

Azure Experiments

Name: Gaurav Kakade

Index

Sr. no.	Name of Experiment	Page no	Date of experiment
1	To attach- managed, unmanaged & shared disk to VM To take snapshot & encrypt disk	3	07/06/2021
2	To deploy VM in Availability set, Availability zone & proximity placement Group	25	14/06/2021
3	Creation of Web App & management app service log, app service network configuration	39	23/06/2021
4	Create a web app instance and a virtual machine which contains database, and try to access the database through web app instance.	51	12/07/2021
5	Creation of Virtual Machine, installation of Docker Engine, Image Management, Launching container	60	19/07/2021
6	Working with container Instance, Kubernetes cluster, network configuration	70	22/07/2021
7	Azure Networking, Address space, & attaching secondary NIC to Virtual Machine.	77	28/07/2021
8	Creation of VNET & Security rule management of Virtual Machine level and subnet level.	83	03/08/2021
9	Creation of Virtual network using ARM management.	93	06/08/2021
10	Creating load balancer in azure- Basic and Standard.	98	10/08/2021

Experiment No: 1

AIM: To attach managed disk, unmanaged disk, shared disk to virtual machine, to take snapshot of managed disk and to encrypt the data disk.

PREREQUISITES: Azure Portal, RDP.

DESCRIPTION:

Azure managed disk: Azure Managed Disks simplifies disk management for Azure IaaS VMs by managing the storage accounts associated with the VM disks. You only have to specify the type (Premium or Standard) and the size of disk you need, and Azure creates and manages the disk for you.

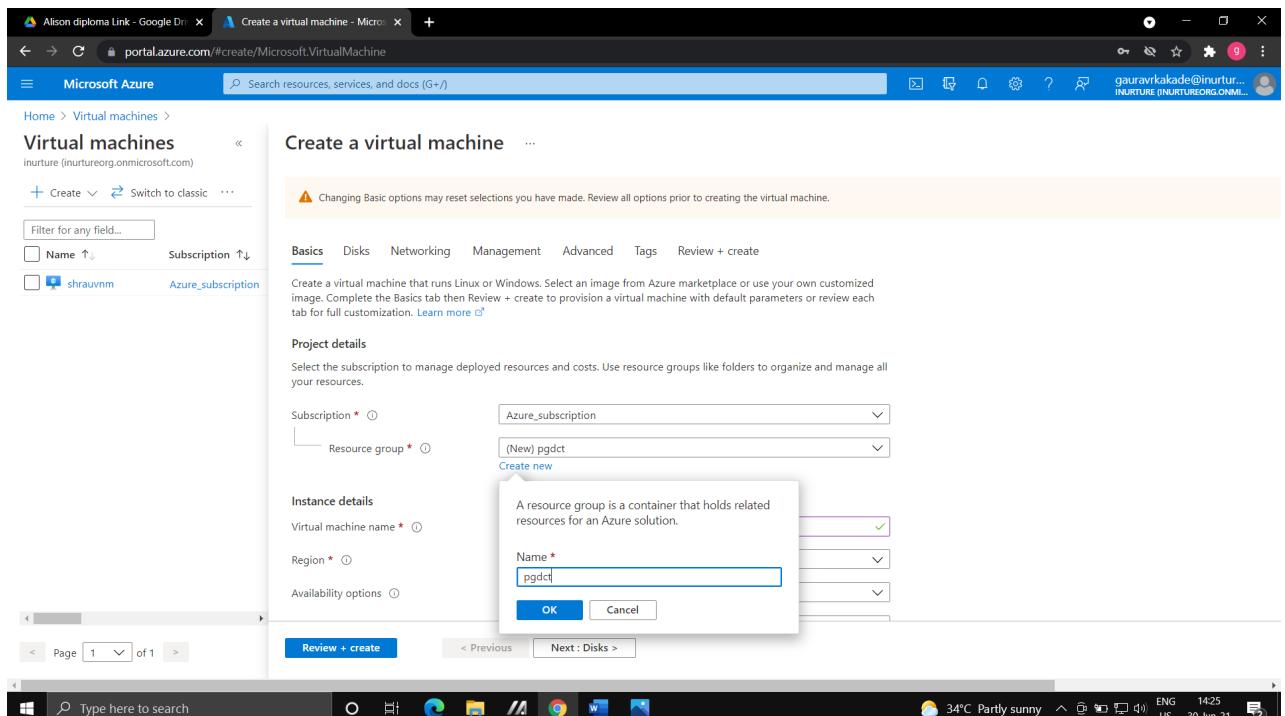
Azure unmanaged disk: Microsoft Azure unmanaged disk is a Microsoft-managed cloud service that provides storage that is highly available, secure, durable, scalable, and redundant. Microsoft takes care of maintenance and handles critical problems for you.

Shared disk: Shared disks is a feature of Azure Disk Storage that allows a single disk to be attached to multiple virtual machines. This enables you to run your most demanding enterprise applications like clustered databases, parallel file systems, persistent containers and machine learning applications in the cloud, without compromising well-known deployment patterns for fast failover and high availability.

ALGORITHM:

Steps to create a virtual machine:

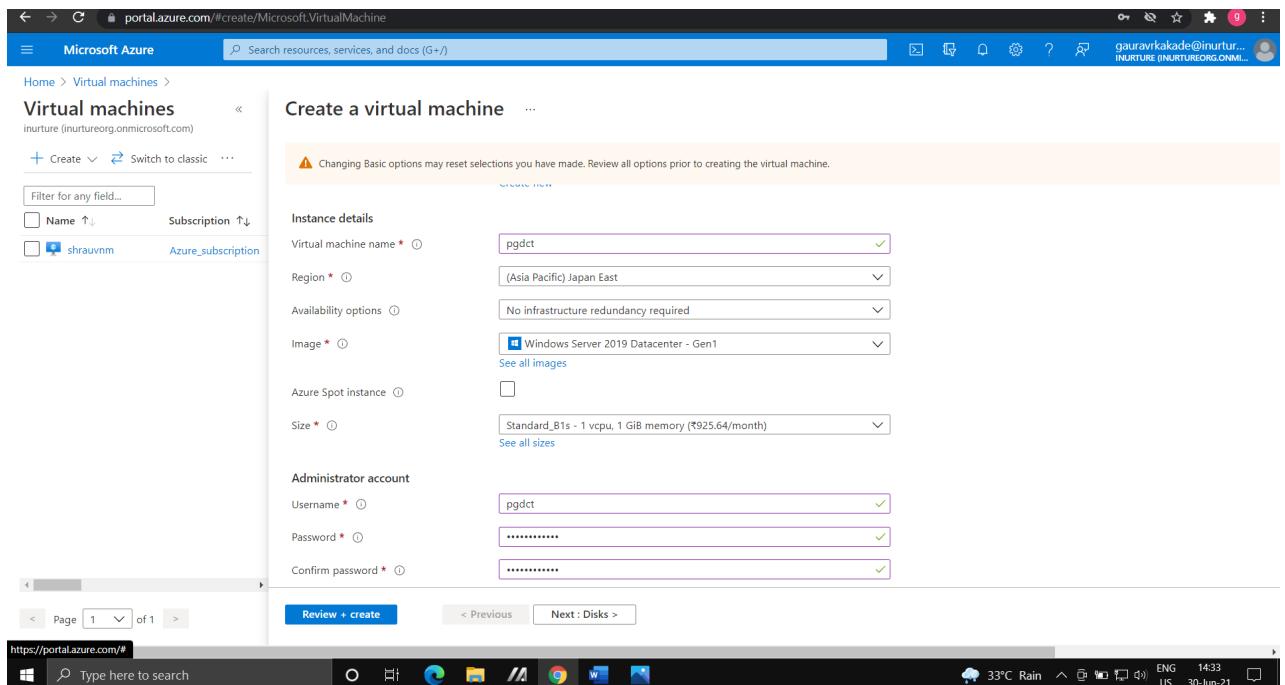
- 1.Click on create virtual machine.
2. a. Create new resource group.
b. Enter name of resource group.
3. a. Enter the Virtual machine name.
b. Select the region.



4. a. Select the image

b. Select the size of virtual machine.

5. Give the username and password for your virtual machine.



6. Select the inbound port rules.

7. Select the OS disk type as Standard HDD.

8. Go with the default settings in disks, networking, management, Advanced and tags.

9. Review and Create.

Steps to attach data/managed disk to virtual machine:

1. Go to the Virtual machine.
2. Select Disk.
3. Click on Create and attach new disk.

4. Give the name of disk, choose type of disk as Standard HDD and choose size then save it.

OS disk

Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (MB/s)	Encryption	Host caching
pgdct_OsDisk_1_74455sec85bf4d4cb03	Standard HDD LRS	127	500	60	SSE with PMK & ADE	Read/write

Data disks

LUN	Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (MB/s)	Encryption	Host caching
0	srk	Standard HDD LRS	32	500	60	SSE with PMK & ADE	None

5. Now Connect your virtual machine with RDP.

RDP

To connect to your virtual machine via RDP, select an IP address, optionally change the port number, and download the RDP file.

IP address *

Public IP address (20.89.109.8)

Port number *

3389

Download RDP File

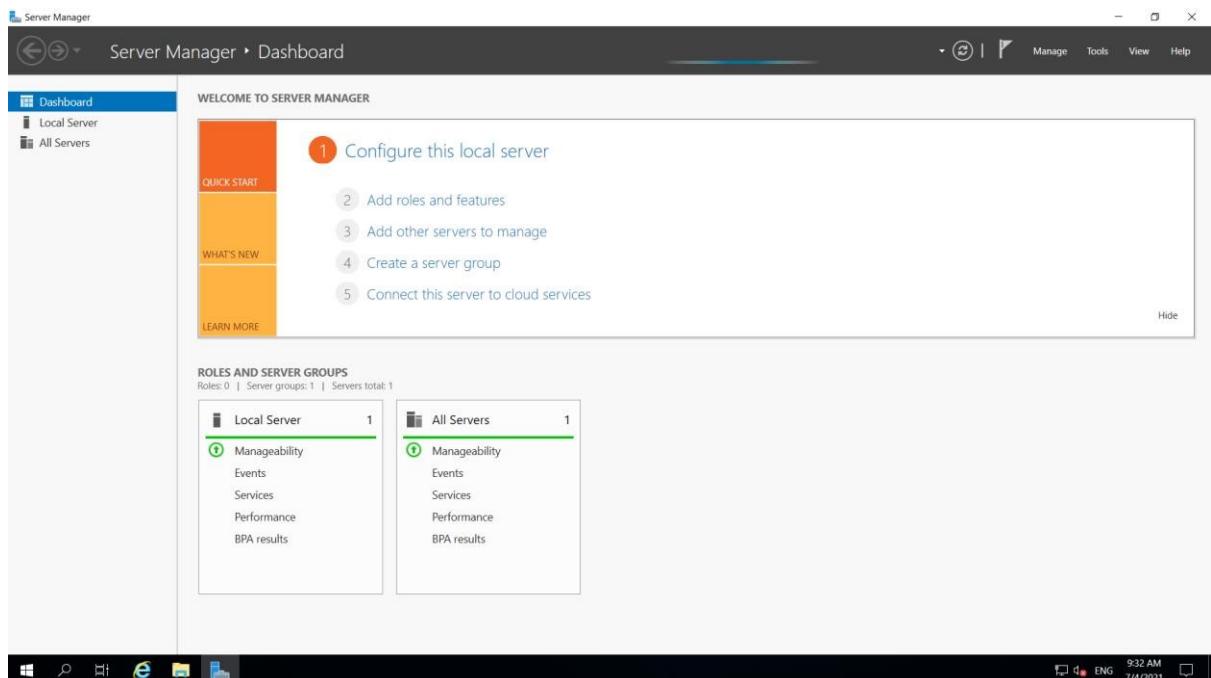
Can't connect?

- Test your connection
- Troubleshoot RDP connectivity issues

Provide feedback

Tell us about your connection experience

- 6.a. Open the Server Manager.



b. Go to the File and Storage block

c. Select Disk

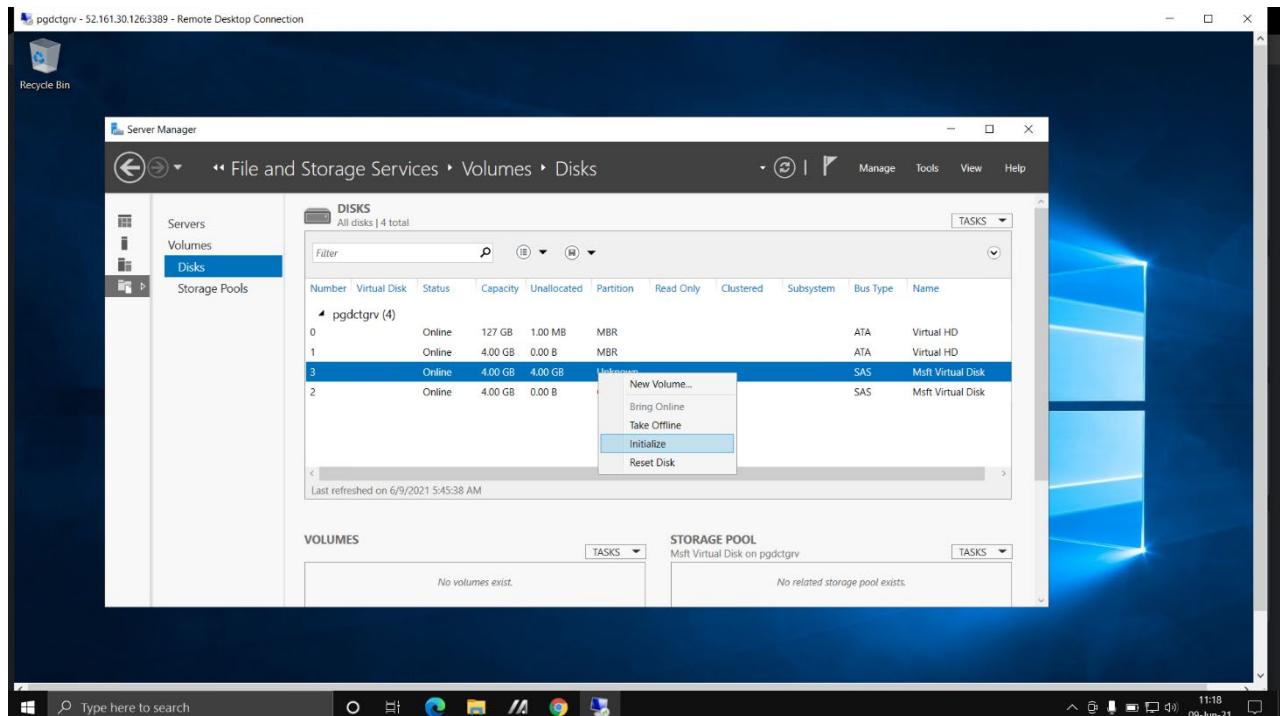
d. You will see the disk is attached.

This screenshot shows the 'Disks' section of the Server Manager under 'File and Storage Services > Volumes > Disks'. The left sidebar has 'Servers', 'Volumes', 'Disks' (which is selected), and 'Storage Pools'. The main pane displays a table of disks:

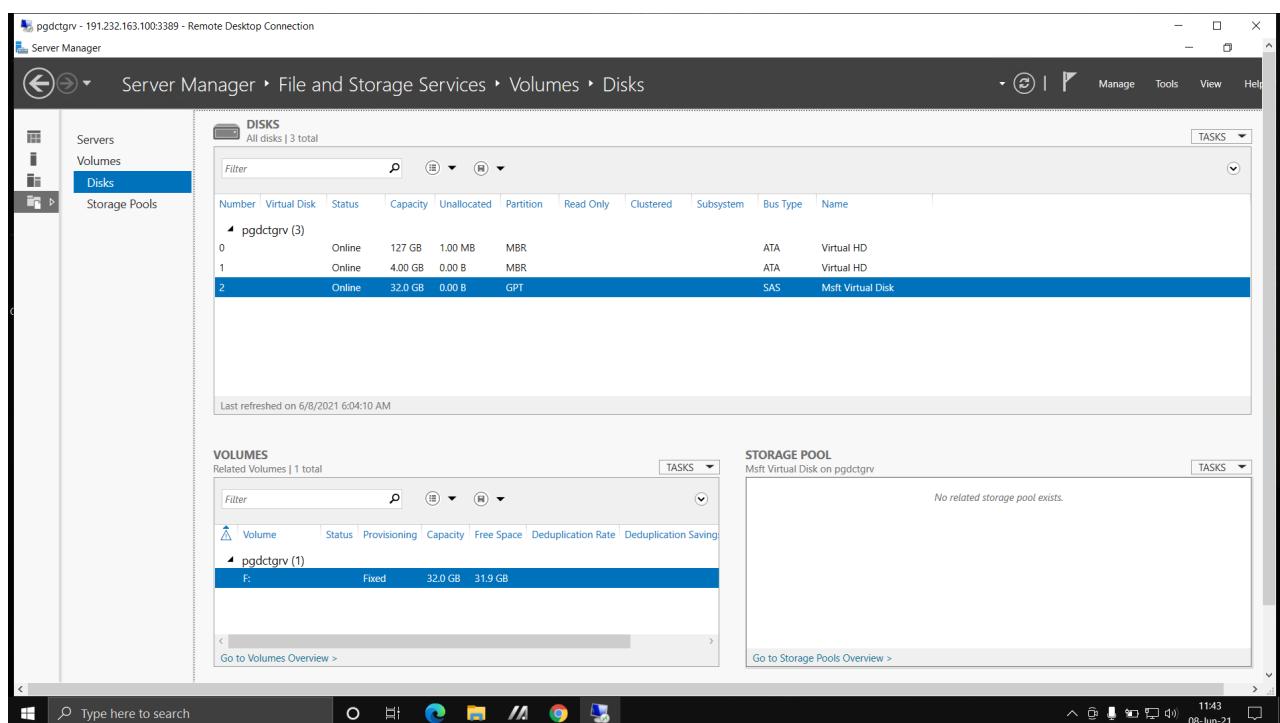
Number	Virtual Disk	Status	Capacity	Unallocated	Partition	Read Only	Clustered	Subsystem	Bus Type	Name
0	pgdctgrv (3)	Online	127 GB	1.00 MB	MBR	No	No	ATA	Virtual HD	
1		Online	4.00 GB	0.00 B	MBR	No	No	ATA	Virtual HD	
2		Online	32.0 GB	0.00 B	GPT	No	No	SAS	Mft Virtual Disk	

Below the table, it says 'Last refreshed on 6/8/2021 6:04:10 AM'. The 'VOLUMES' and 'STORAGE POOL' sections are also visible but show no data.

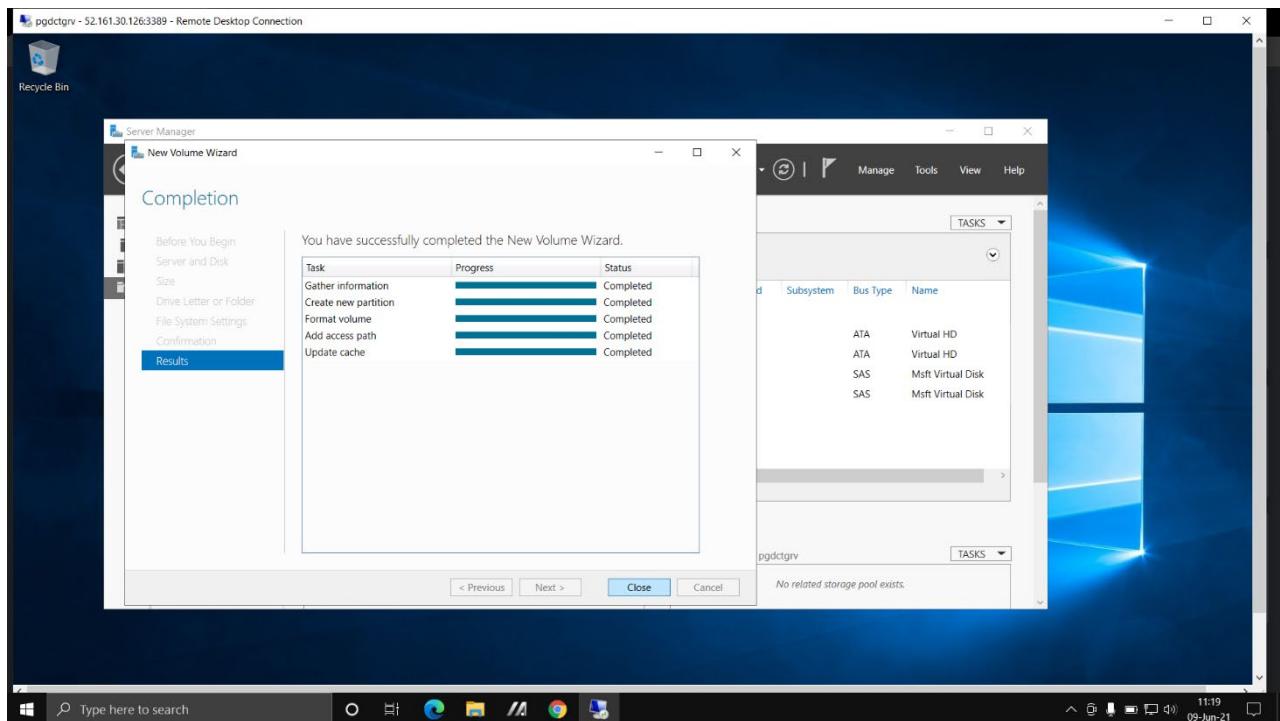
7. a. Right click on disk; you will see the initialize



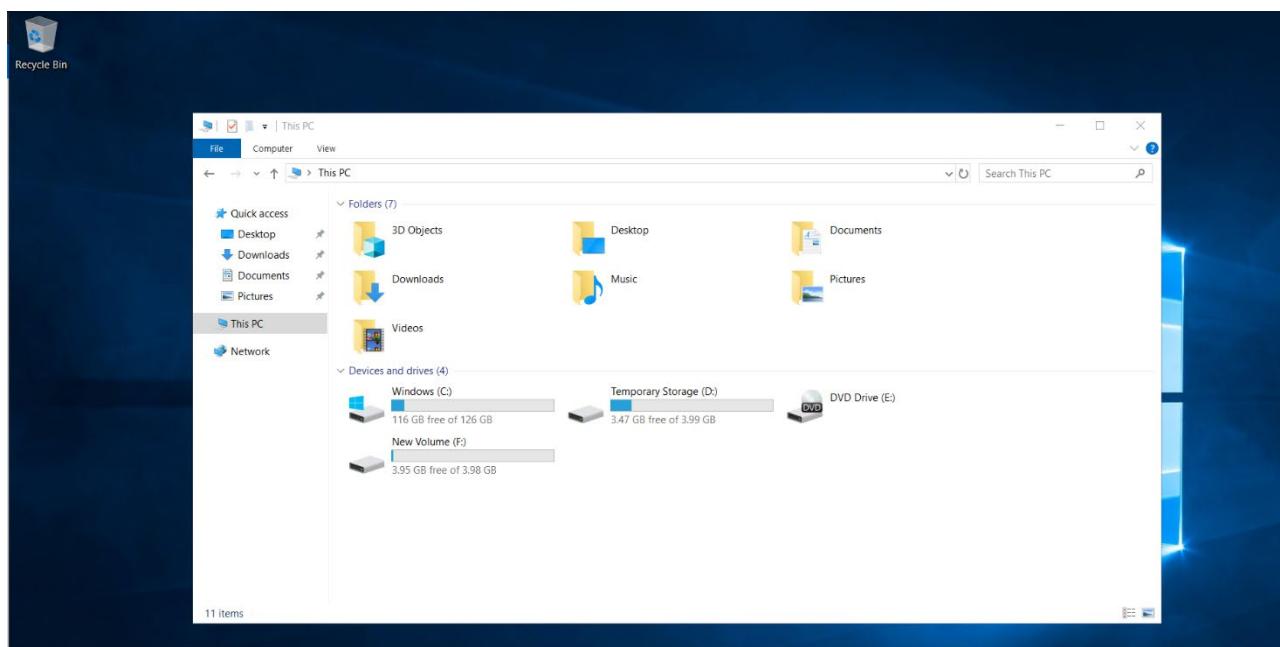
b. Again click on disk and then click on new volume.



