_64)

*** Documentation: https://help.ubuntu.com**

*** Management: https://landscape.canonical.com**

*** Support: https://ubuntu.com/pro**

System information as of Sat Aug 10 05:39:34 UTC 2024

System load: 0.19 Processes: 146

Usage of /: 5.0% of 28.02GB Users logged in: 0

Memory usage: 1% IPv4 address for eth0: 10.1.0.5

Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.

See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;

the exact distribution terms for each program are described in the

individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by

applicable law.

To run a command as administrator (user "root"), use "sudo <command>".

See "man sudo_root" for details.

anandhu@vm:~$ sudo apt update

Hit:1 http://azure.archive.ubuntu.com/ubuntu noble InRelease

Get:2 http://azure.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]

Get:3 http://azure.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]

Hit:4 http://azure.archive.ubuntu.com/ubuntu noble-security InRelease

Get:5 http://azure.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]

Get:6 http://azure.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]

Get:7 http://azure.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]

Get:8 http://azure.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]

Get:9 http://azure.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]

Get:10 http://azure.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]

Get:11 http://azure.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]

Get:12 http://azure.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]

Get:13 http://azure.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [344 kB]

Get:14 http://azure.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [321 kB]

Get:15 http://azure.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [135 kB]

Get:16 http://azure.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [45.0 kB]

Get:17 http://azure.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [12.7 kB]

Get:18 http://azure.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [14.1 kB]

Get:19 http://azure.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [3608 B]

Get:20 http://azure.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [212 B]

Get:21 http://azure.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [532 B]

Get:22 http://azure.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 B]

Get:23 http://azure.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [112 B]

Get:24 http://azure.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [10.3 kB]

Get:25 http://azure.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [10.5 kB]

Get:26 http://azure.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [17.6 kB]

Get:27 http://azure.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1016 B]

Get:28 http://azure.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]

Get:29 http://azure.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]

Get:30 http://azure.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]

Get:31 http://azure.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]

Get:32 http://azure.archive.ubuntu.com/ubuntu noble-security/universe amd64 Packages [249 kB]

Get:33 http://azure.archive.ubuntu.com/ubuntu noble-security/universe Translation-en [108 kB]

Get:34 http://azure.archive.ubuntu.com/ubuntu noble-security/universe amd64 Components [8632 B]

Get:35 http://azure.archive.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [9376 B]

Get:36 http://azure.archive.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [10.6 kB]

Get:37 http://azure.archive.ubuntu.com/ubuntu noble-security/multiverse Translation-en [2808 B]

Get:38 http://azure.archive.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]

Get:39 http://azure.archive.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [344 B]

Fetched 27.2 MB in 4s (6112 kB/s)

Reading package lists... Done

Building dependency tree... Done

Reading state information... Done

All packages are up to date.

anandhu@vm:~$ sudo apt install git

Reading package lists... Done

Building dependency tree... Done

Reading state information... Done

git is already the newest version (1:2.43.0-1ubuntu7.1).

git set to manually installed.

0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.

anandhu@vm:~$ sudo apt install nginx

Reading package lists... Done

Building dependency tree... Done

Reading state information... Done

The following additional packages will be installed:

nginx-common

Suggested packages:

fcgiwrap nginx-doc ssl-cert

The following NEW packages will be installed:

nginx nginx-common

0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.

Need to get 552 kB of archives.

After this operation, 1596 kB of additional disk space will be used.

Do you want to continue? [Y/n] yes

Get:1 http://azure.archive.ubuntu.com/ubuntu noble/main amd64 nginx-common all 1.24.0-2ubuntu7 [31.2 kB]

Get:2 http://azure.archive.ubuntu.com/ubuntu noble/main amd64 nginx amd64 1.24.0-2ubuntu7 [521 kB]

Fetched 552 kB in 0s (1470 kB/s)

Preconfiguring packages ...

Selecting previously unselected package nginx-common.

(Reading database ... 64525 files and directories currently installed.)

Preparing to unpack .../nginx-common_1.24.0-2ubuntu7_all.deb ...