8. Assign a letter to your disk and click create.



9. Go to file manager you will see your disk is attached.



Steps to attach unmanaged disk to virtual machine:

1. Create the Virtual Machine.
2. In disk go to the advanced tab and untick the managed disk option.
3. It will show you to create storage account, give name to storage account and click OK.

4. Go to the default settings and create virtual machine.

5. Go to disk, click on data disk and it will show you attach unmanaged disk.

6. Give the name to disk, choose storage type as standard HDD.

7. In storage container, go to your storage account and create container.

Storage accounts

Containers

Name	Last modified	Public access level	Lease state
bootdiagnostics-pgdc-5120a3e8-9bd2-4625-a3c3-f545a0023d2a	7/7/2021, 2:00:38 PM	Private	Available
grv	7/7/2021, 2:26:02 PM	Private	Available

8.Go to the properties, copy the URL and paste it in storage container option.

Attach unmanaged disk

Name *
grvdisk

Source type *
New (empty disk)

Storage type *
Standard HDD

Size (GiB) *
1024

PERFORMANCE

Provisioned IOPS: 500
Provisioned throughput (MB/s): 60

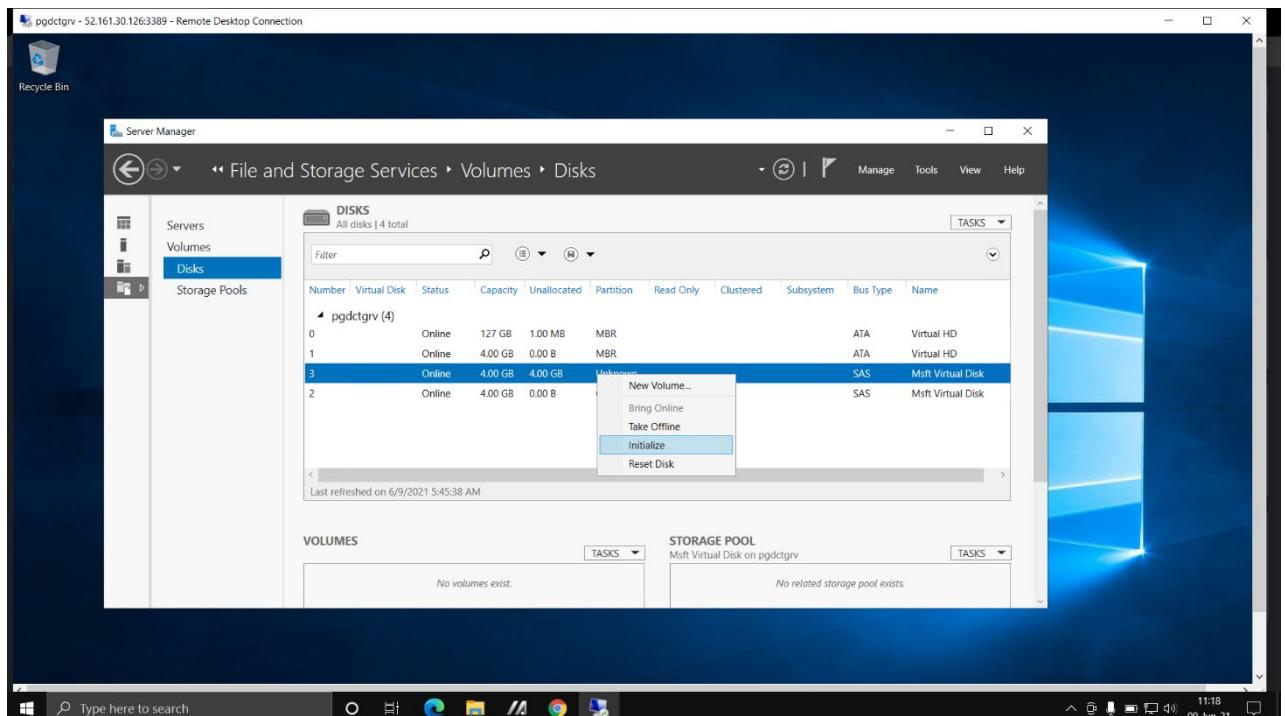
Storage container *
https://pgdctgrvdiag.blob.core.windows.net/grv

Storage blob name *
grvdisk.vhd

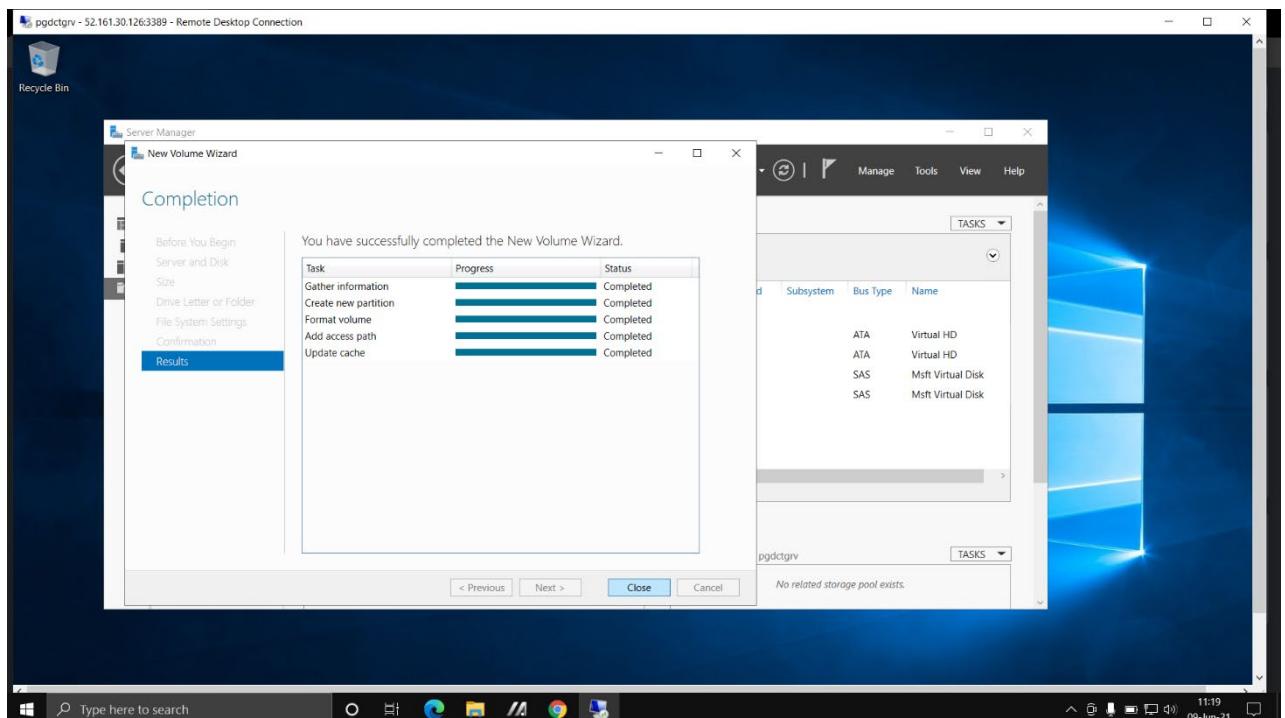
OK

9.Now connect with RDP and go to server manager.

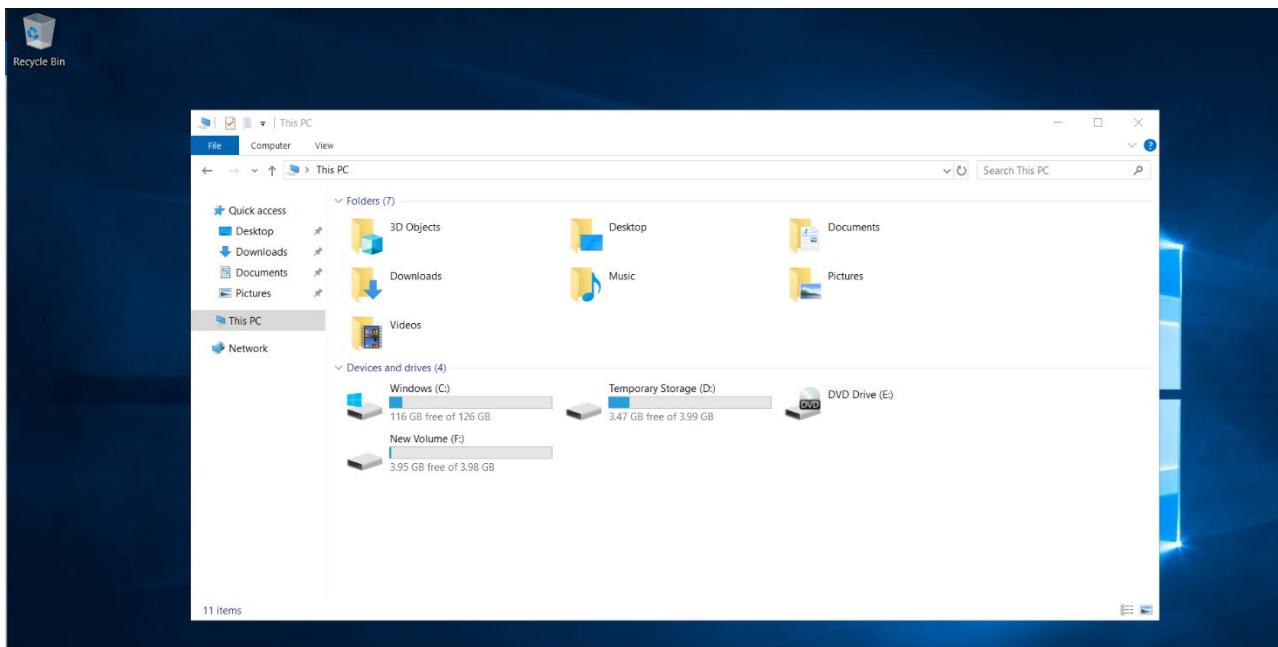
- a. Go to the file and storage.
- b. Go to the Disk and you will see the disk.



10. Initialize the disk and create new volume.



11. Go to file manager you will see the unmanaged disk has been attached.



Steps to attached shared disk to virtual machine:

1. Go to disk.
2. Click on create data disk.
3. Choose your resource group and give the name of disk and choose region.

Create a managed disk

Basics [Encryption](#) [Networking](#) [Advanced](#) [Tags](#) [Review + create](#)

Select the disk type and size needed for your workload. Azure disks are designed for 99.999% availability. Azure managed disks encrypt your data at rest, by default, using Storage Service Encryption. [Learn more about disks.](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

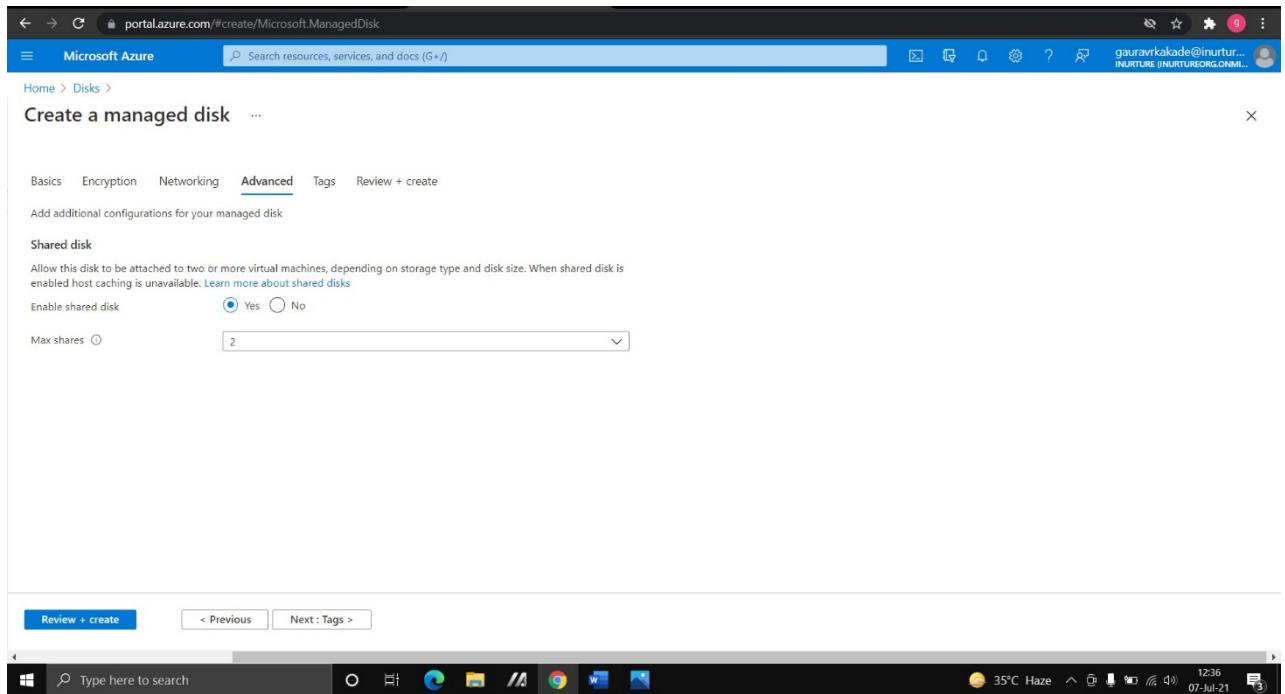
Subscription * [Azure_subscription](#)
 Resource group * [\(New\) pgdctgrv](#) [Create new](#)

Disk details

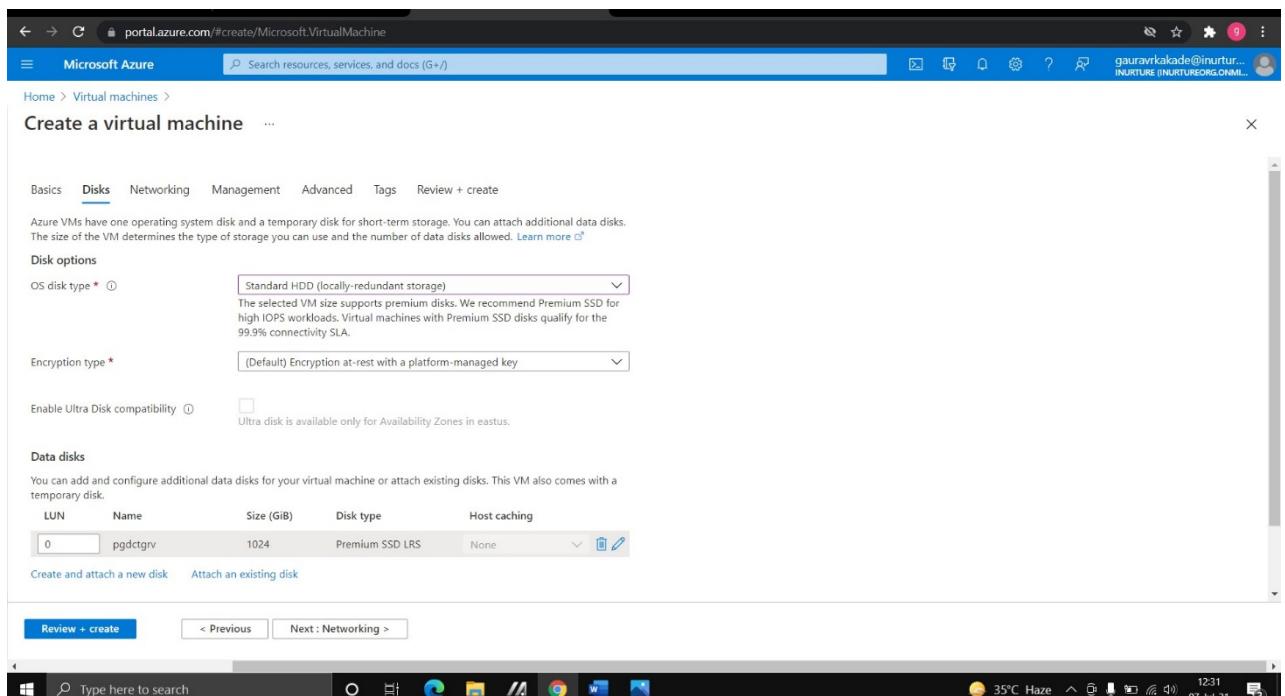
Disk name * [srk](#)
 Region * [\(US\) East US](#)
 Availability zone [None](#)
 Source type [None](#)
 Size * [1024 GB](#) [Premium SSD, IOPS](#)

[Review + create](#) [Next : Encryption >](#)

4. Go with default settings.
5. In advanced enable shared disk option and click on next and create disk.



6.Create two virtual machines and in disk option click on attach existing disk.



7.Now connect both the virtual machine with RDP.

8.In server manager go to file and storage, then go to disk ,your shared disk is attached to both the virtual machines.

The image displays two side-by-side screenshots of the Windows Server Manager interface, specifically the File and Storage Services section under Volumes > Disks.

Left Screenshot (pgdctrv):

- DISKS:** Shows three disks:

Number	Virtual Disk	Status	Capacity	Unallocated	Partition	Read Only	Clusters
0	Online	127 GB	1.00 MB		MBR		
1	Online	4.00 GB	0.00 B		MBR		
2	Online	4.00 GB	4.00 GB		Unknown		
- VOLUMES:** Shows 'Related Volumes | 0 total'. A message says 'No volumes exist.' and 'To create a volume, start the New Volume Wizard.'

Right Screenshot (pgdct):

- DISKS:** Shows three disks:

Number	Virtual Disk	Status	Capacity	Unallocated	Partition	Read Only	Clusters
0	Online	127 GB	1.00 MB		MBR		
1	Online	4.00 GB	0.00 B		MBR		
2	Online	1.00 TB	1.00 TB		Unknown		
- VOLUMES:** Shows 'Related Volumes | 2 total'. A table lists two volumes:

Volume	Status	Provisioning	Capacity	Free Space	Deduplication Rate
\\?\Volume{78...}	Fixed	500 MB	465 MB		
C:	Fixed	127 GB	117 GB		

Steps to take snapshot of managed disks:

1. Create a virtual machine.
2. Create a disk.
3. Go to snapshot, enter the name of snapshot and select region & select snapshot type.
4. Select the disk whose snapshot you have to take.
5. Select the storage type as standard HDD.

The image shows a screenshot of the Microsoft Azure portal, specifically the 'Create snapshot' wizard.

Basics Step:

- Project details:** Subscription: Azure_subscription, Resource group: (New) pgdctrv.
- Instance details:**
 - Name: gvsnap
 - Region: (US) East US
 - Snapshot type: Full - make a complete read-only copy of the selected disk.
 - Incremental - save on storage costs by making a partial copy of the disk based on the difference between the last snapshot.
- Source subscription:** Azure_subscription.

Buttons at the bottom:

- Review + create
- < Previous
- Next : Encryption >

6. Create snapshot.

The screenshot shows the Microsoft Azure portal interface. The main title bar says "Microsoft Azure". Below it, a breadcrumb trail shows "Home > Snapshot.grvsnap-20210707130650 >". The main content area is titled "grvsnap" and "Snapshot". On the left, there's a sidebar with sections like Overview, Activity log, Access control (IAM), Tags, Settings, Encryption, Networking, Snapshot export, Properties, Locks, Automation, Tasks (preview), Export template, Support + troubleshooting, and New support request. The "Overview" tab is selected. The main pane displays details for the snapshot, including Resource group (pgdctgrv), Provisioning state (Succeeded), Location (East US), Subscription (Azure_subscription), Subscription ID (7a229d68-2778-43b5-ac57-37077d7b34d5), Date created (7/7/2021, 1:07:01 PM), Snapshot state (Unattached), Storage type (Zone-redundant), Source disk (srk), Size (1024 GiB), Encryption (Platform-managed key), Snapshot type (Full), and Network access policy (AllowAll). A "Tags (change)" section with a "Click here to add tags" link is also present.



7. Click on the create disk.

- a. Enter the disk name
- b. Size of the disk
- c. Go to the default settings and create the disk.

The screenshot shows the Microsoft Azure portal with a browser tab for "Create a managed disk - Microsoft Compute". The main title bar says "Microsoft Azure". Below it, a breadcrumb trail shows "Home > Snapshot.grvsnap-20210609124018 > grvsnap > Create a managed disk". The main content area is titled "Create a managed disk". At the top, there are tabs for Basics, Encryption, Networking, Advanced, Tags, and Review + create. The "Basics" tab is selected. The form fields include:

- Subscription: Azure_subscription
- Resource group: window (with a "Create new" link)
- Disk name: grvdisk
- Region: (US) West Central US
- Availability zone: None
- Source type: Snapshot
- Source subscription: Azure_subscription

At the bottom, there are "Review + create" and "Next : Encryption >" buttons. The Windows taskbar at the bottom shows the same status as the previous screenshot.

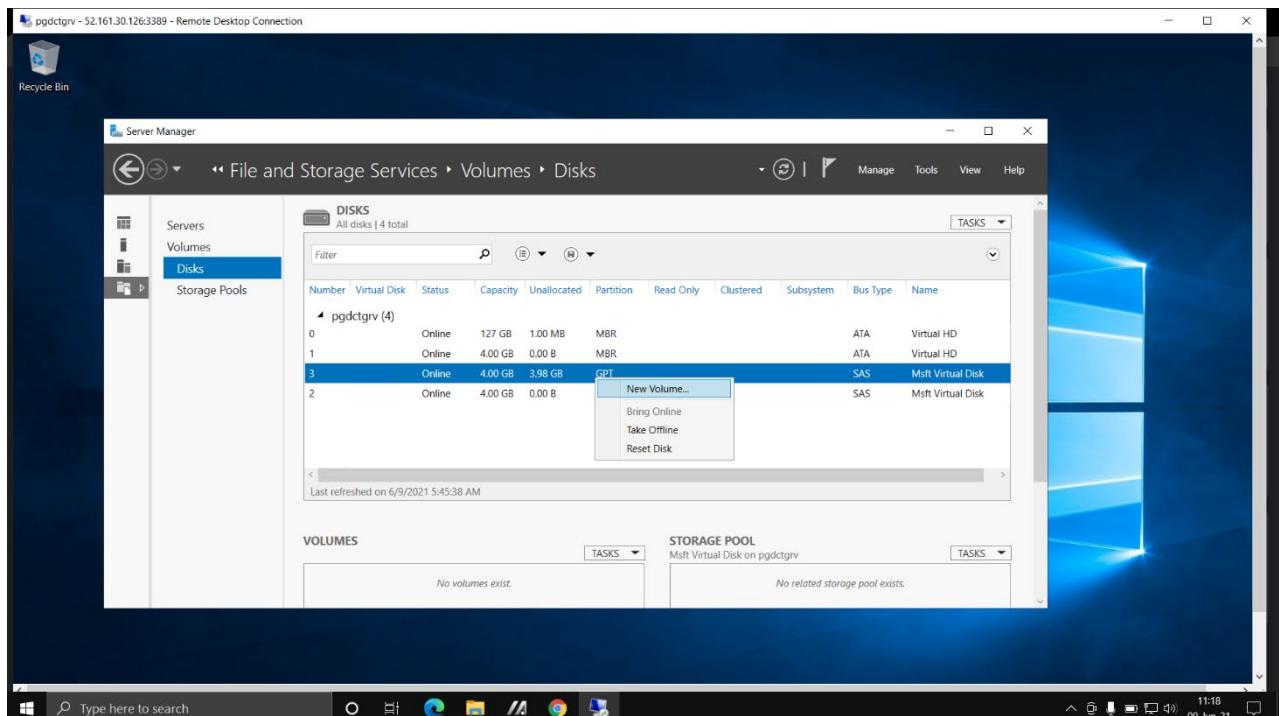
8. Go to virtual machine and create a new virtual machine.

The screenshot shows the Microsoft Azure portal interface. At the top, the URL is portal.azure.com/#blade/HubsExtension/DeploymentDetailsBlade/overview/id/%2fsubscriptions%2f7a229d68-2778-43b5-ac57-37077d7b34d5%2fResourceGroups%2Fpgdctgr%2fprovider.... The page title is "Microsoft.ManagedDisk-20210707125302 | Overview". A deployment status message says "Deployment succeeded" with a green checkmark. Deployment details include: Deployment name: Microsoft.ManagedDisk-20210707125302, Subscription: Azure_subscription, Resource group: pgdctgr. Deployment start time was 7/7/2021, 12:54:00 PM. Correlation ID: dd65e8ec-fca3-48f2-9a23-026521dbf4bb. On the left, there's a sidebar with "Overview", "Inputs", "Outputs", and "Template" options. Below the main content, there's a "Feedback" link. On the right, there's a "Security Center" section with links to "Secure your apps and infrastructure" and "Go to Azure security center >". There's also a "Free Microsoft tutorials" section with "Start learning today >" and a "Work with an expert" section with "Find an Azure expert >". The taskbar at the bottom shows various pinned icons.

9. Go to disk and click on attached existing disk and attach the disk whose snapshot you have been created.

The screenshot shows the Microsoft Azure portal interface. The URL is portal.azure.com/#@inurtureorg.onmicrosoft.com/resource/subscriptions/7a229d68-2778-43b5-ac57-37077d7b34d5/resourcegroups/pgdct/providers/Microsoft.Compute/virtualMachines/pgdct/disks. The page title is "pgdct | Disks". The left sidebar shows "Overview", "Activity log", "Access control (IAM)", "Tags", "Diagnose and solve problems", "Networking", "Connect", "Windows Admin Center (previ...", and "Disks" (which is selected). The main content area shows two sections: "OS disk" and "Data disks". Under "OS disk", there's a table with one row: Disk name: pgdct_OsDisk_1_74455sec85bf4d4cb03, Storage type: Standard HDD LRS, Size (GiB): 127, Max IOPS: 500, Max throughput: 60, Encryption: SSE with PMK & ADE, Host caching: Read/write. Under "Data disks", there's a table with one row: LUN: 0, Disk name: srk, Storage type: Standard HDD LRS, Size (GiB): 32, Max IOPS: 500, Max throughput: 60, Encryption: SSE with PMK & ADE, Host caching: None. The taskbar at the bottom shows various pinned icons.

10. Open the virtual machine and in server manager you will see your disk has attached.



Steps to Encrypt the Disk:

1. Create a Virtual Machine.
2. Connect to the RDP.
3. Go to the Vaults and create key vault.
4. Enter the name of key vault, region and choose pricing tier as standard.

Create key vault

Basics Access policy Networking Tags Review + create

Azure Key Vault is a cloud service used to manage keys, secrets, and certificates. Key Vault eliminates the need for developers to store security information in their code. It allows you to centralize the storage of your application secrets which greatly reduces the chances that secrets may be leaked. Key Vault also allows you to securely store secrets and keys backed by Hardware Security Modules or HSMs. The HSMs used are Federal Information Processing Standards (FIPS) 140-2 Level 2 validated. In addition, key vault provides logs of all access and usage attempts of your secrets so you have a complete audit trail for compliance.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *: Azure_subscription

Resource group *: pgdctgrv

Key vault name *: srk

Region *: East US

Review + create < Previous Next : Access policy >

5. In enable access to select disk encryption for volume encryption.
6. In key permission select all the options.

7. Create key vault.

8. Go to keys and click on Generate and Import.

9. Enter the name of the key and select the activation and expiration date.

Home > srk > srk > Create a key ...

Options Generate

Name * ⓘ srk

Key type ⓘ RSA (selected) EC

RSA key size ⓘ 2048 (selected) 3072 4096

Set activation date ⓘ Activation date: 07/07/2021 1:17:17 PM (UTC+05:30) Chennai, Kolkata, Mumbai, New Delhi

Set expiration date ⓘ

Enabled Yes (selected) No

Create

10. Click on create the key.
11. Go to virtual machine, click disk, create a data disk and save it.
12. Go to additional setting in disk.
13. In disk to encrypt choose OS and data disk, select the key vault and save it.

Inbox (468) - infogrk1@gmail.com ✉ Disk settings - Microsoft Azure ✉ +

Home > pgdct > Disk settings ...

Ultra disk

Enable Ultra disk compatibility ⓘ Yes No

Ultra disk is available only for Availability Zones in japaneast.

Encryption settings

Azure Disk Encryption (ADE) provides volume encryption for the OS and data disks. [Learn more about Azure Disk Encryption.](#)

Disks to encrypt ⓘ OS disk

Azure Disk Encryption is integrated with Azure Key Vault to help manage encryption keys. As a prerequisite, you need to have an existing key vault with encryption permissions set. For additional security, you can create or choose an optional key encryption key to protect the secret.

Key Vault * ⓘ srk
Manage selected vault
Create new

Key ⓘ srk
Create new

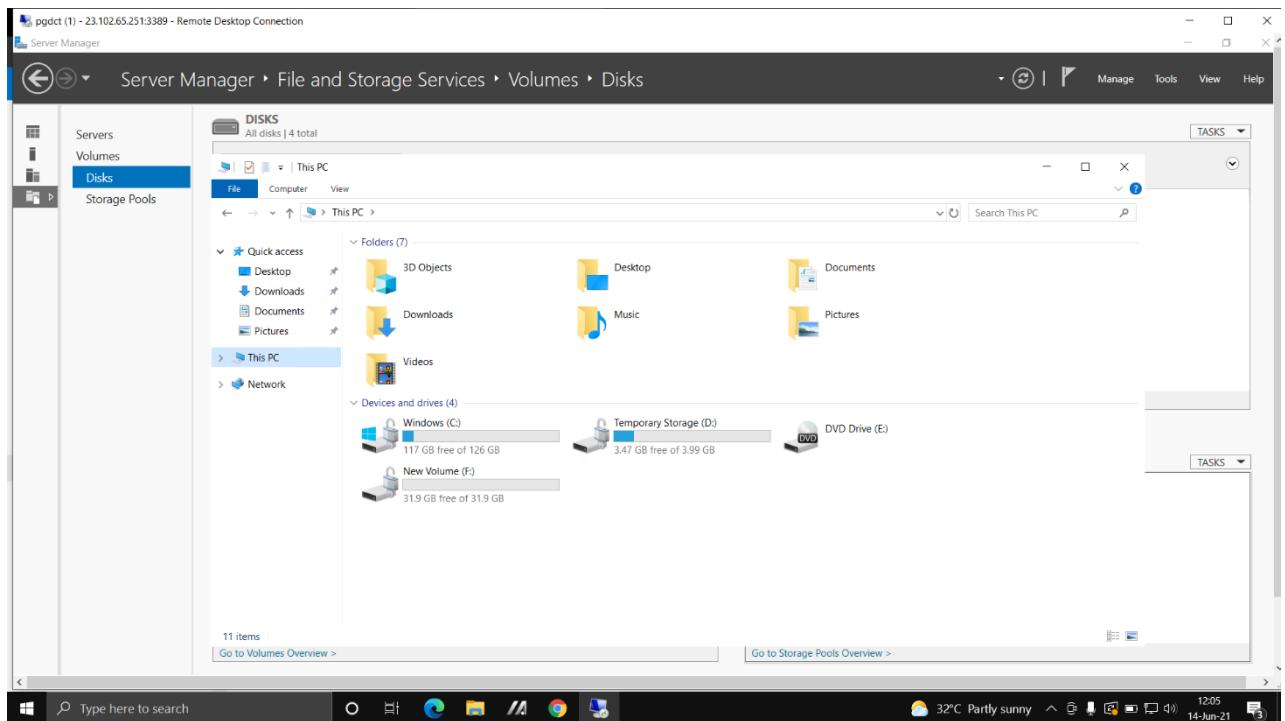
Version ⓘ b90b132802ed4821bee3ec7a4a9a6d64 (Current version)

Save Cancel

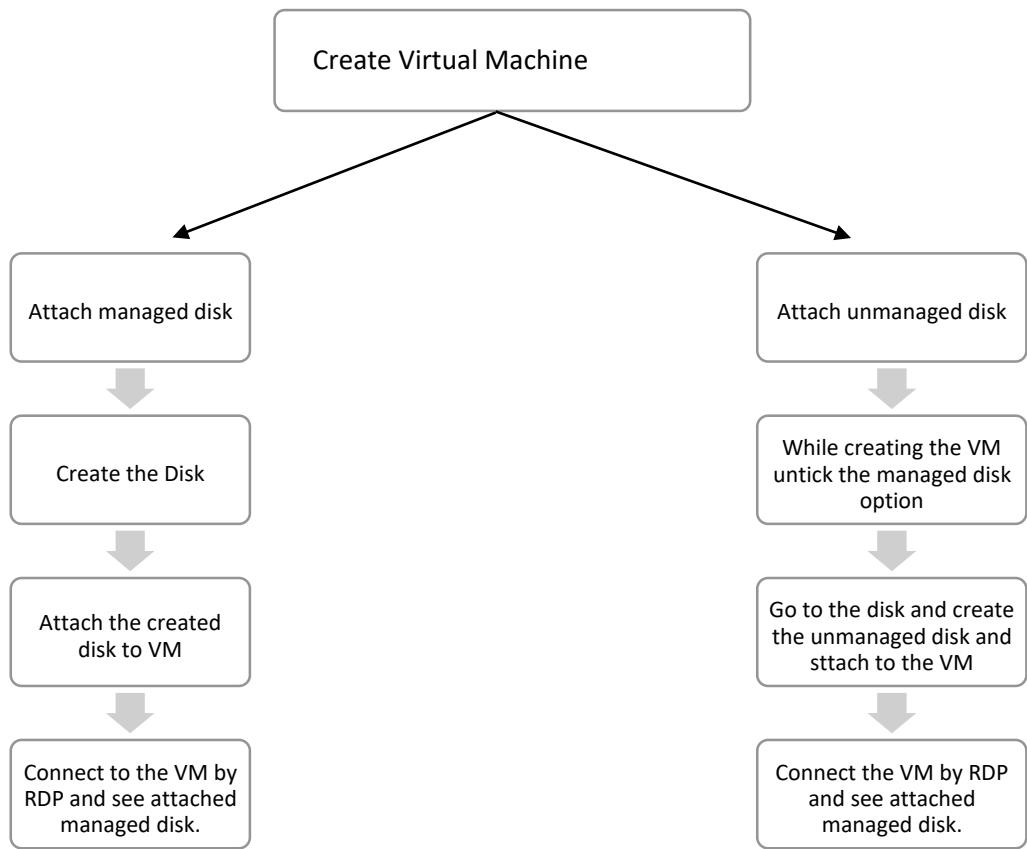
14. Now open the virtual machine and go to server manager, then to file and storage and to Disk.

15. You will see your data disk is attached.

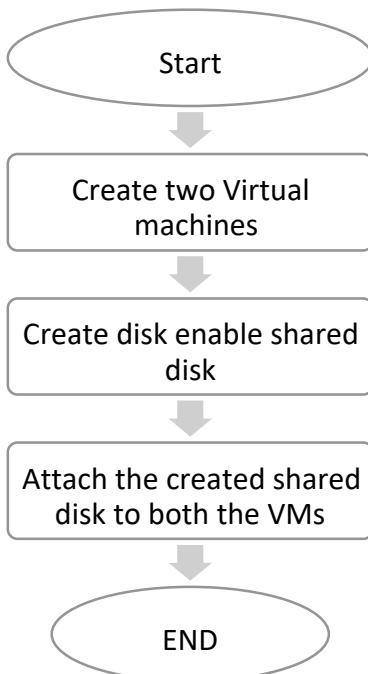
16. Go to file manager and you will see the OS disk and data disk is encrypted.



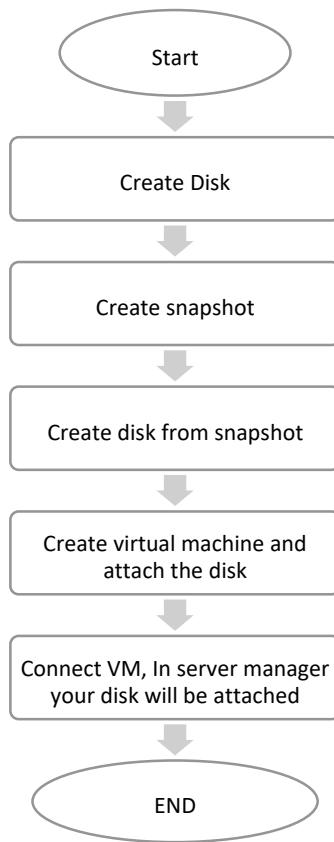
FLOWCHART:



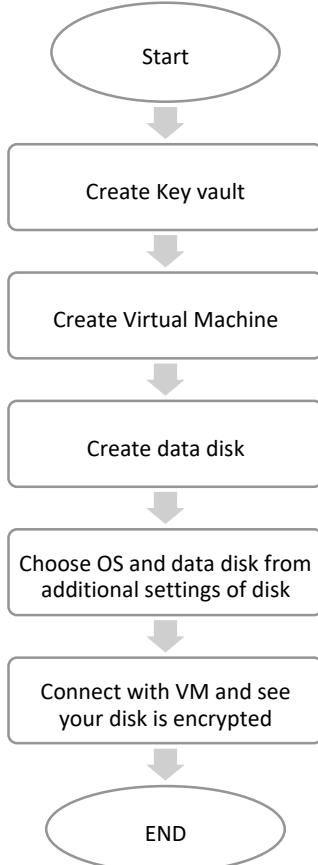
Shared Disk:



Taking Snapshots:



To Encrypt the Disk:



RESULT:

1. We can attach the data managed, unmanaged and shared disk to virtual machine and see the corresponding output by connecting virtual machine through RDP. Then in server manager we can see our disk is attach and we can also initialize the disk and can attach the disk by creating new volume.
2. We can also take snapshot of disk and then can attach the disk to virtual machine.
3. We can also encrypt the disk.

CONCLUSION:

We have successfully attached the data, shared and unmanaged disk to virtual machine and also have successfully take the snapshot of disk and encrypt the disk.

Experiment No. 2

Name of Experiment:

- 1.Deploying virtual machine in availability sets.
- 2.Deploying virtual machine in availability zone.
- 3.Deploying virtual machine in proximity placement group.

Prerequisites: Azure Portal, RDP.

Description:

Availability set: An availability set is a logical grouping of VMs that allows Azure to understand how your application is built to provide for redundancy and availability. Each virtual machine in your availability set is assigned an update domain and a fault domain by the underlying Azure platform. Each availability set can be configured with up to three fault domains and twenty update domains.

Availability zone: An Availability Zone is a high-availability offering that protects your applications and data from datacenter failures. Availability Zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking.

Proximity placement group: To get VMs as close as possible, achieving the lowest possible latency, you should deploy them within a proximity placement group. A proximity placement group is a logical grouping used to make sure that Azure compute resources are physically located close to each other. Proximity placement groups are useful for workloads where low latency is a requirement.

Algorithm:

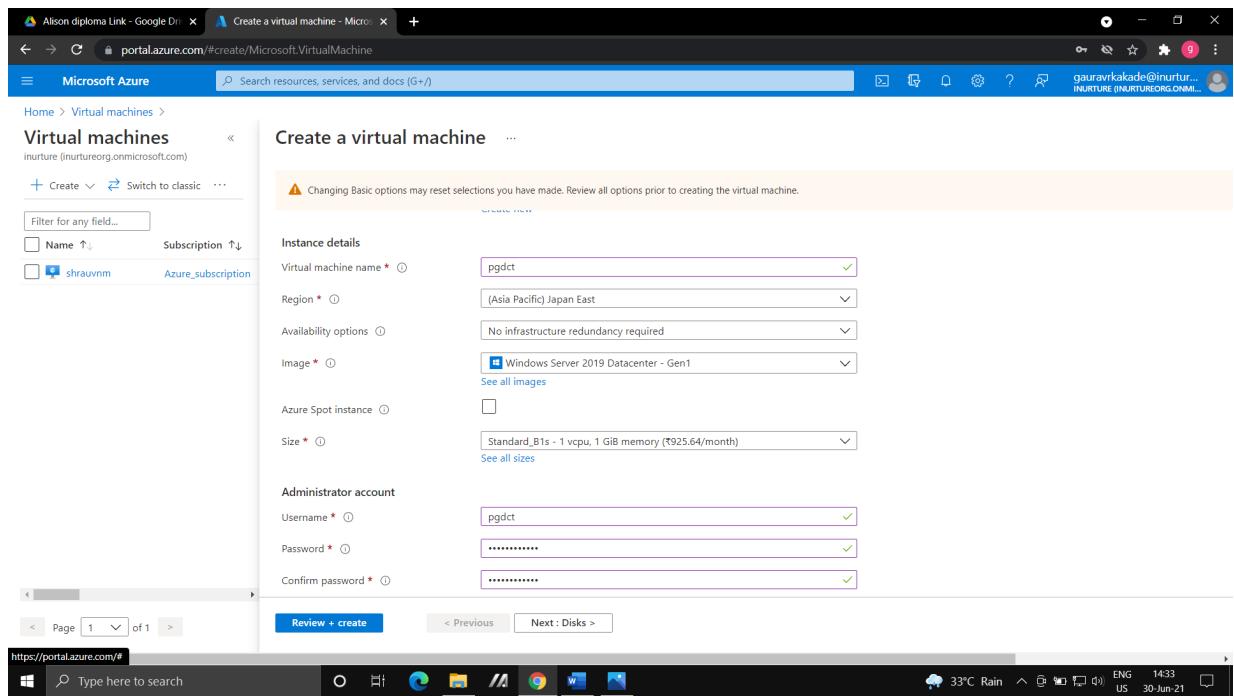
Steps to deploy virtual machine in availability sets:

- 1.Go to resource group and create a resource group.
- 2.Go to availability sets>>create availability sets>>enter the name of availability sets.
- 3.Select 2 fault domain and 2 update domain.

4.Create availability set.

5.Go to virtual machine>>select availability set you have created>>go with the default setting.

6.Create another virtual machine and choose availability set that you have created >>create virtual machine.



7.Go to availability set>>select your availability set>>you will see 2 virtual machine that you have created.