Unpacking nginx-common (1.24.0-2ubuntu7) ...

Selecting previously unselected package nginx.

Preparing to unpack .../nginx_1.24.0-2ubuntu7_amd64.deb ...

Unpacking nginx (1.24.0-2ubuntu7) ...

Setting up nginx (1.24.0-2ubuntu7) ...

Setting up nginx-common (1.24.0-2ubuntu7) ...

debconf: unable to initialize frontend: Dialog

debconf: (Dialog frontend requires a screen at least 13 lines tall and 31 columns wide.)

debconf: falling back to frontend: Readline

Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.

Processing triggers for ufw (0.36.2-6) ...

Processing triggers for man-db (2.12.0-4build2) ...

Scanning processes...

Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.

anandhu@vm:~$ sudo systemctl start nginx

anandhu@vm:~$ sudo systemctl enable nginx

Synchronizing state of nginx.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.

Executing: /usr/lib/systemd/systemd-sysv-install enable nginx

anandhu@vm:~$ cd /var/www/html

anandhu@vm:/var/www/html$ sudo rm -rf *

anandhu@vm:/var/www/html$ sudo git clone https://github.com/ananthasudeesh/flex.git .

Cloning into '.'...

remote: Enumerating objects: 50, done.

remote: Counting objects: 100% (50/50), done.

remote: Compressing objects: 100% (49/49), done.

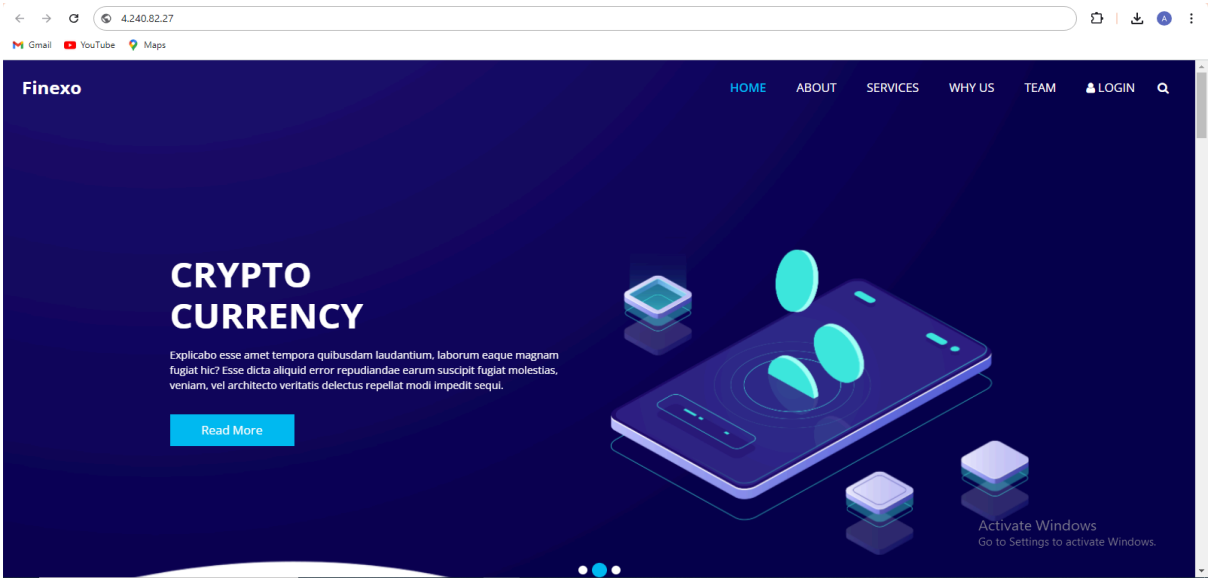
remote: Total 50 (delta 4), reused 0 (delta 0), pack-reused 0

Receiving objects: 100% (50/50), 1.99 MiB | 13.67 MiB/s, done.