Name	Status	Colocation status	Fault Domain	Update Domain
ironex1	Running	0	0	0
ironex2	Running	1	1	1

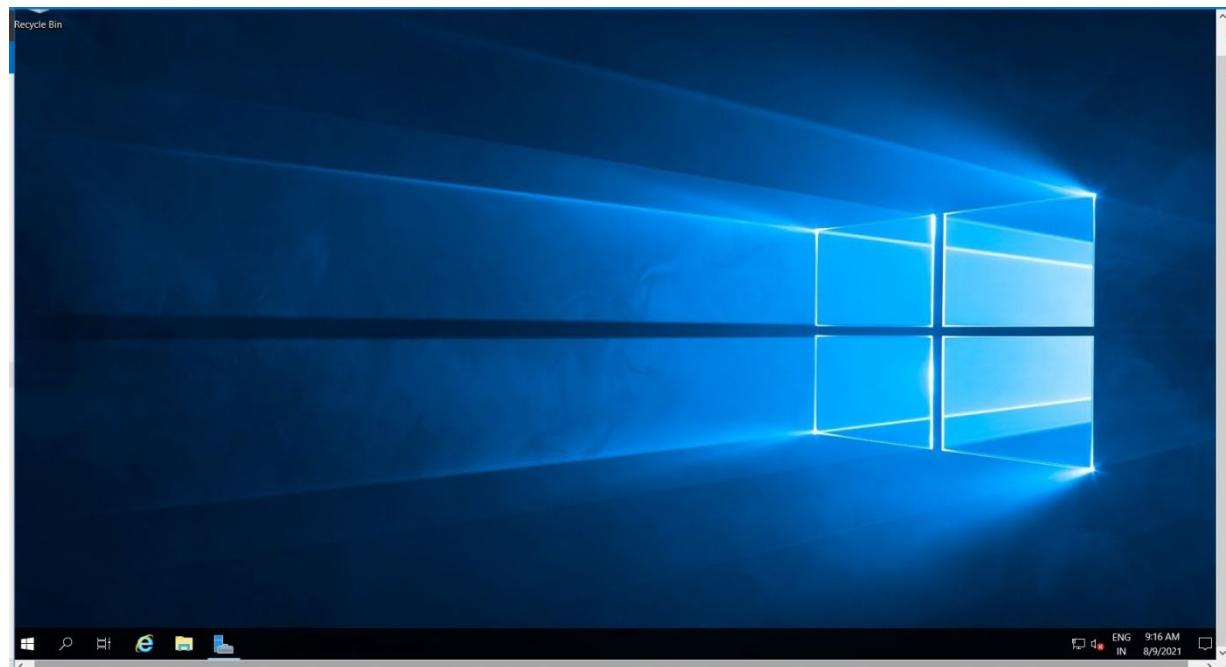
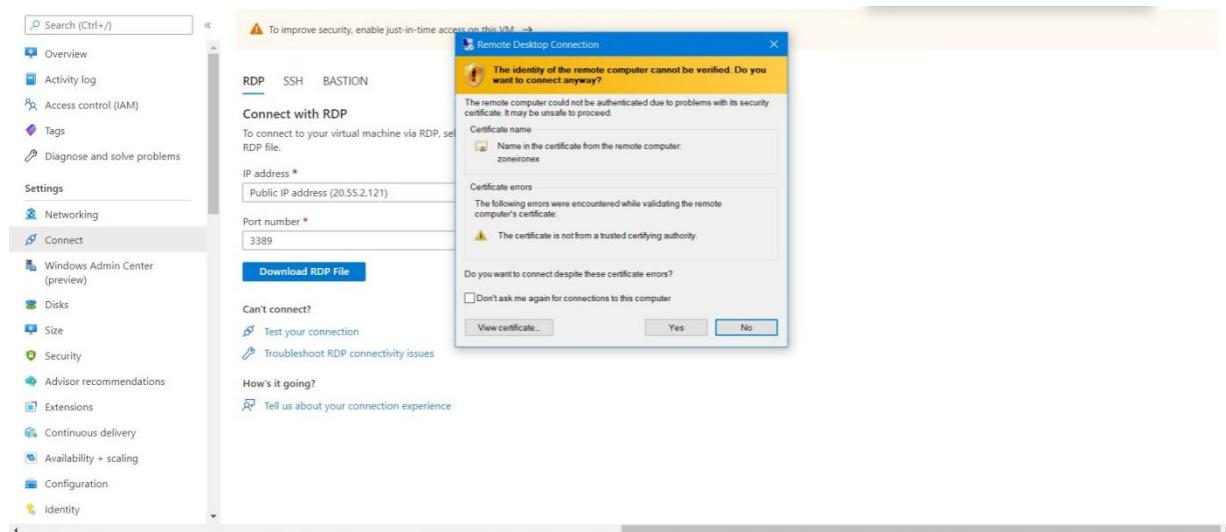
Steps to deploy virtual machine in availability zone:

- 1.go to virtual machine>>create new virtual machine.
- 2.Select the region where availability zone is available.
- 3.In availability options>select availability zone.

4. Select in which availability zone you want your virtual machine.

5. Keep everything as default and create virtual machine.

6. Try to connect with RDP.



Steps to deploy virtual machine in proximity placement group:

First use case: without availability set deploy virtual machine in proximity placement group.

- 1.Create resource group.
- 2.Create proximity group.

No proximity placement groups to display

A proximity placement group is a logical grouping used to make sure that Azure compute resources are physically located close to each other. Proximity placement groups are useful for workloads where low latency is a requirement.

[Create proximity placement group](#)

[Learn more](#)

[Review + create](#) | [Next : Tags >](#)

3.Go to virtual machine>>In advance select the proximity placement group that you have created.

[Enable user data](#)

Host

Azure Dedicated Hosts allow you to provision and manage a physical server within our data centers that are dedicated to your Azure subscription. A dedicated host gives you assurance that only VMs from your subscription are on the host, flexibility to choose VMs from your subscription that will be provisioned on the host, and the control of platform maintenance at the level of the host. [Learn more](#)

Host group No host group found

Proximity placement group

Proximity placement groups allow you to group Azure resources physically closer together in the same region. [Learn more](#)

Proximity placement group proxynex

VM generation

Generation 2 VMs support features such as UEFI-based boot architecture, increased memory and OS disk size limits, Intel® Software Guard Extensions (SGX), and virtual persistent memory (vPMEM). [Click here to learn more about Gen2 virtual machine capabilities.](#)

VM generation Gen 1

[Review + create](#) | [Next : Tags >](#)

4.Now try to connect the virtual machine with RDP.



Second Use Case: to check the combination of availability set and proximity placement group.

1.Create availability set.

2.Select your proximity placement group>>create availability set.

A screenshot of the Microsoft Azure portal. The URL in the address bar is https://portal.azure.com/#@inutureorg.onmicrosoft.com/resource/subscriptions/7a29d68-2778-43b5-ac57-37077d7b34d5/resourceGroups/ironex/providers/Microsoft.Compute/availabilitySets/setironex. The page title is "setironex | Configuration".

The left sidebar shows "Availability sets" with a sub-section for "setironex". The main content area has tabs for "Overview", "Activity log", "Access control (IAM)", and "Tags". The "Configuration" tab is selected. On the right, there is a "Proximity placement group" dropdown set to "proxynex". The bottom of the screen shows a navigation bar with "Page 1 of 1".

3.Go to virtual machine>>choose the availability set that you have created.

The screenshot shows the Azure portal interface for creating a new virtual machine. The top navigation bar includes links for Home, Virtual machines, and a search bar. The main content area is titled 'Create a virtual machine'. The first step, 'Project details', is active, showing the selection of 'Subscription' (Azure_subscription) and 'Resource group' (ironex). Below this, the 'Instance details' step is shown, with fields for 'Virtual machine name' (ironexcombo), 'Region' (US East US), 'Availability options' (Availability set), and 'Image' (Windows Server 2019 Datacenter - Gen1). At the bottom, there are buttons for 'Review + create' and 'Next : Disks >'. A sidebar on the left lists other virtual machines: akard, proxyironex, and zoneironex, each associated with a specific subscription.

4.Go with default option>>create virtual machine.

5.Try to connect with RDP.

Third Use Case: we will create 2 virtual machine with same availability zone and attach proximity group to 2 virtual machine.

1.Create resource group.

2.Create proximity placement group.

<https://portal.azure.com/#create/Microsoft.ProximityPlacementGroup>

Create Proximity Placement Group

Basics Tags Review + create

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * Resource group * Create new

Instance details

Region * Proximity placement group name *

Review + create < Previous Next : Tags >

3.Create 2 virtual machine with same availability zone.

<https://portal.azure.com/#create/Microsoft.VirtualMachine>

Virtual machines

Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * Resource group * Create new

Instance details

Virtual machine name * Region * Availability options * Availability zone * Image * See all images

Review + create < Previous Next : Disks >

The screenshot shows the Azure portal interface for creating a new virtual machine. The main title is 'Create a virtual machine'. On the left, there's a sidebar with 'Virtual machines' and a list of hosts: ironexcombo (Azure_subscription), proxyironex (Azure_subscription), and zoneironex (Azure_subscription). The main form has several sections: 'Host' (with a note about Dedicated Hosts), 'Proximity placement group' (with a dropdown set to 'iproxy'), 'VM generation' (radio buttons for Gen 1 and Gen 2, with Gen 1 selected), and 'Review + create' at the bottom. A note at the top right says 'virtual machine. Don't use user data for storing your secrets or passwords. Learn more about user data for VMs'.

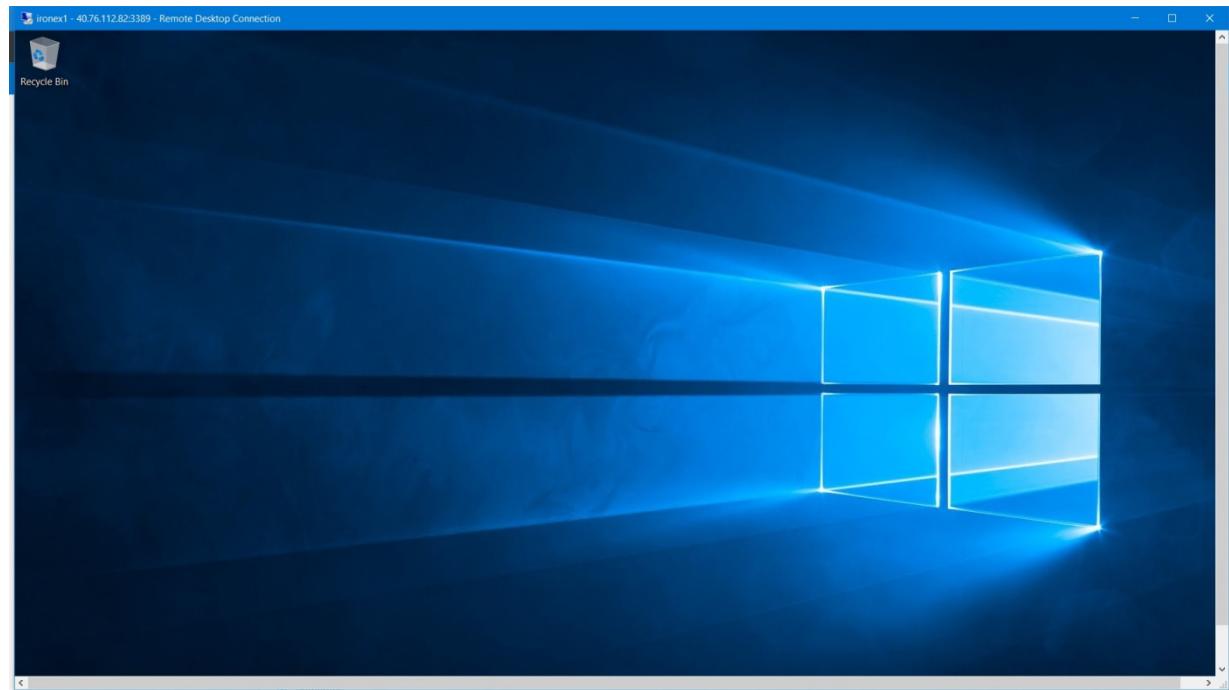
4.In advanced, select the proximity placement group.

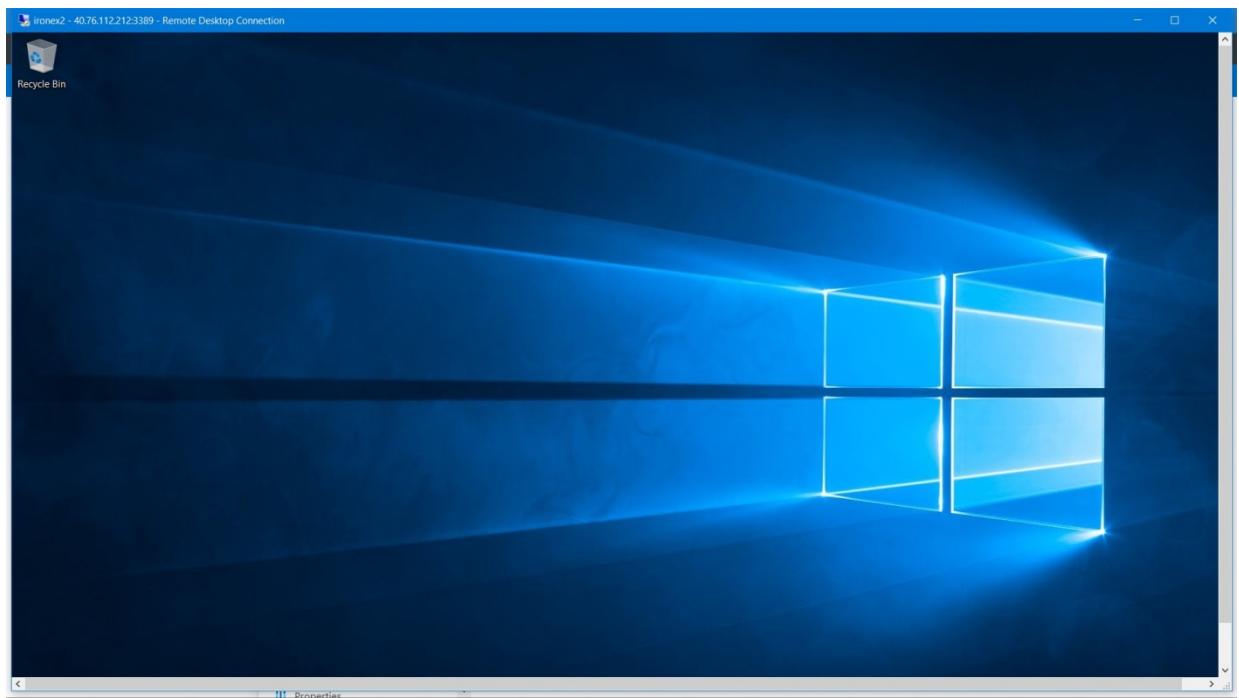
This screenshot shows the 'Create a virtual machine' wizard on the 'Basics' tab. It includes sections for 'Project details' (Subscription: Azure_subscription, Resource group: ironex) and 'Instance details' (Virtual machine name: ironex2, Region: (US) East US, Availability zone: 2, Image: Windows Server 2019 Datacenter - Gen1). Navigation buttons at the bottom include 'Review + create', '< Previous', and 'Next : Disks >'.

The screenshot shows the Microsoft Azure portal interface for creating a new virtual machine. The URL in the address bar is <https://portal.azure.com/#create/Microsoft.VirtualMachine>. The main title is "Create a virtual machine". On the left, there's a sidebar titled "Virtual machines" with a list of hosts: "ironexcombo" (selected), "proxyironex", and "zoneironex", each associated with "Azure_subscription". The "Host" section describes Azure Dedicated Hosts. The "Host group" dropdown shows "No host group found". The "Proximity placement group" section shows "iproxy" selected. The "VM generation" section has "Gen 1" selected. At the bottom, there are navigation buttons: "Review + create" (highlighted in blue), "< Previous", and "Next : Tags >".

5.Create virtual machines.

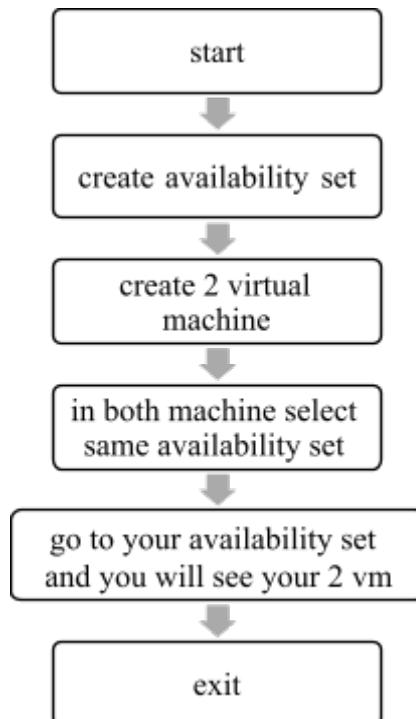
6.Try to connect both virtual machine with RDP.



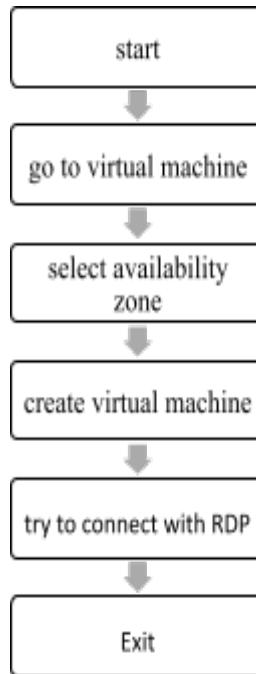


Flowchart:

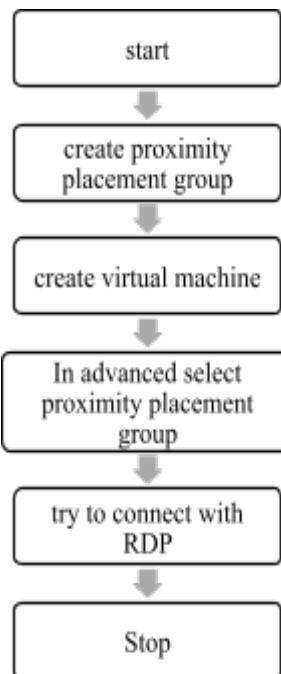
For availability set:

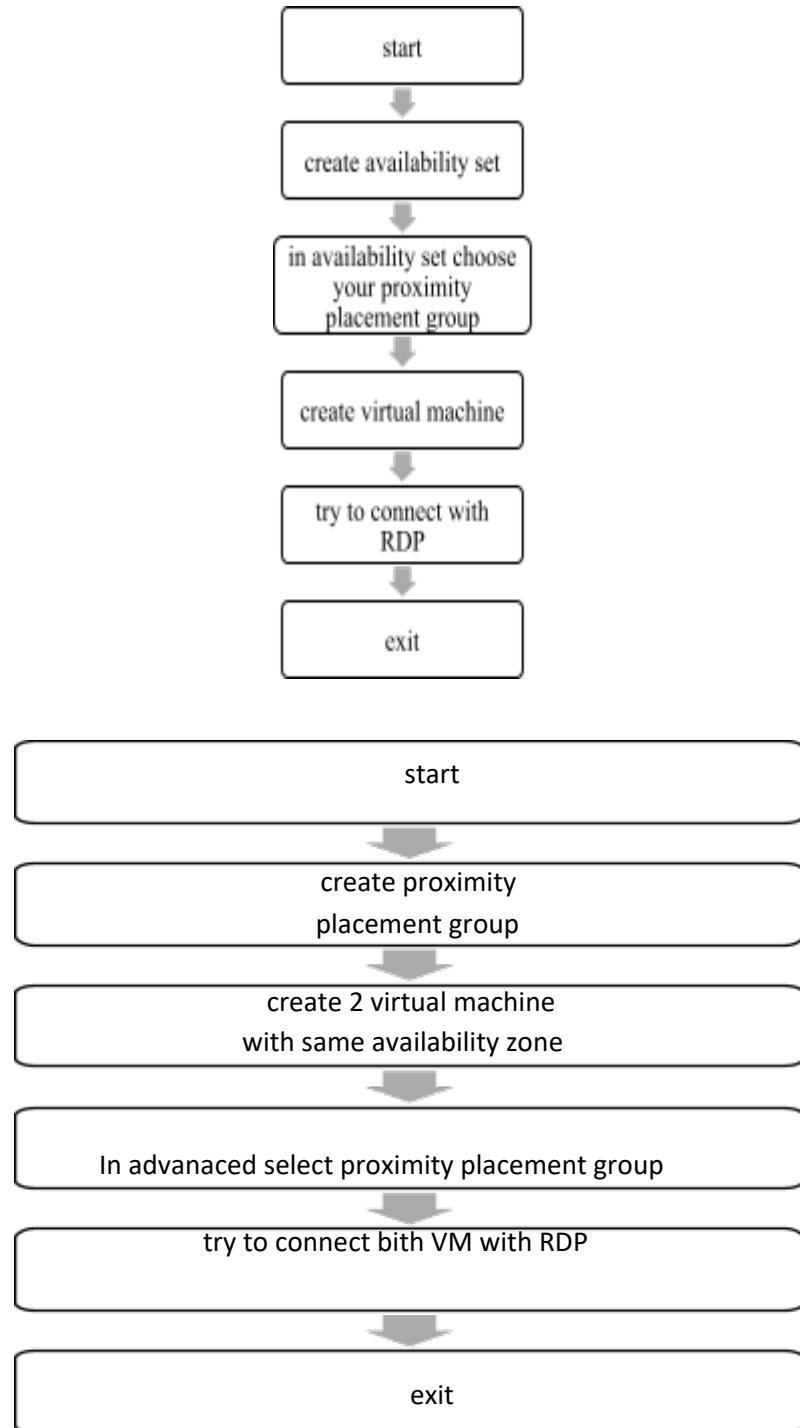


For availability zone:



For proximity placement group:





Result: Different virtual machine has been deployed in availability set, availability zone and proximity placement group. We have also seen the combination of proximity placement group with availability set and availability zone.

Conclusion: we have successfully deployed virtual machines in availability set, availability zone and proximity placement group.

EXPERIMENT NO: 3

AIM: creation of web app and to see app service logs.

PREREQUISITES: Azure portal, visual studio.

DESCRIPTION:

Azure web app:

- Azure Web Apps provides a platform to build an App in Azure without having to deploy, configure and maintain your own Azure VM's.
- You can build Web App using the ASP.NET, PHP, Node.js and Python.
- They also integrate common development environments which could be Visual Studio and GitHub.
- Azure Web App provides a host service that developers can use it to develop mobile or web app. Apart from this the developer can use to build API apps or Logic apps, which provides integration with SaaS.

ALGORITHM:

Steps to create web app:

- 1.Go to app service plan.
- 2.Give the name of plan.
- 3.Select the pricing tier as free f1>>create app service plan.