Resolving deltas: 100% (4/4), done.

anandhu@vm:/var/www/html$ sudo chown -R www-data:www-data /var/www/html

anandhu@vm:/var/www/html$



CREATION OF STORAGE ACCOUNT IN MICROSOFT:

To Create A Storage Account In Microsoft Azure, Follow These Steps:

1. Sign in to Azure Portal: Log in to the Azure portal at https://portal.azure.com.

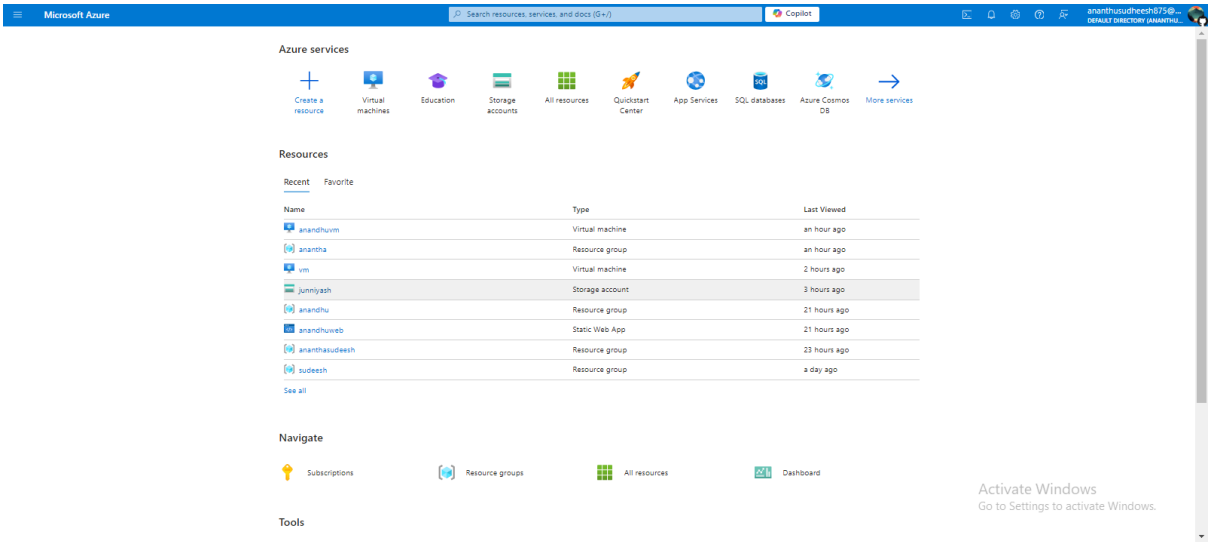
2. Create a Resource: Click on "Create a resource" and select "Storage account" under the "Storage" category.

3. Configure the Basics: Choose a subscription, resource group, and storage account name. Select the region, performance tier (Standard or Premium), and replication option (e.g., LRS, GRS).

4. Set Advanced Options: Configure additional settings like access tier (Hot or Cool), security options, and networking.

5. Review and Create: Review the configuration and click "Create" to deploy the storage account.

6. Access the Storage Account: After deployment, access the storage account to manage containers, blobs, files, tables, or queues.



MANAGING OF STORAGE ACCOUNT

To Upload An Image Into A Container In An Azure Storage Account, Follow These Steps:

a. Access the Storage Account: Sign in to the Azure portal and navigate to your Storage Account.