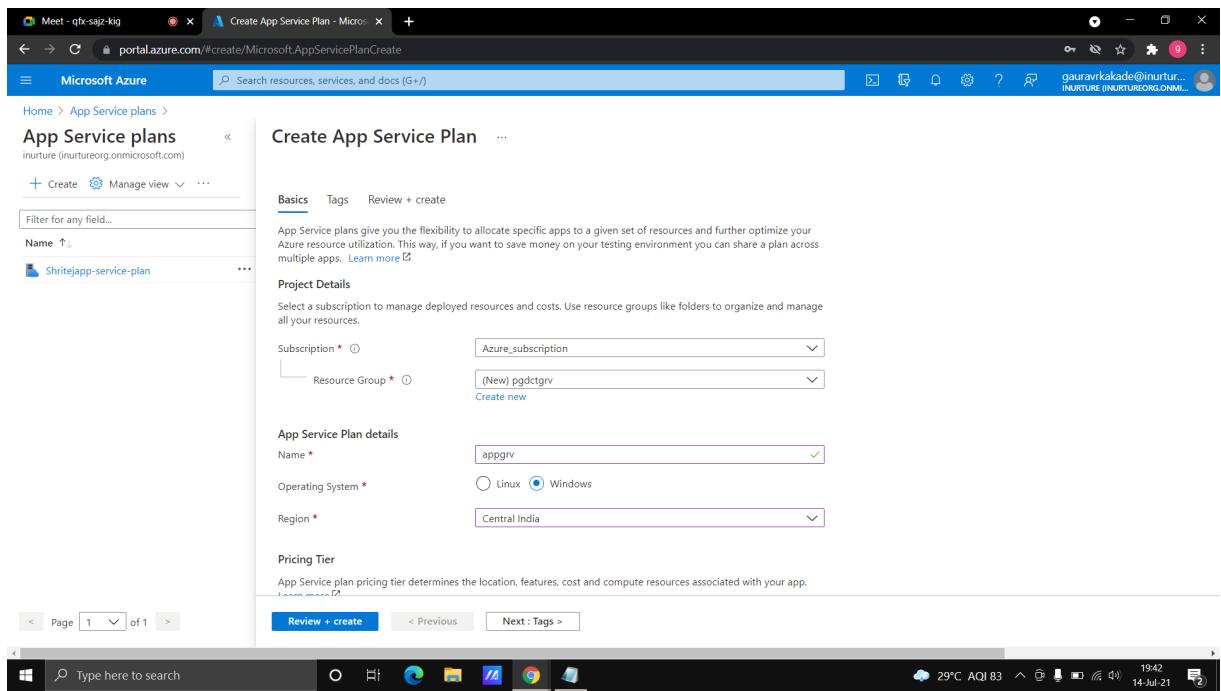


Fig: create app service plan

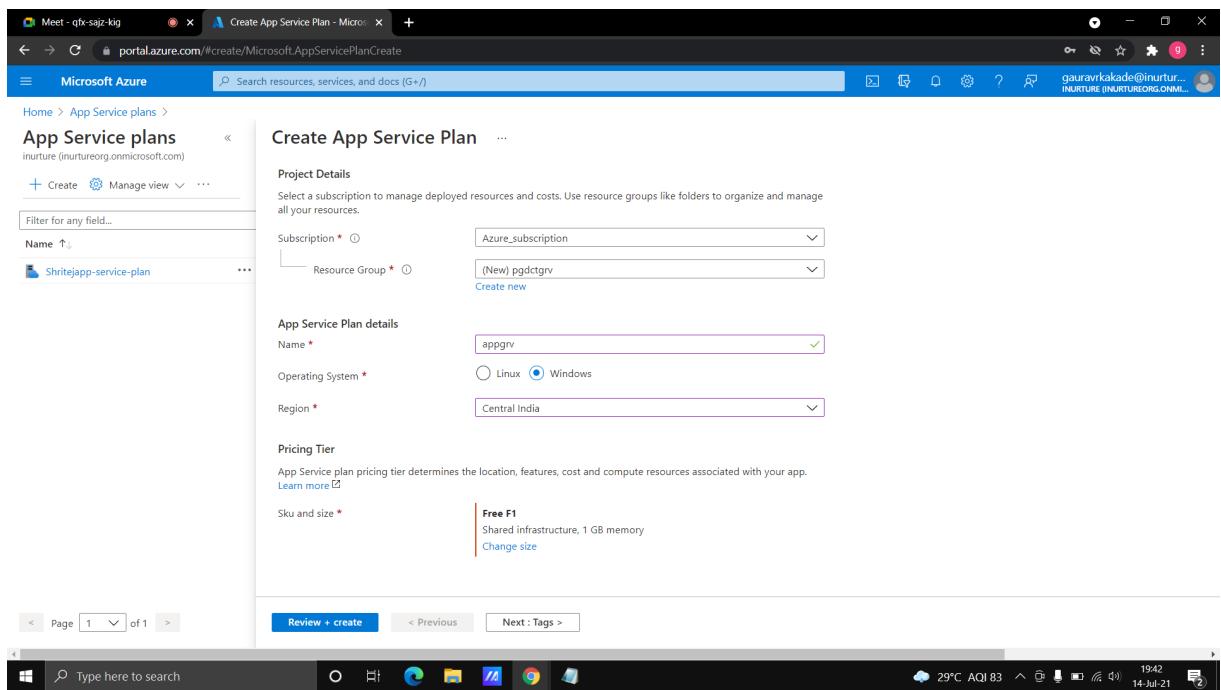


Fig: select size

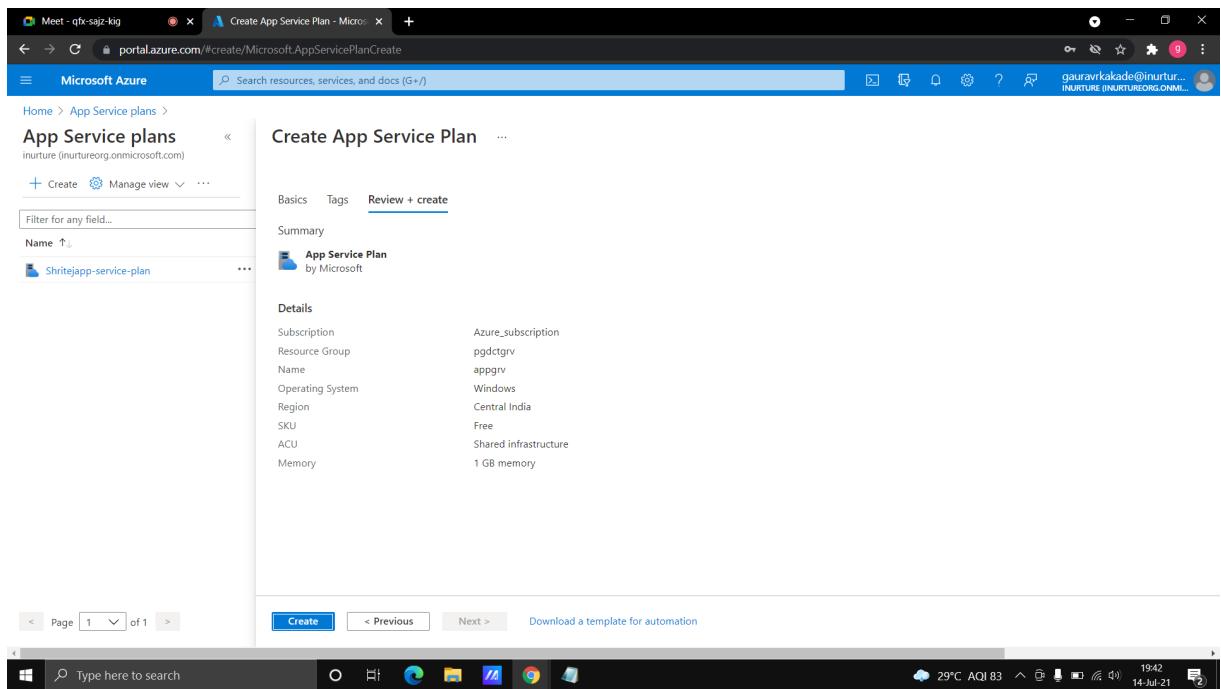


Fig: create app service plan

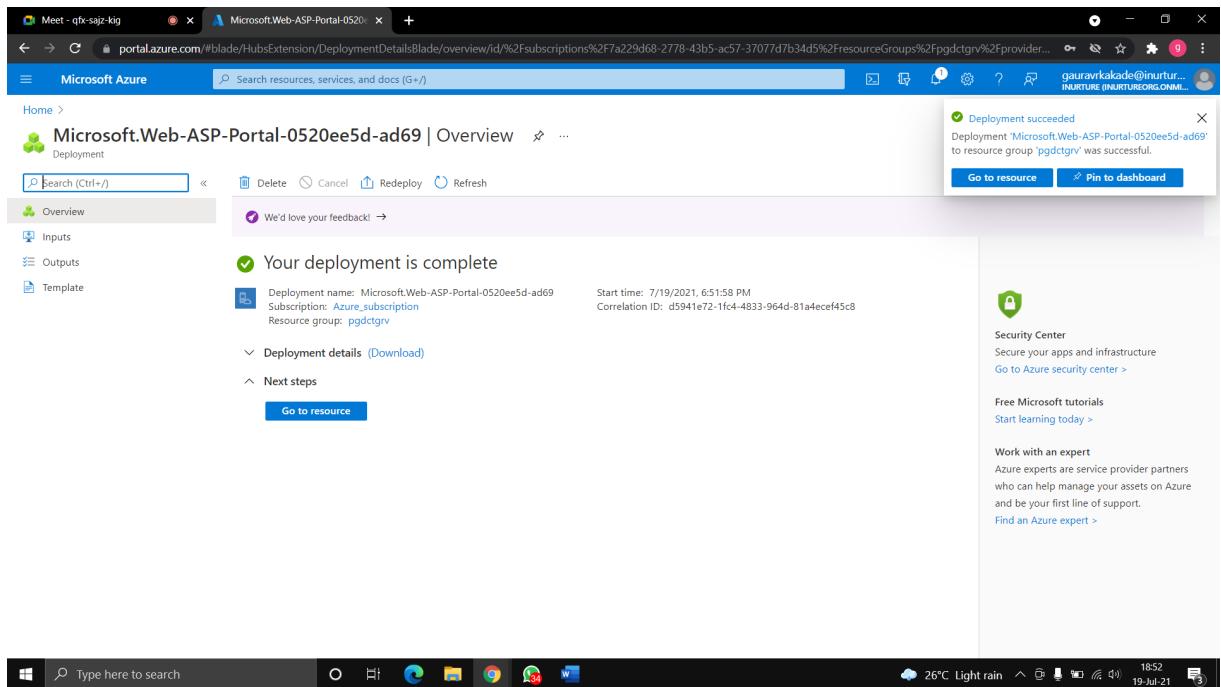


Fig: created app service plan

4.Go to web app.

5.Give the name of web app>>select the runtime stack as .NET core 3.1(LTS).

6.In windows plan select your windows plan.

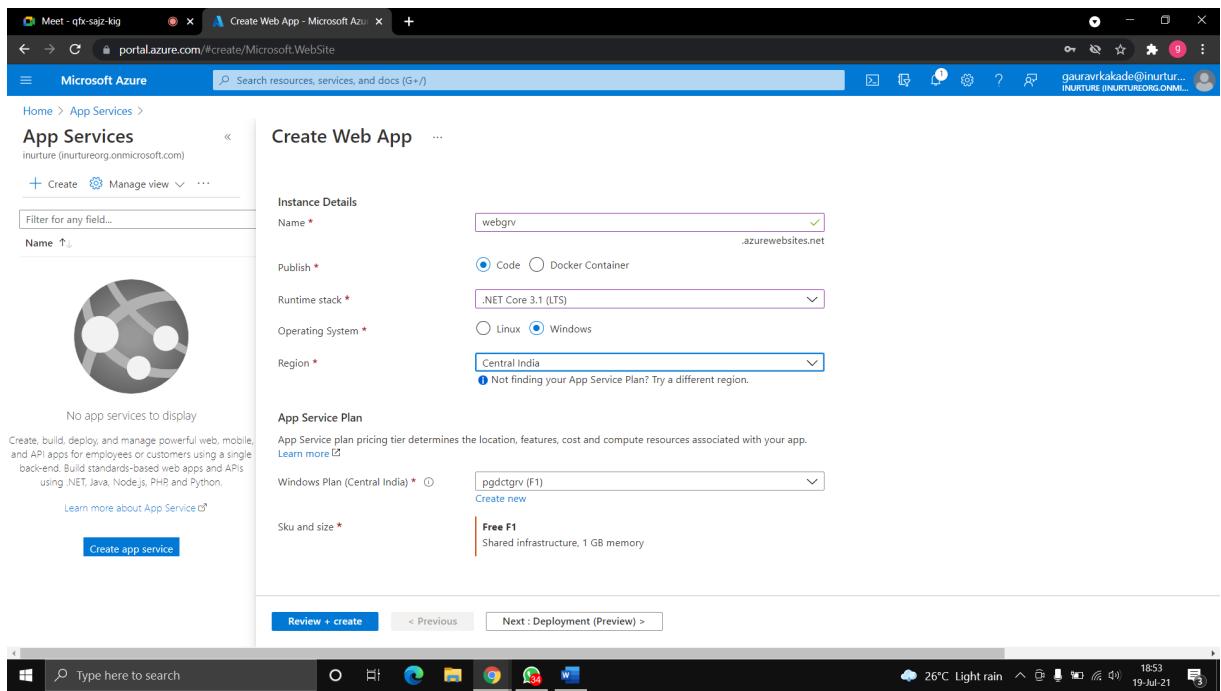


Fig: create web app

7.Keep the deployment setting as disable.

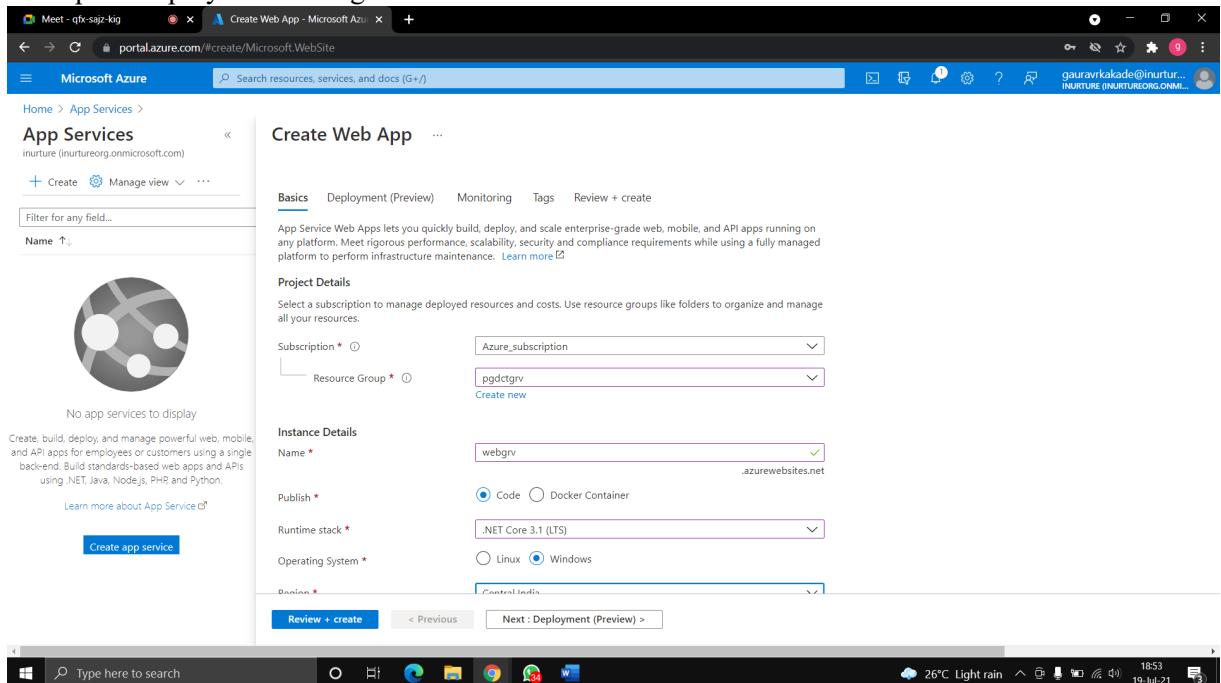


Fig: deployment settings

8.In monitoring keep application insights as no.

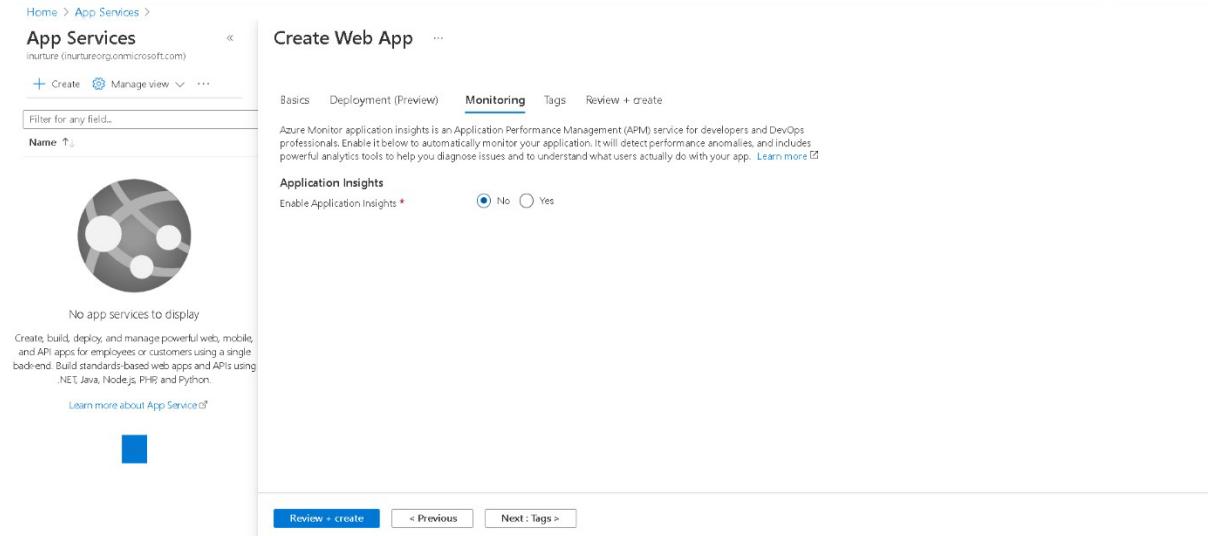


Fig: configure monitoring

9.Create the web app.

~ Steps to install visual studio and sink with azure:

- 1.Go to browser and download visual studio.
- 2.While installing you can select which features you want in your visual studio.
- 3.Open the visual studio>>create project.

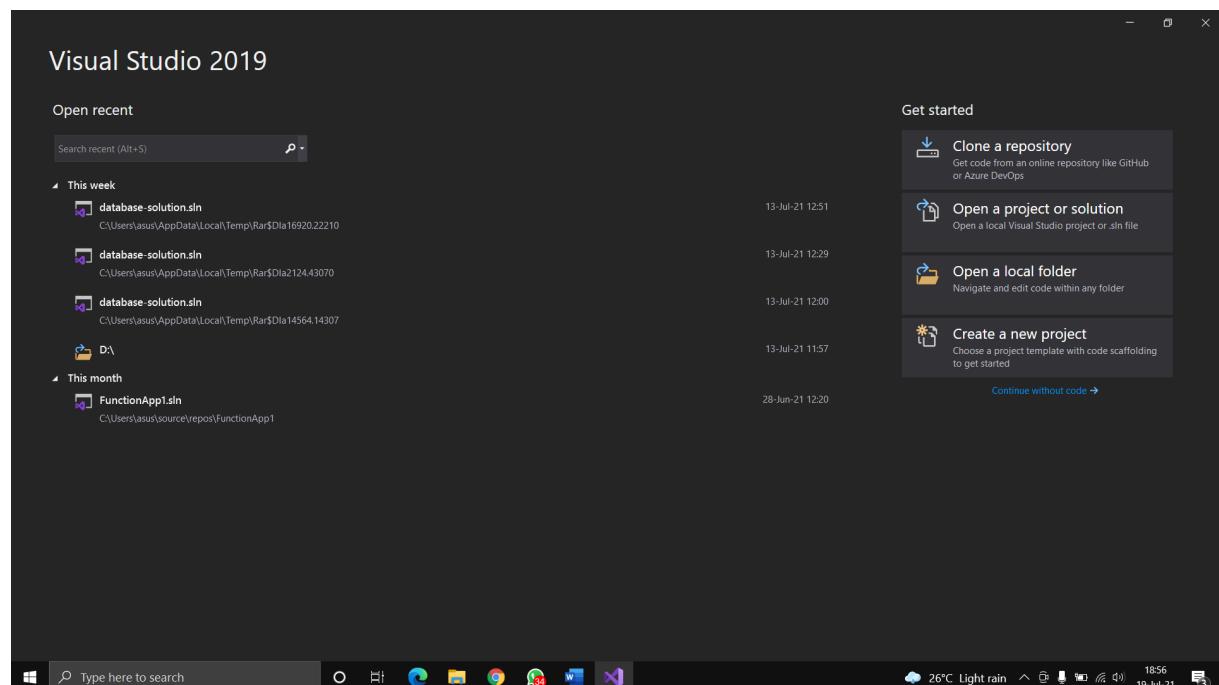


Fig: install VS and create new project

4.Select ASP.NET core web app.

5.Select target framework as .NET 3.1 and create the project.

6.Go to publish.

7.In target select azure and in specific target select azure app service(windows).

8.In app service>>sign in with your azure account.

9.After sign in>>select your resource group>>click on create.

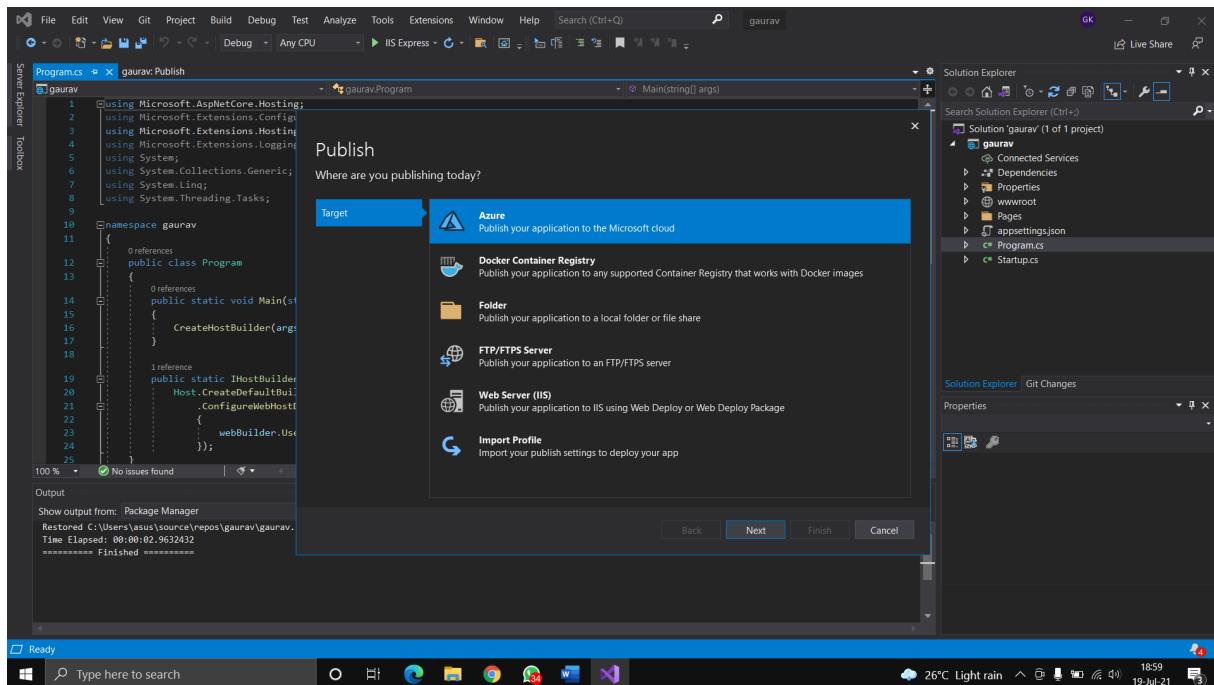


Fig: select your app service instance

~Steps to see app service logs:

1.Go to azure portal>>app service logs.

2.Go to web server logging>>storage.