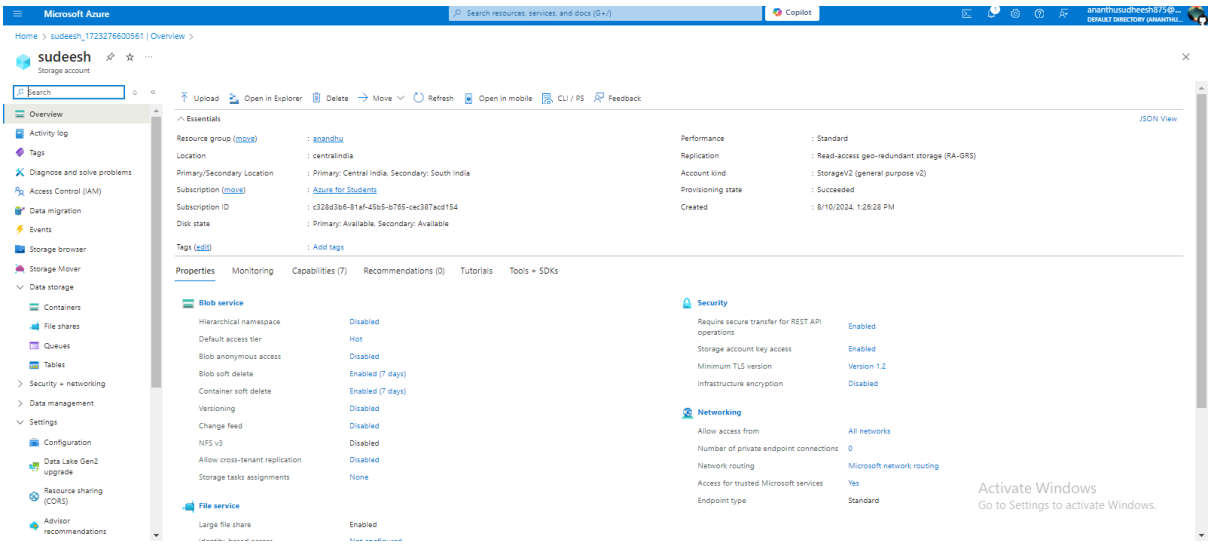
b. Create a Container: In the Storage Account, select "Containers" and click "Add Container." Name the container and set the access level (private, blob, or container).

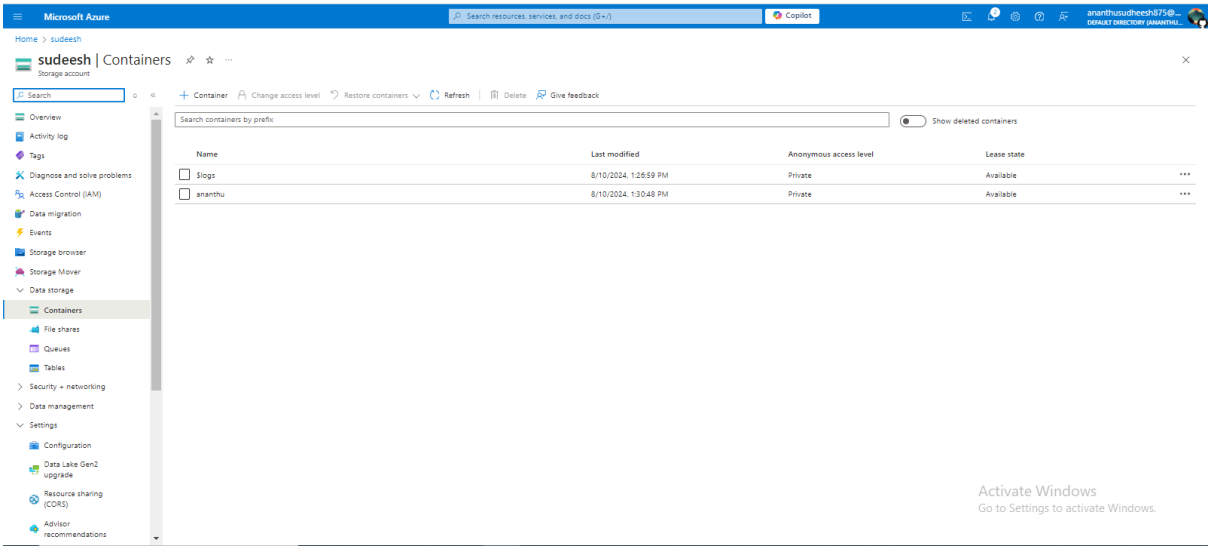
c. Open the Container: Once created, click on the container to open it.