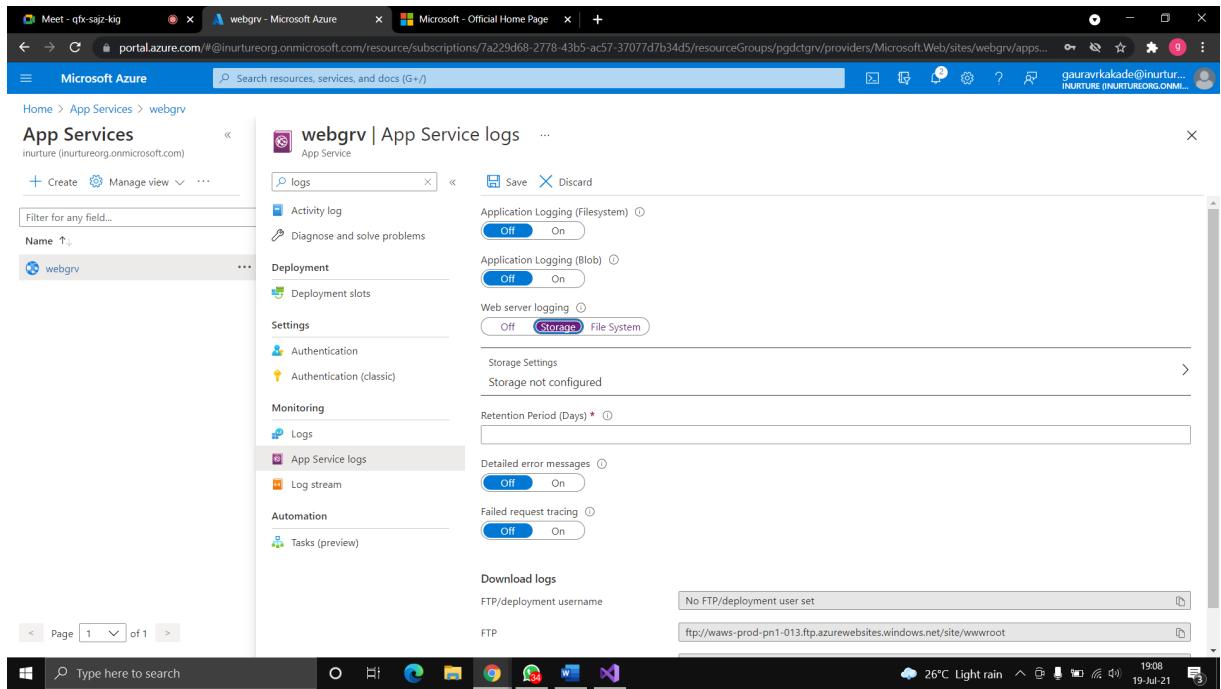


Fig: create app service logs

3.Create storage account>>create container.

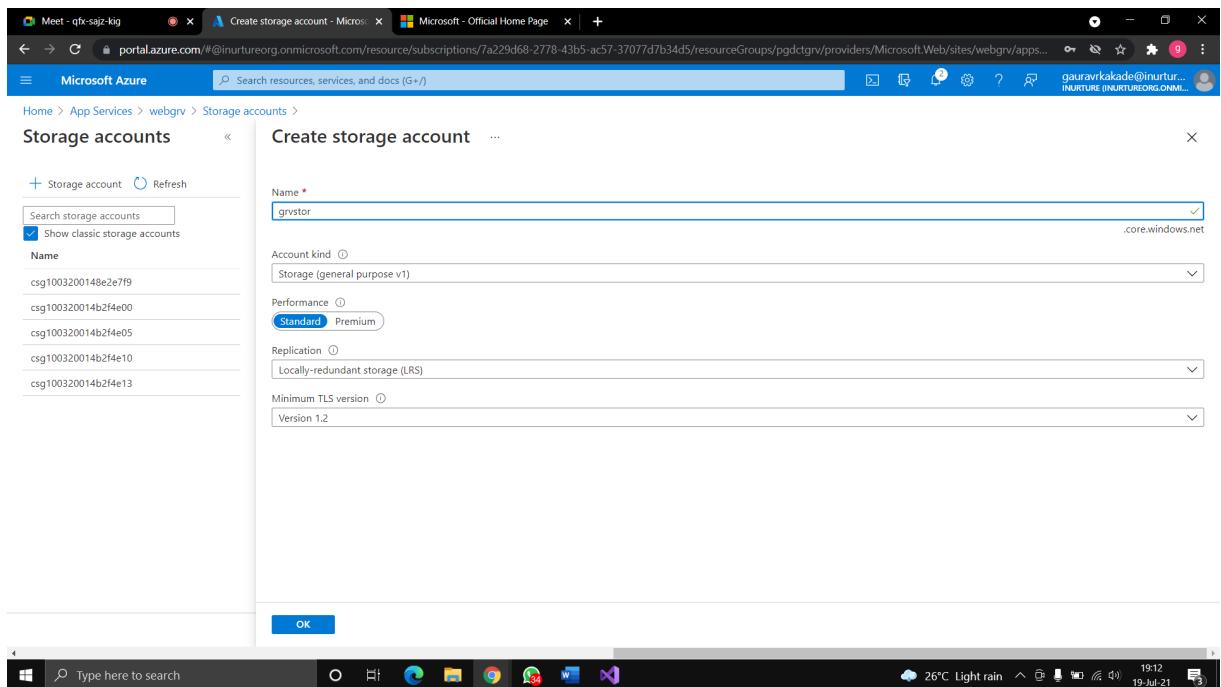


Fig: create storage account

4.Select the container and save.

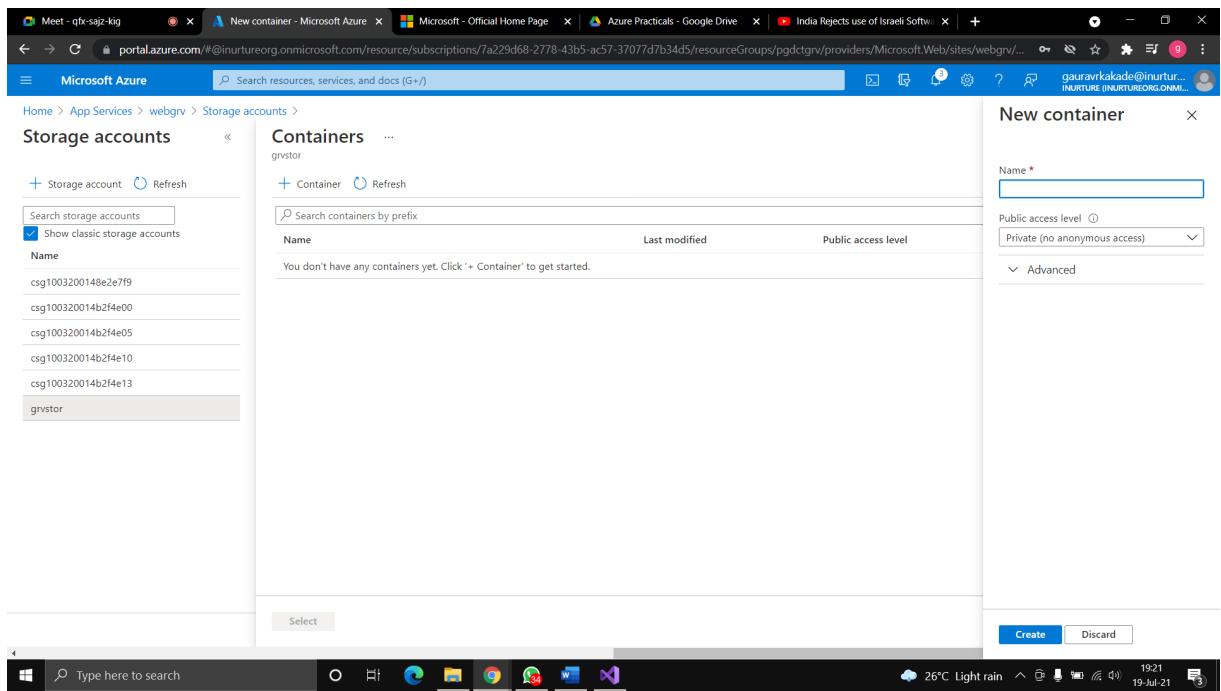


Fig: create new container

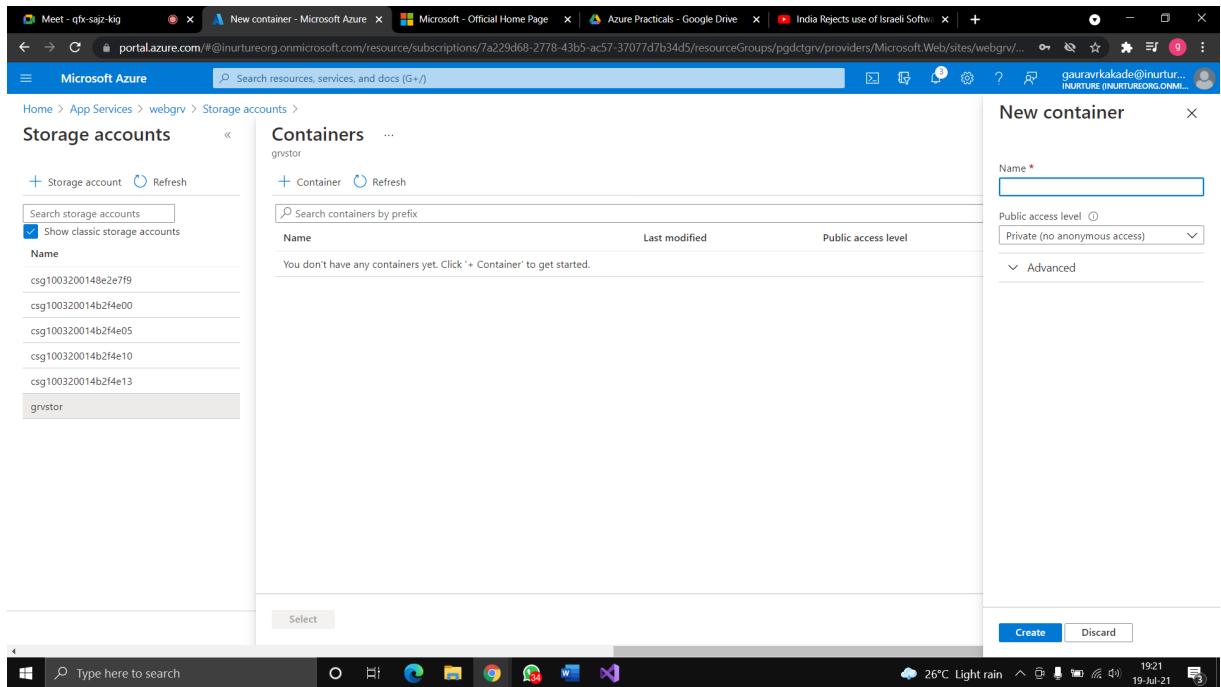


Fig: created container

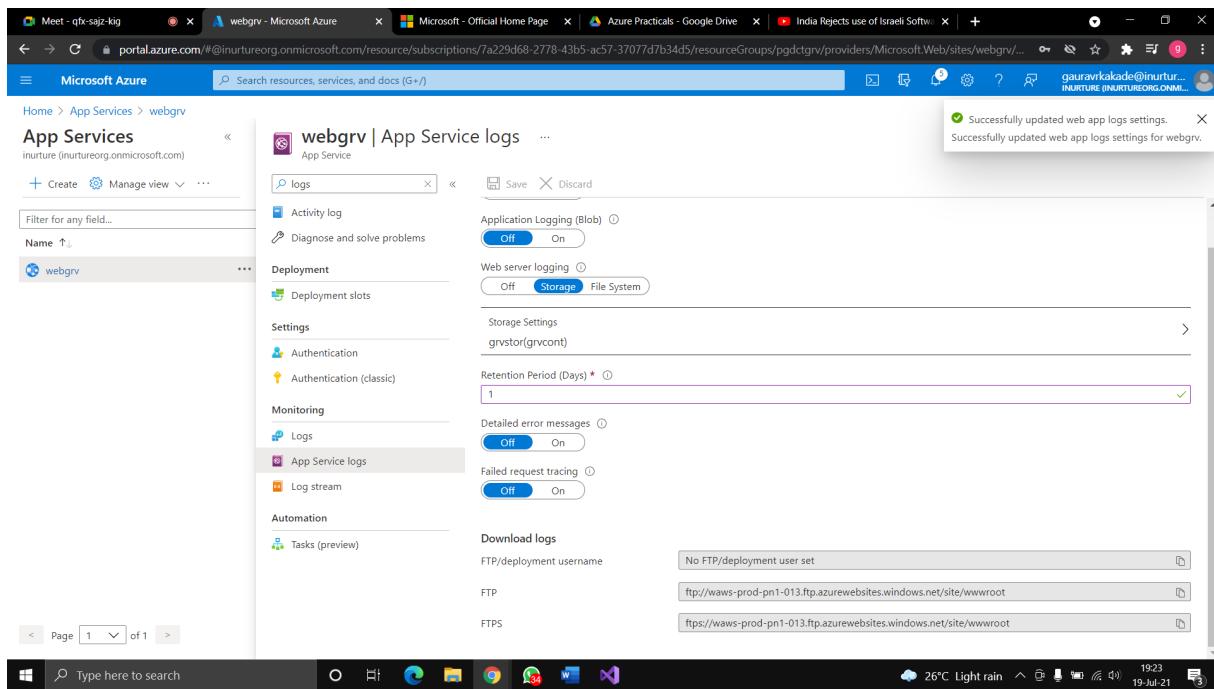


Fig: created container

5. Now go to storage>>container.

6. You will see one folder is created.

7. Open that folder>>you will see your logs.

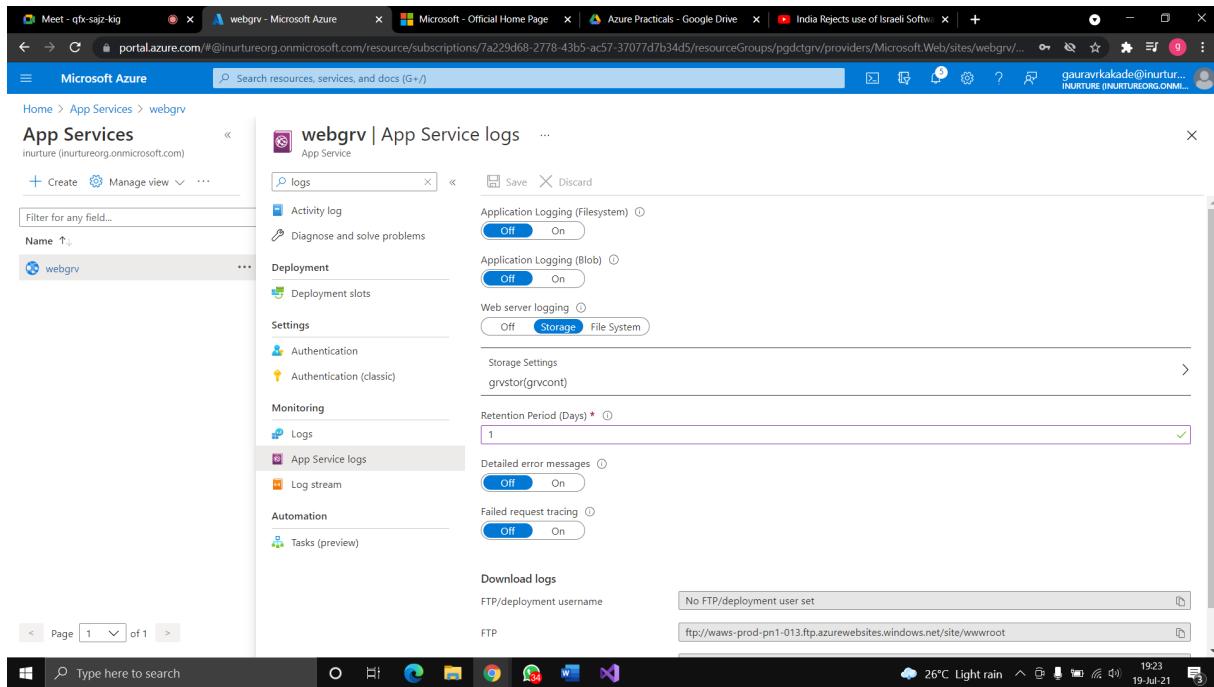


Fig: overview window

8.Go to overview>>copy the url and paste in browser.

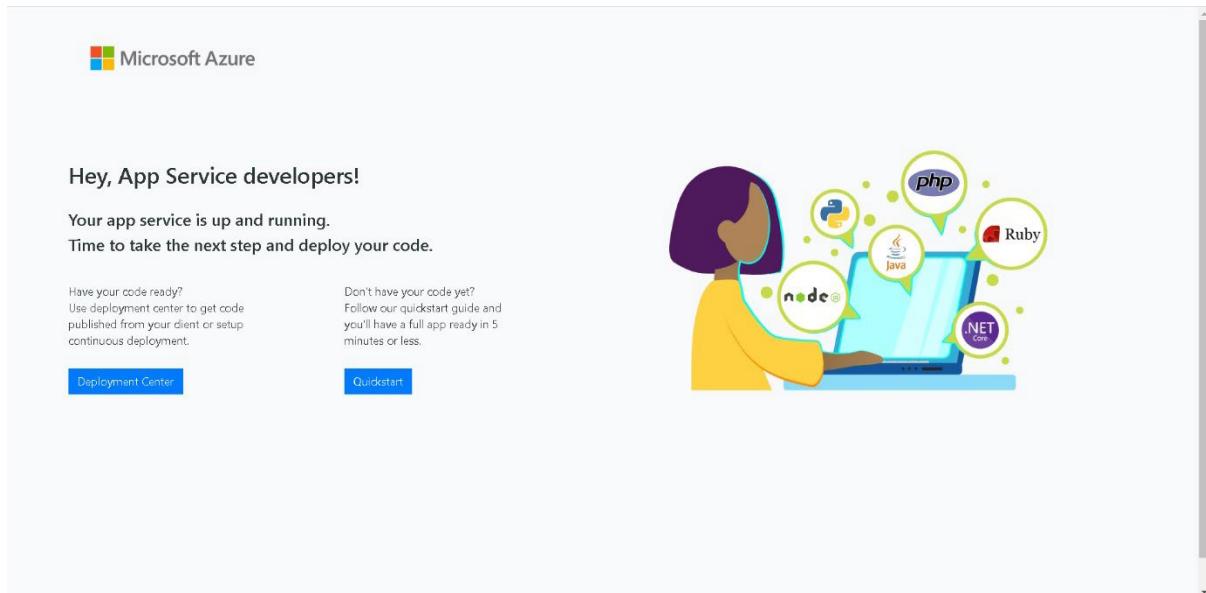
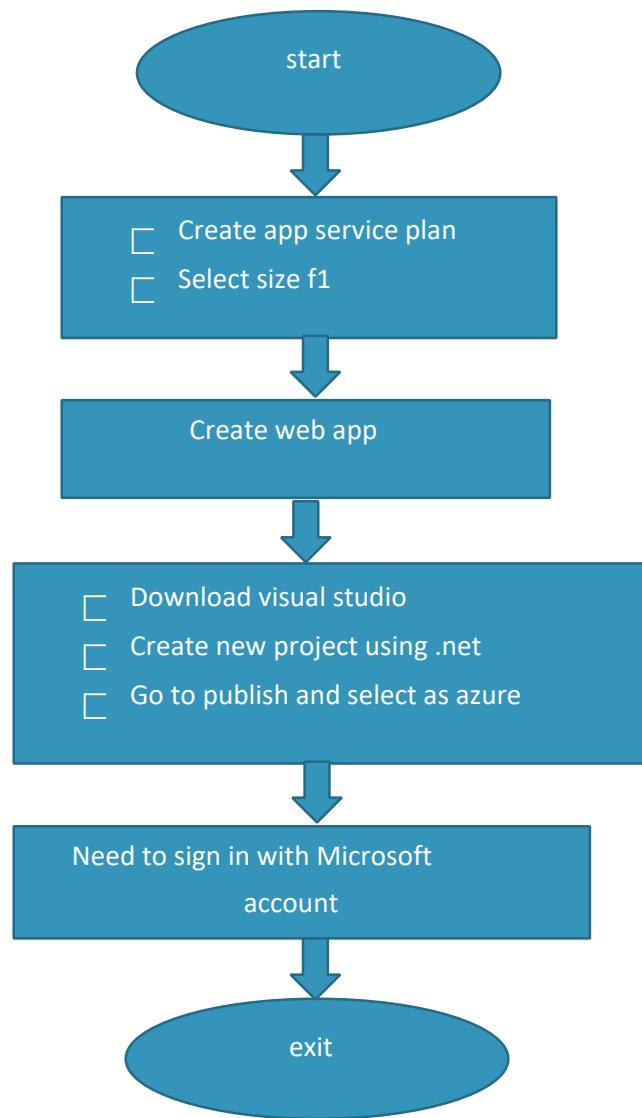
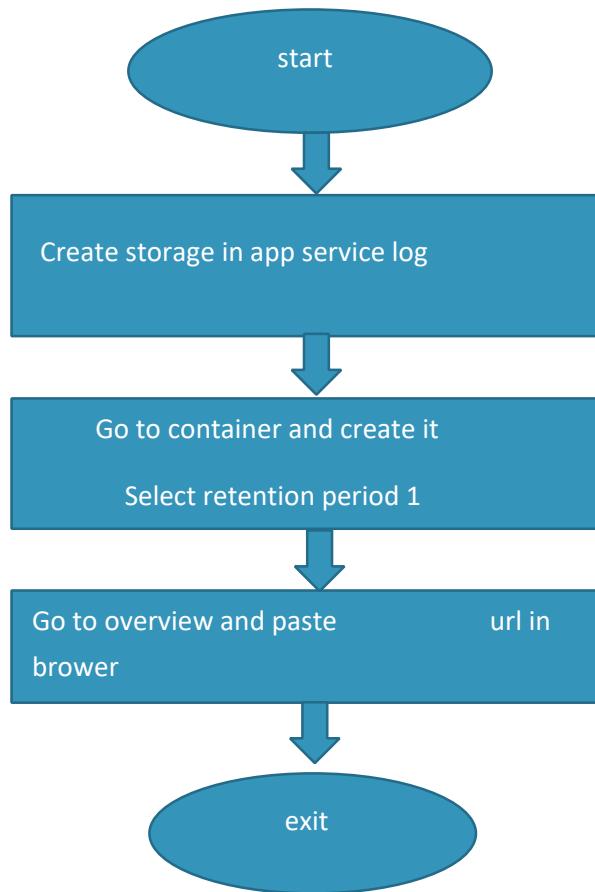


Fig: Page of AppService web

Flowchart:



Flowchart to see the logs:



Result: The web app has been created and the azure account has been sink with visual studio, where we can see our web app. The app service logs was created and can been seen in container.

Conclusion: The web app was successfully created and azure account was successfully sink with visual studio.

EXPERIMENT NO: 4

AIM: Create a web app instance and a virtual machine which contains database, and try to access the database through web app instance.

PREREQUISITES: Azure portal, visual studio, SQL server management studio.

DESCRIPTION:

Azure app Service plan: An App Service plan defines a set of compute resources for a web app to run. Azure Web Apps provides a platform to build an App in Azure without having to deploy, configure and maintain your own Azure VM's. You can build Web App using the ASP.NET, PHP, Node.js and Python. They also integrate common development environments which could be Visual Studio and GitHub.

Visual studio: Microsoft Visual Studio is an integrated development environment from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps.

SQL server management studio: SQL Server Management Studio is a software application first launched with Microsoft SQL Server 2005 that is used for configuring, managing, and administering all components within Microsoft SQL Server.

ALGORITHM:

1.Create a virtual network.

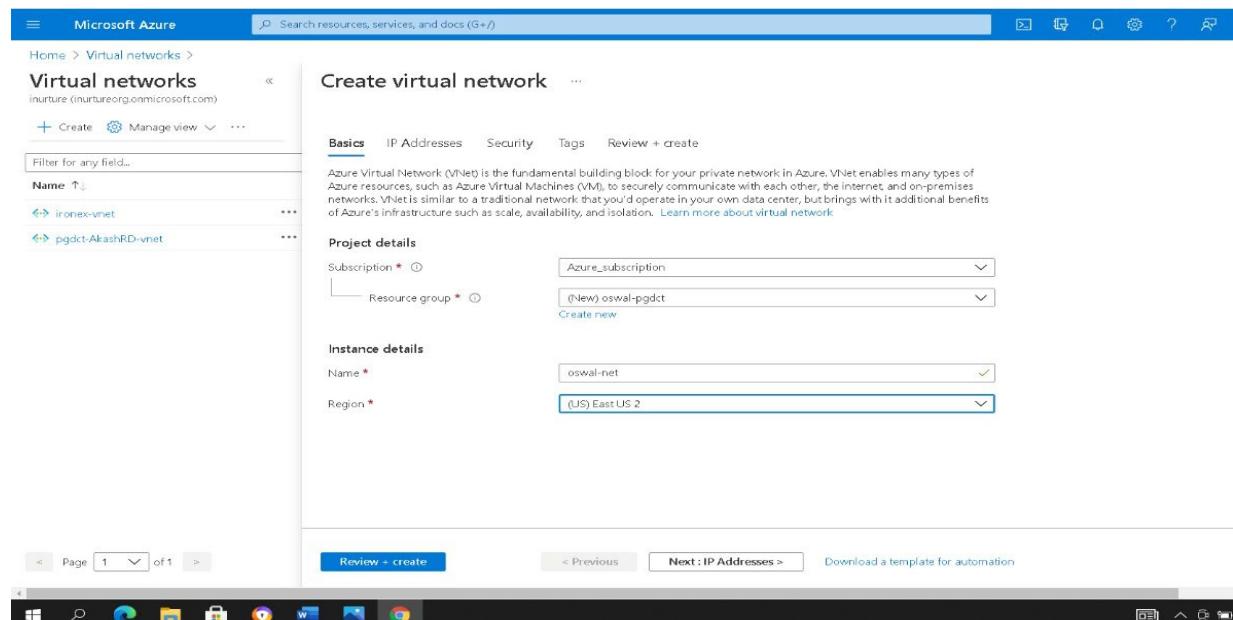


Fig: create virtual network 2. Go to app

service plan>>create a app service plan.

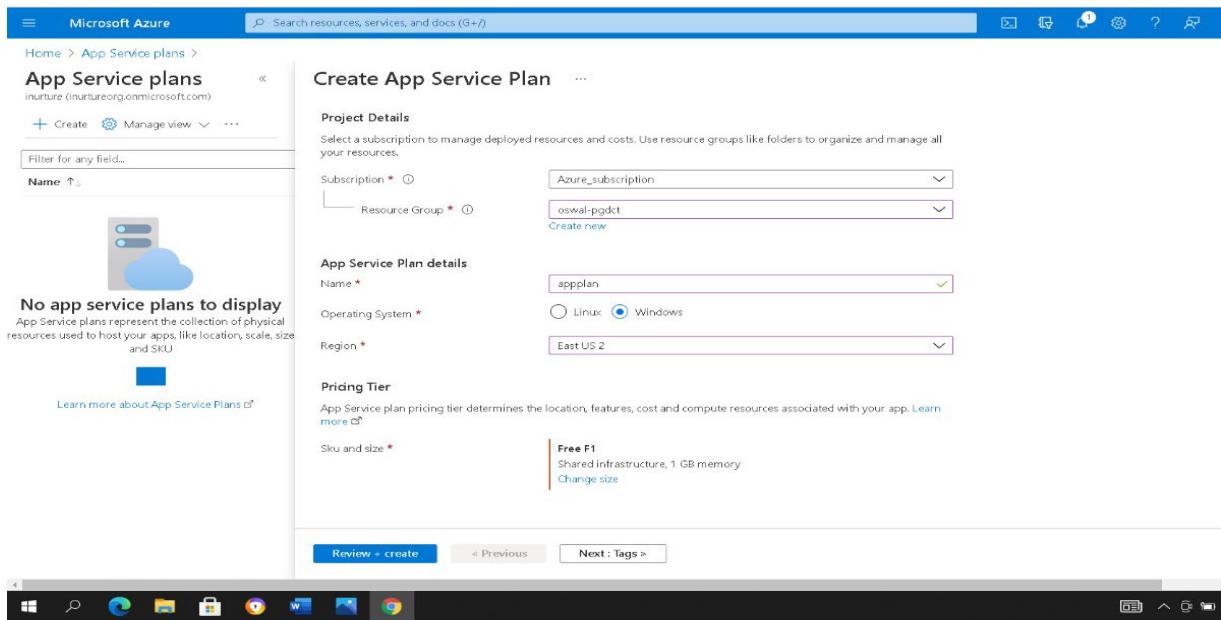


Fig: create app service plan

3.Go to resource of app service plan>>Networking>>go to VNet integration.

4.Choose the plan S1 and apply.

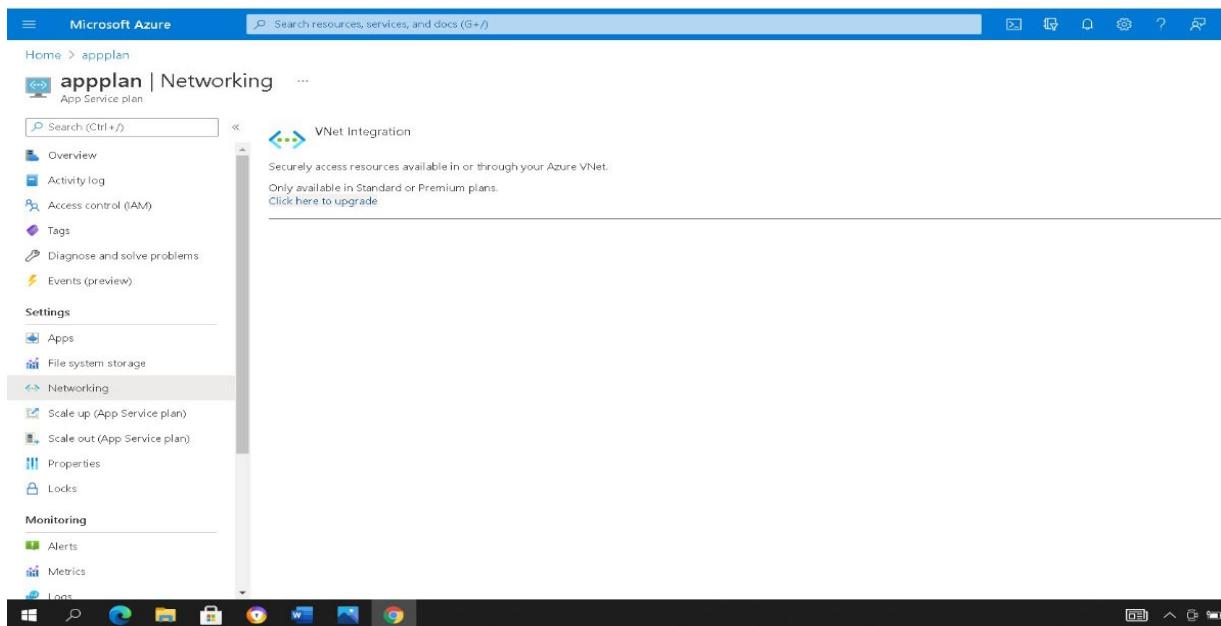


Fig: upgrade vnet integration

Spec Picker

See only recommended options

Additional pricing tiers

Tier	ACU	Memory	Price (INR/Month)
S1	100 total ACU 1.75 GB memory A-Series compute equivalent	1.75 GB memory 4 vCPU	5259.30 INR/Month (Estimated)
S2	200 total ACU 3.5 GB memory A-Series compute equivalent	3.5 GB memory 8 vCPU	10518.61 INR/Month (Estimated)
S3	400 total ACU 7 GB memory A-Series compute equivalent	7 GB memory 16 vCPU	21037.22 INR/Month (Estimated)
P1	100 total ACU 1.75 GB memory A-Series compute equivalent	1.75 GB memory 4 vCPU	15777.91 INR/Month (Estimated)

Included features
Every app hosted on this App Service plan will have access to these features:

- Custom domains / SSL

Included hardware
Every instance of your App Service plan will include the following hardware configuration:

- Azure Compute Units (ACU)

Apply

Fig: select S1 pricing tiers

5.Go to app service>>click on create a web app.

6.Give the name of instance>>choose runtime as **ASP.NET V4.8**.

Create Web App

Instance Details

Name * oswal-web

Publish * Code

Runtime stack * ASP.NET V4.8

Operating System * Windows

Region * East US 2

App Service Plan

Windows Plan (East US 2) * appplan (F1)

Sku and size * Free F1 Shared infrastructure, 1 GB memory

Review + create < Previous Next : Deployment (Preview) >

Fig: create web app

7.Keep enable application insights as no.

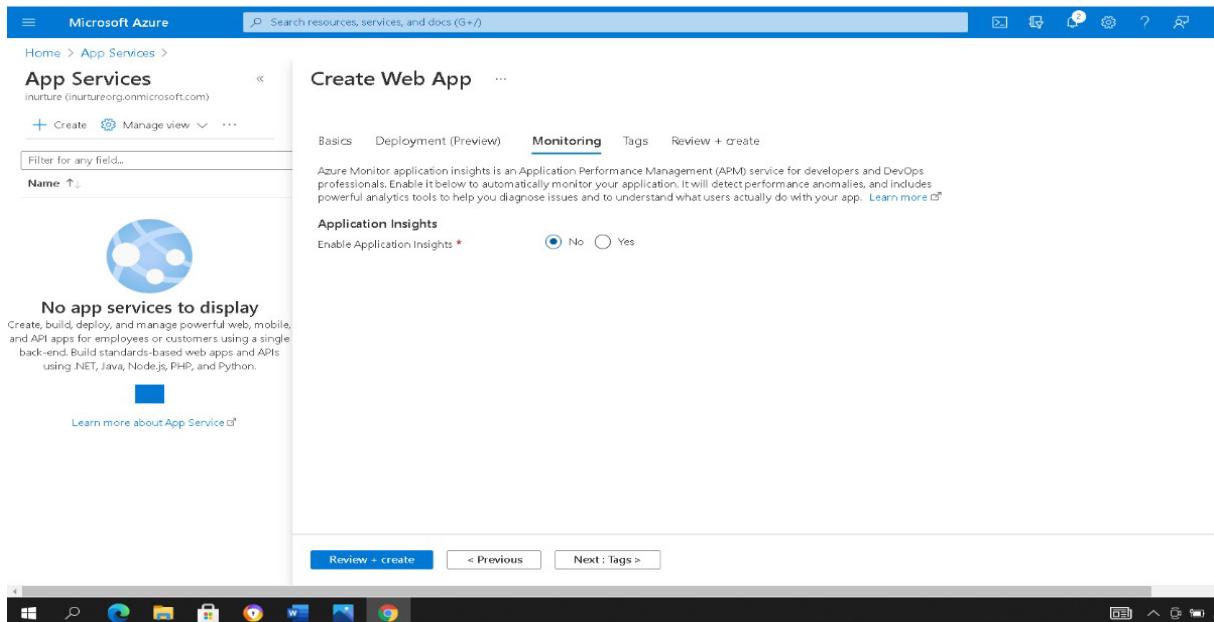


Fig: configure monitoring

8.Create web app.

9.Go to resource>>Networking>>go to VNet integration.

10.Click on add VNet>>select your virtual network and click on ok.

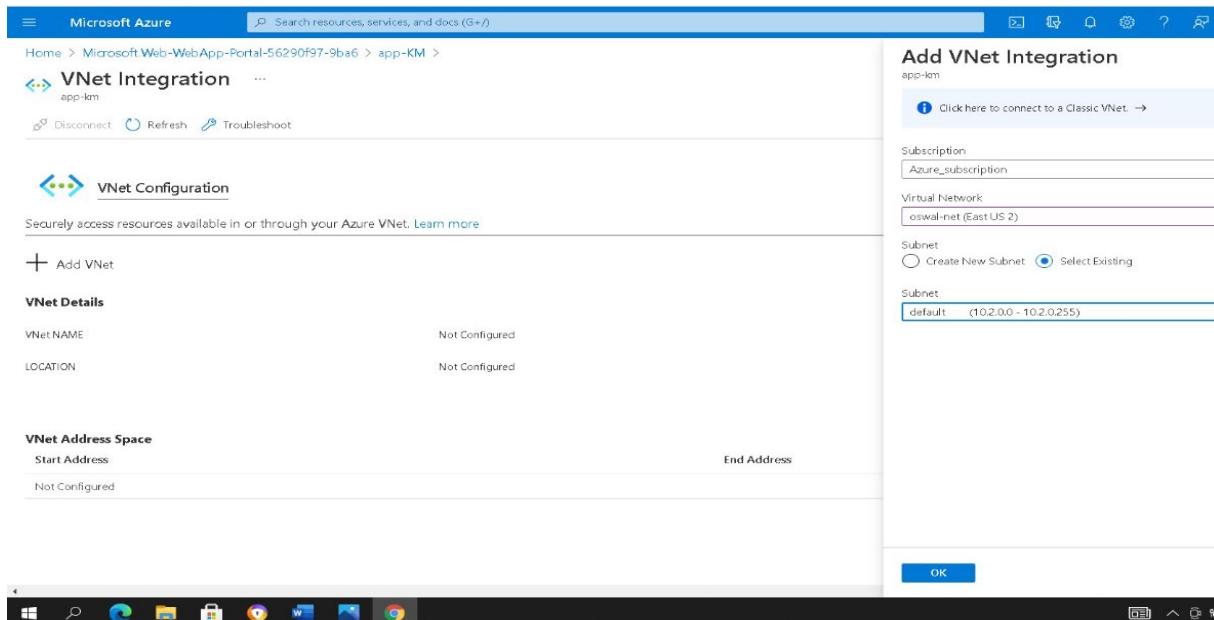


Fig: Add vnet integration

11.Now create a virtual machine with the image of **SQL server 2019 web on windows server 2019-Gen1**.

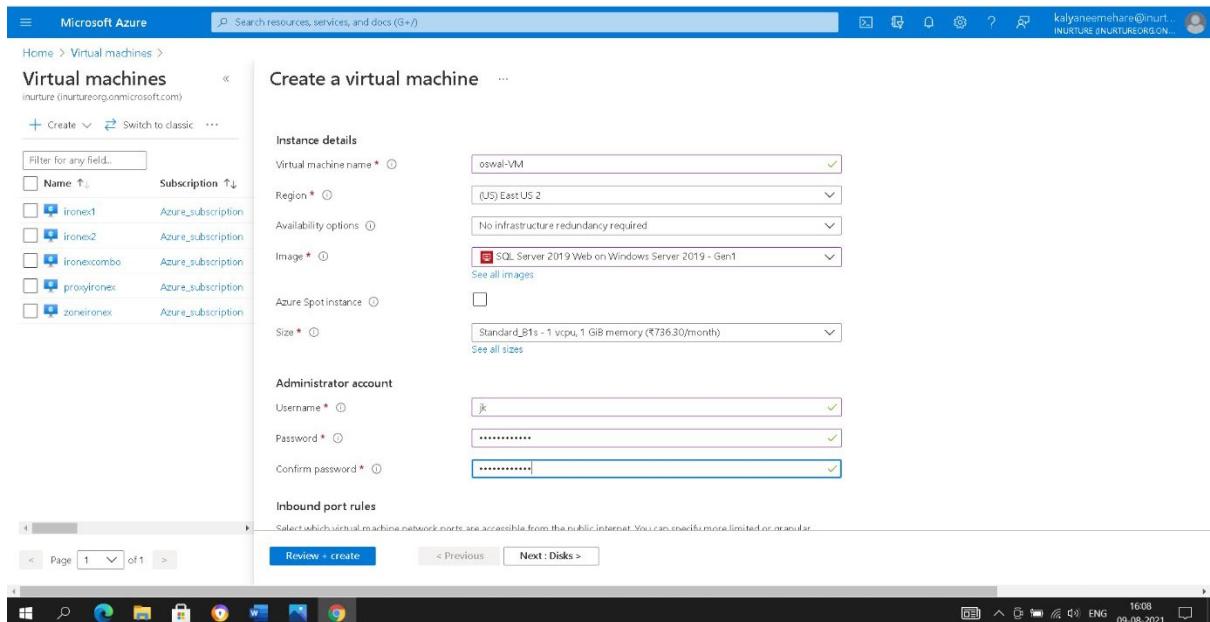


Fig: create virtual machine

12.In networking>>go to subnet by selecting manage configuration and create a new subnet.

Name	IPv4	IPv6	Available IPs	Delegated to
default	10.2.0.0/24	-	251	MicrosoftWeb/serverfarms
mysub	10.2.1.0/24	-	251	-

Fig: create subnet

13.In SQL server settings>>keep the SQL connectivity as public and enable SQL authentication, enter the password and username.

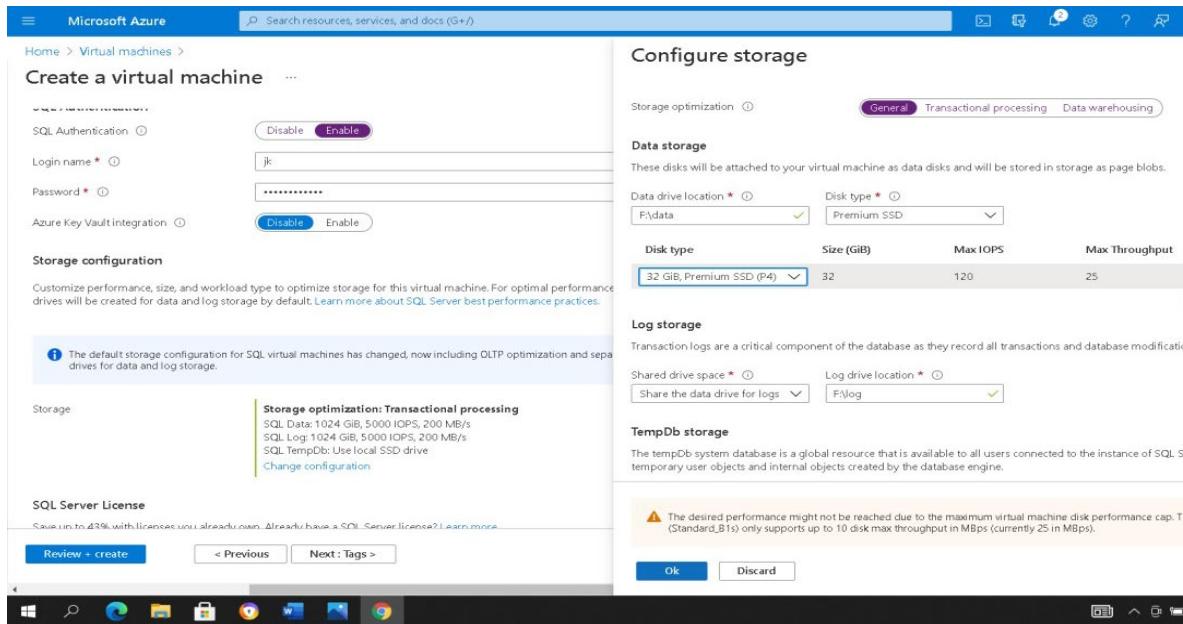


Fig: configure storage

14.Create the virtual machine.

15.Open the SQL server management studio, In server name enter your virtual machine IP address and choose SQL authentication, enter username and password and click on connect.

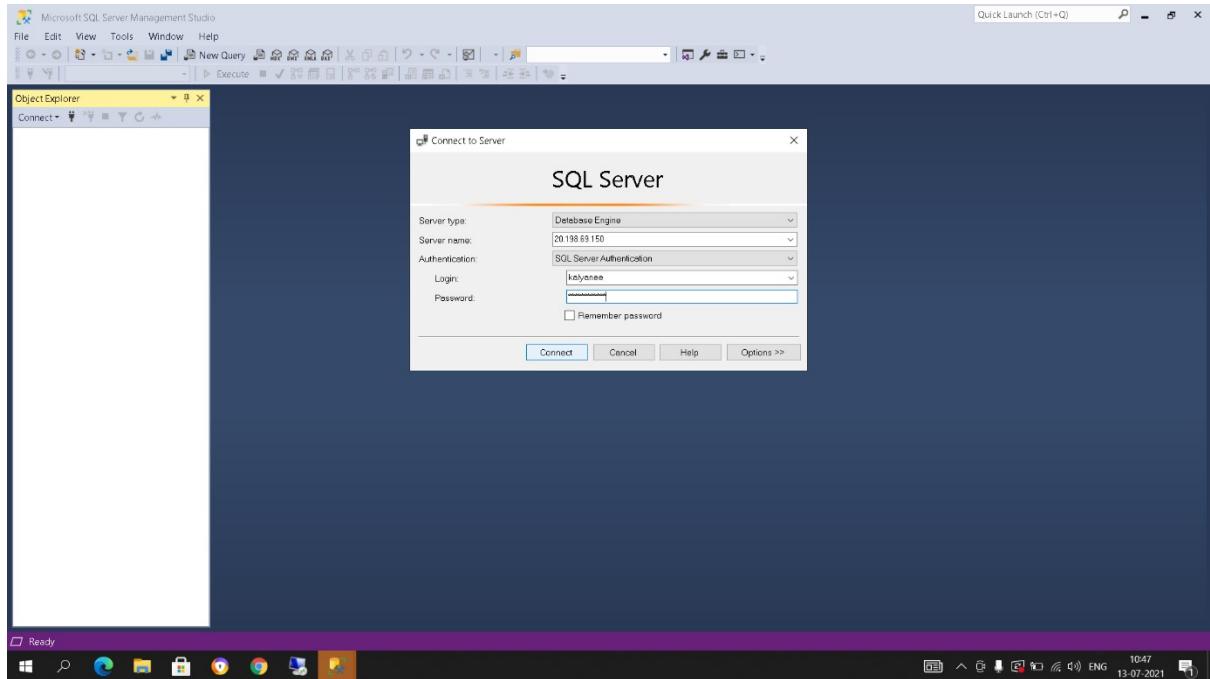


Fig: give credentials

16.Create new database and create a table.