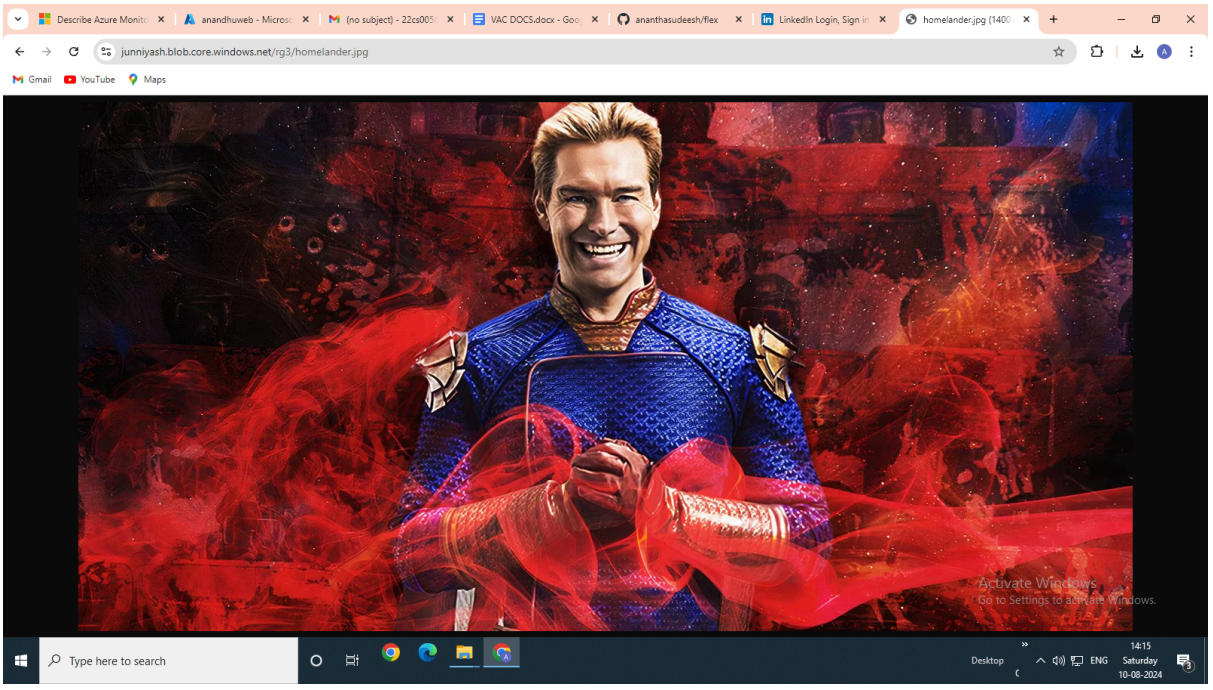
d. Upload the Image: Click the "Upload" button within the container. In the upload window, browse your local machine to select the image file.

e. Configure Upload Settings: Optional - You can set advanced upload options like overwriting existing files, setting metadata, or assigning blob tier.

f. Start the Upload: Click "Upload" to start the process. Once the upload is complete, your image will be stored in the container and accessible based on the access level you set.







URL :https://junniyash.blob.core.windows.net/rg3/homelander.jpg

STATIC WEB PAGE:

Deploying a Static Web Page on Azure

Using Azure Static Web App:

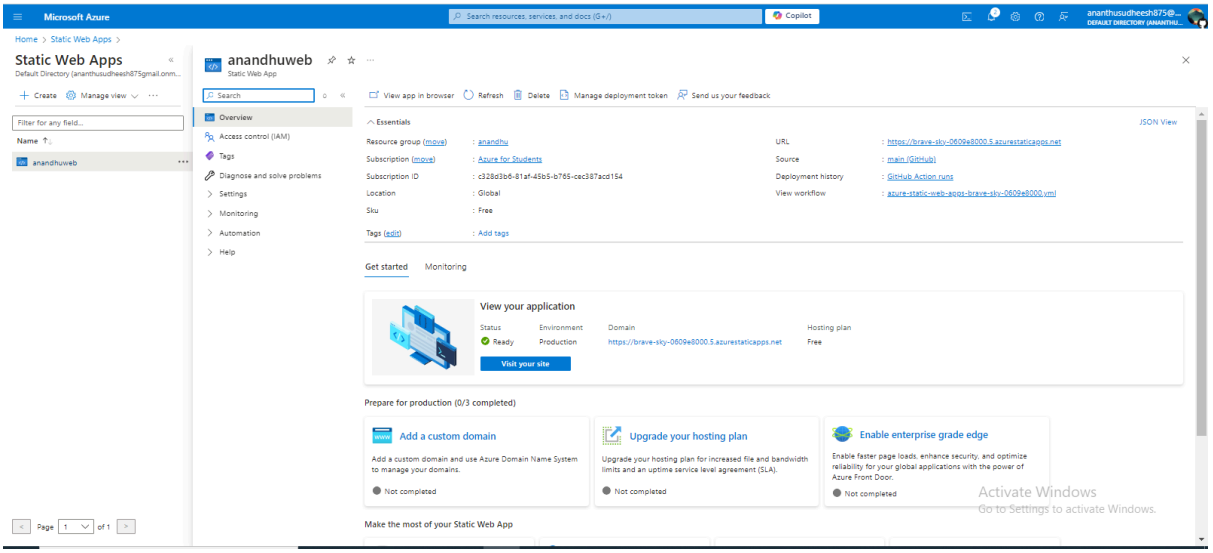
Prepare Your Site: Develop your static site and push it to a GitHub repository.

Set Up Azure Static Web Apps:

1. Sign in to Azure Portal.
2. Click **Create a resource** > **Static Web Apps**.
3. Connect to your GitHub repo and branch.

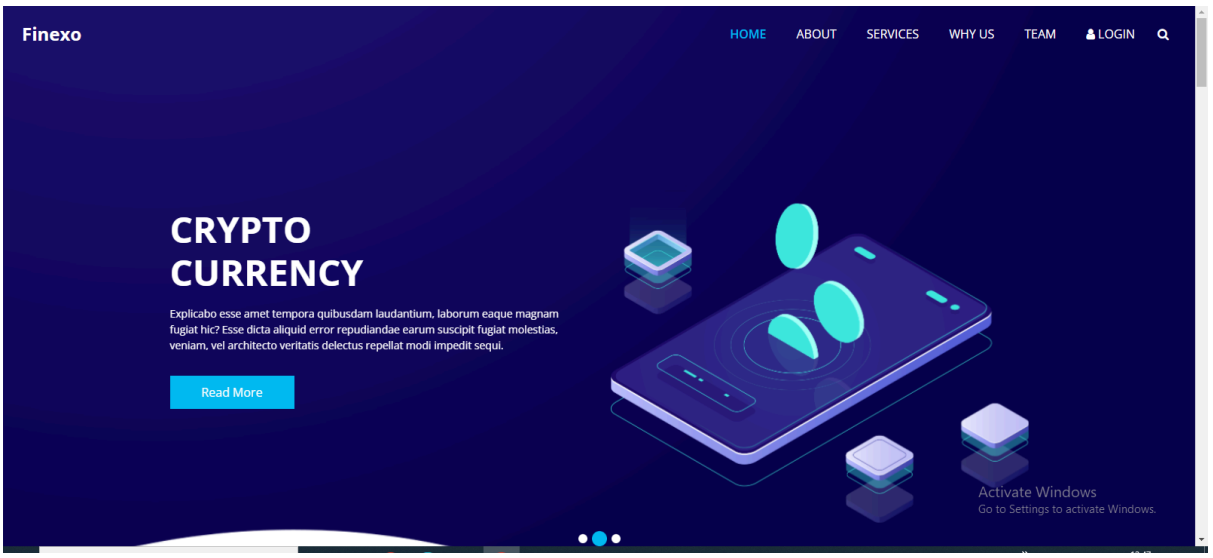
Deploy and Access:

1. Azure deploys your site automatically.
2. Access it via the provided URL.



URL:https://brave-sky-0609e8000.5.azurestaticapps.net

OUTPUT:



CREATION OF LOCK

You can create a lock on a resource by navigating to the resource, selecting Locks under Settings, and then adding a lock with either a Read-only or Delete option to prevent accidental modifications or deletions.