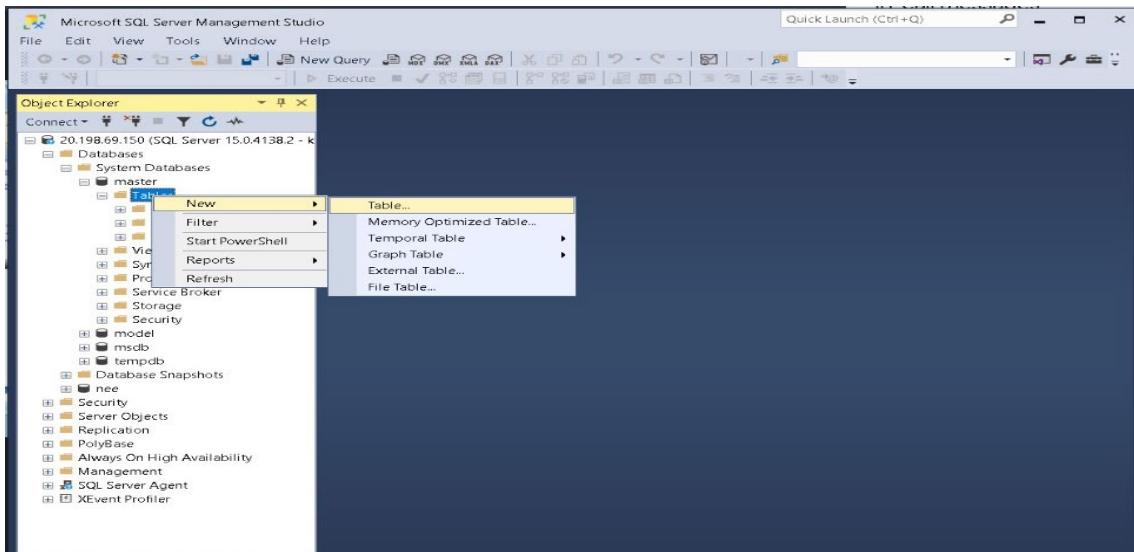


Fig: Add new tables

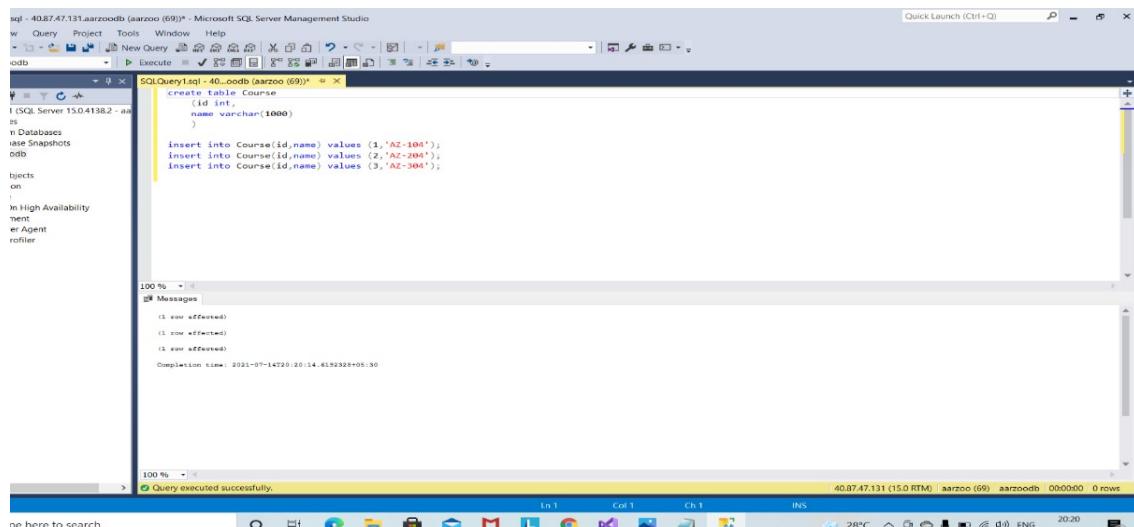


Fig: created table

17.Go to virtual machine>>networking>>go to public IP address and disassociate the public IP.

18. Open the database project in visual studio, go to webconfig file edit the code and enter the proper credentials of your virtual machine in that code.

19 Now publish and sink with azure account, select your resource group and click on finish.

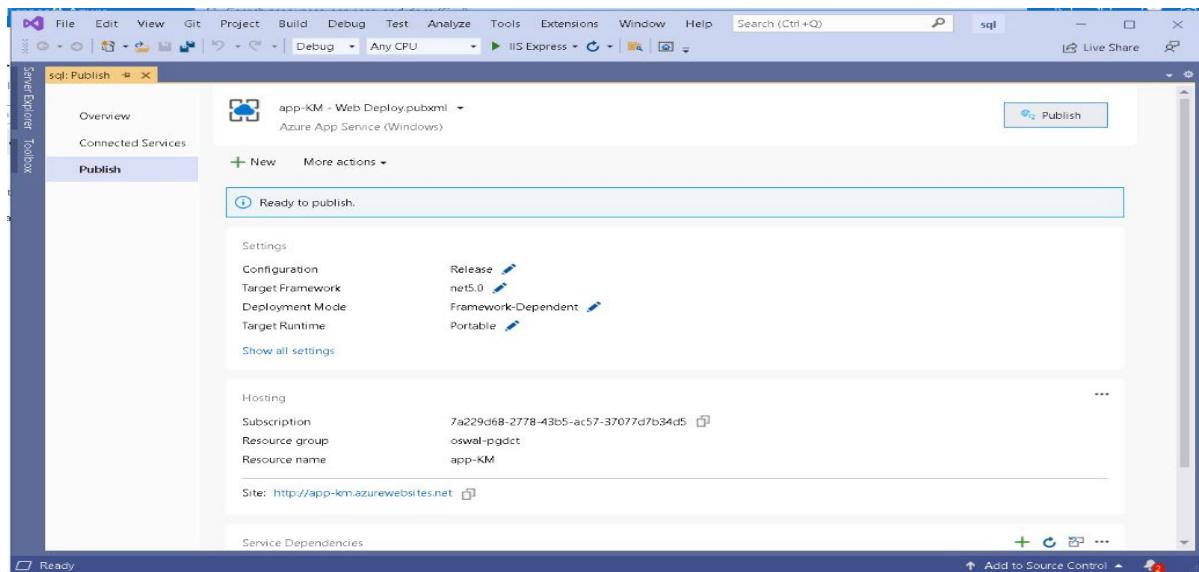


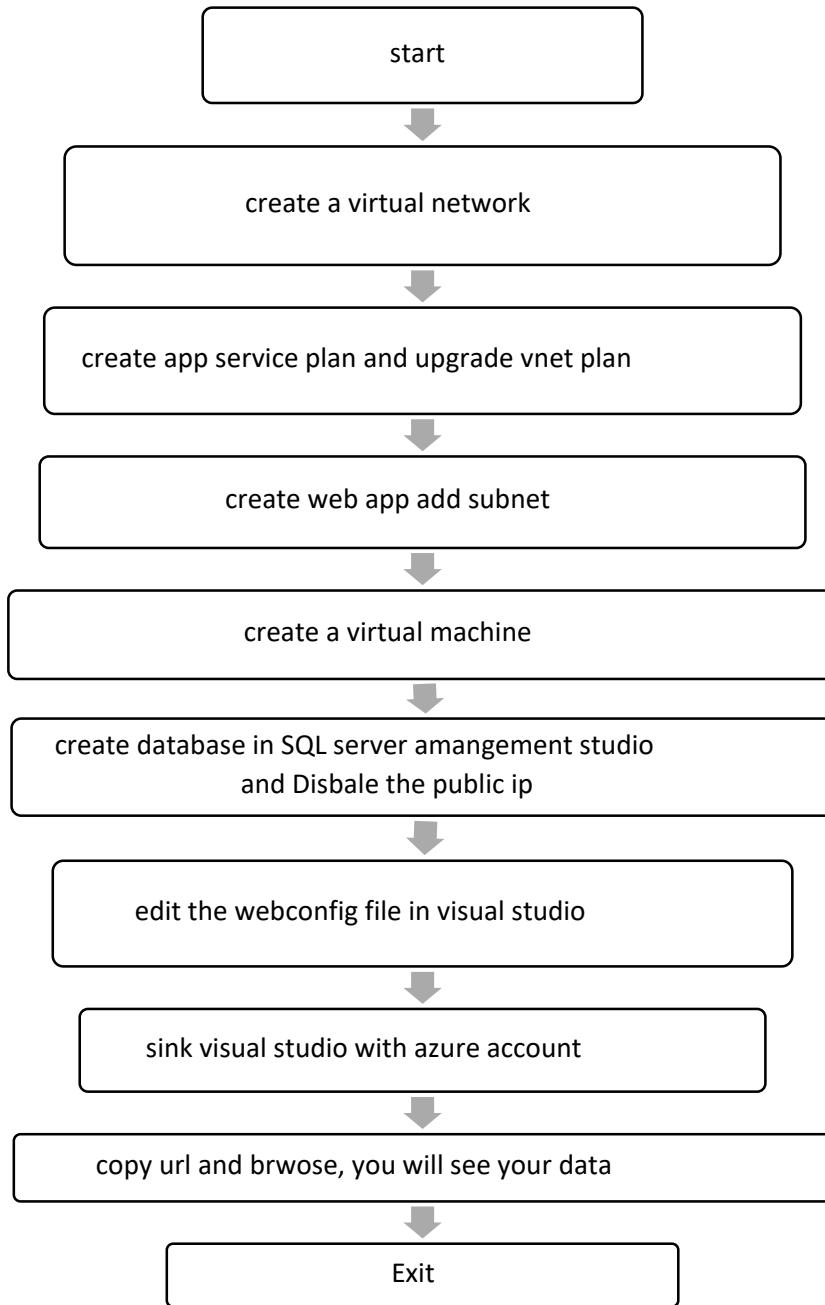
Fig: published

20. Now copy the URL and browse it, you will see your data.

A screenshot of a web browser showing the output of a database table. The page has a header with links for 'Application name', 'Home', 'About', and 'Contact'. Below the header is a table titled 'Index' with columns 'id' and 'name'. The table contains three rows with data: (1, AZ-104), (2, AZ-204), and (3, AZ-304). At the bottom of the page is a copyright notice: '© 2021 - My ASP.NET Application'.

Fig: output of created table

Flowchart :



Result: We have created the web app instance and a virtual machine in which we have our database, we have seen that how web app has accessed our database.

Conclusion: we have successfully accessed the database through web app.

EXPERIMENT NO: 5

AIM:Creation of Virtual Machine, installation of Docker Engine, Image Management, Launching container

PREREQUISITES:Azure portal **DESCRIPTION:**

Docker:

Docker is a set of platforms as a service (PaaS) product that use OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries and configuration files; they can communicate with each other through well-defined channels. With Docker, you can manage your infrastructure in the same ways you manage your applications.

Container instance:

Azure Container Instances is a solution for any scenario that can operate in isolated containers, without orchestration. Run event-driven applications, quickly deploy from your container development pipelines, and run data processing and build jobs

ALGORITHM:

1. First, we need to create virtual machine.
2. Go to create virtual machine
3. create resource group>>give it a name and select region.
4. Select image ubuntu server 18.0-LTS -gen1>> password as authentication type
5. Give username and password>>open http and rdp ports

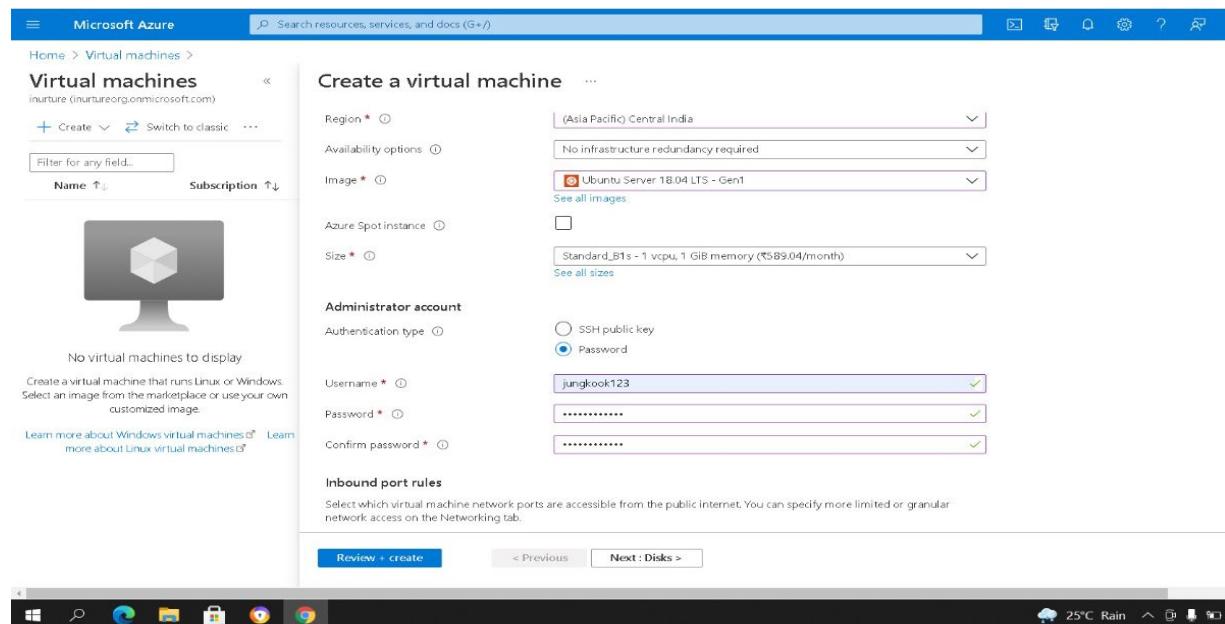


Fig: create virtual machine

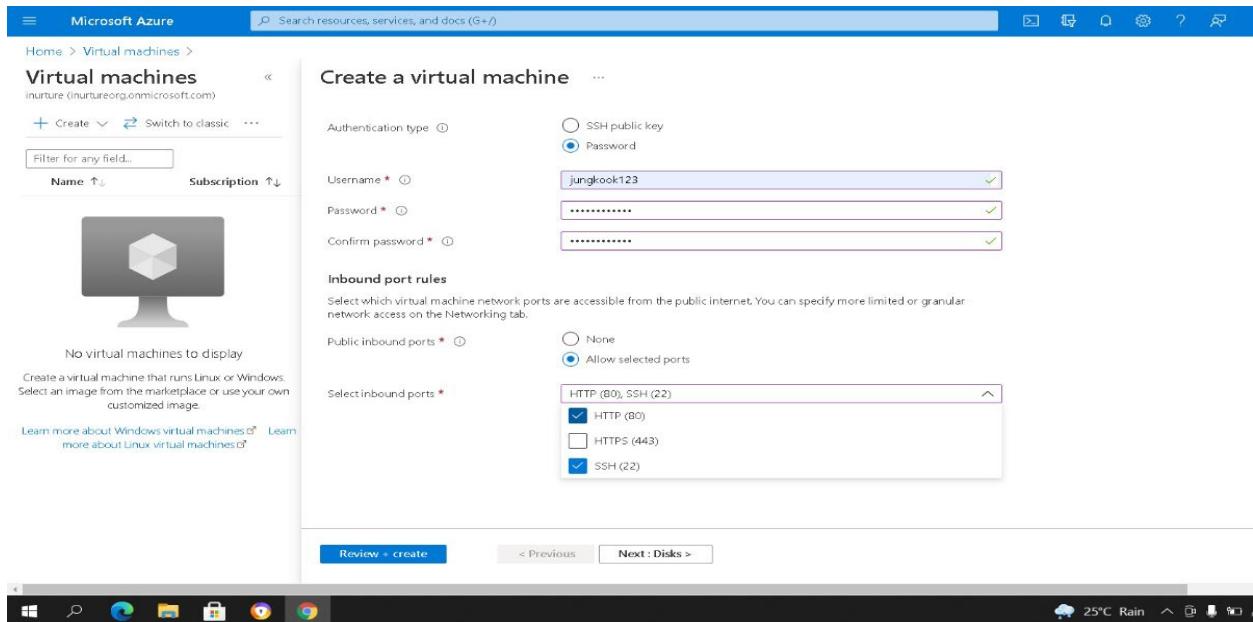


Fig: open port

6. Select os disk type standard HDD and everything set as a default and create virtual machine.

Properties	Value
Computer name	vm-doc
Operating system	Linux (Ubuntu 18.04)
Publisher	Canonical
Offer	UbuntuServer
Plan	18.04-LTS
VM generation	V1
Agent status	Ready
Agent version	2.31.1
Host group	None
Host	-

Networking	Value
Public IP address	52.172.255.67
Public IP address (IPv6)	-
Private IP address	10.0.0.4
Private IP address (IPv6)	-
Virtual network/subnet	pgdct-kalyanee-vnet/default
DNS name	Configure

Size	Value
Size	Standard B1s
vCPUs	1
RAM	1 GB

Fig: connect with ssh

7. After creation just connect vm to ssh

8. Open putty and add the username and password in it.

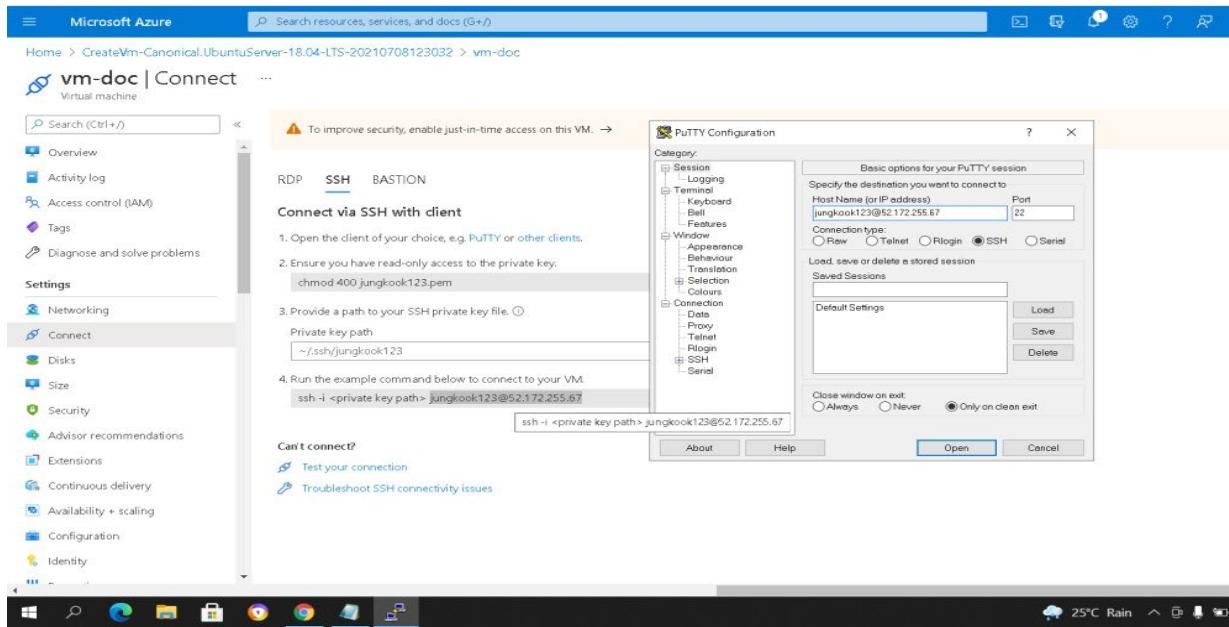


Fig: enter username and ip

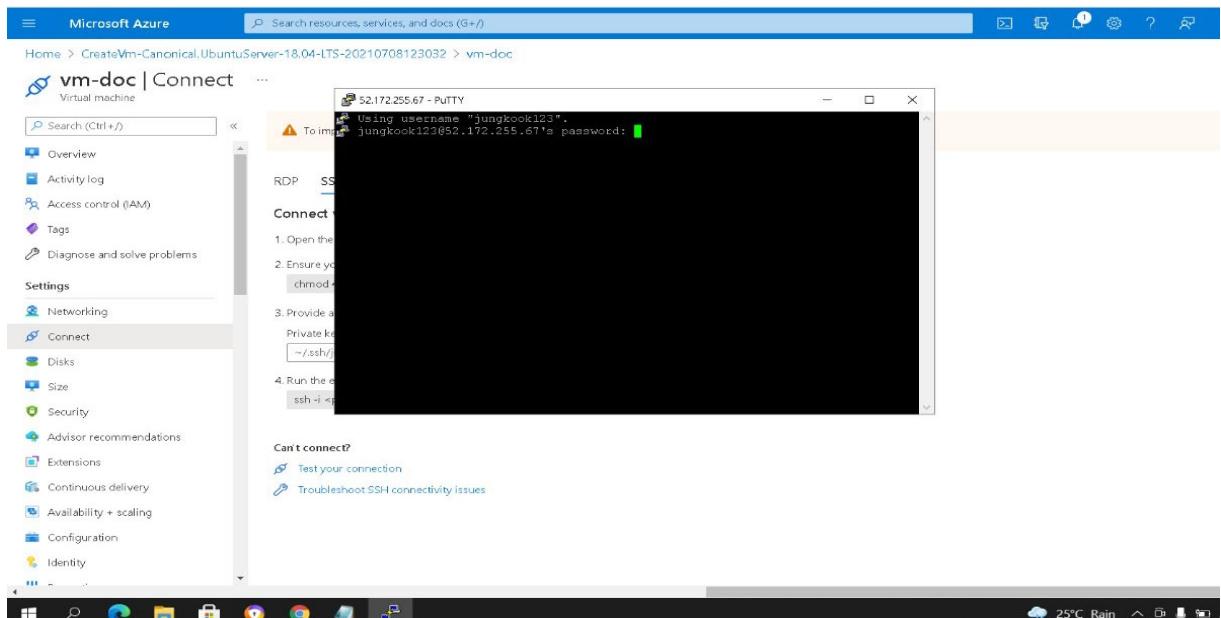


Fig: connected putty

9.after connecting with putty just enter your password and enter following commands.

10.sudo apt-get update

11.it will update packages

```

jungkook123@vm-doc: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

jungkook123@vm-doc:~$ sudo apt-get update
Hit:1 http://azure.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Hit:3 http://azure.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://azure.archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [857
1. Open the
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:6 http://azure.archive.ubuntu.com/ubuntu bionic/universe Translation-en [494
2. Ensure y
  chmod Get:7 http://azure.archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [1
  51 kB]
Get:8 http://azure.archive.ubuntu.com/ubuntu bionic/multiverse Translation-en [1
  51 kB]
3. Provide a
  Private key Get:9 http://azure.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages
  [2131 kB]
  -/ssh/Get:10 http://azure.archive.ubuntu.com/ubuntu bionic-updates/main Translation-en
  [422 kB]
4. Run the
  Get:11 http://azure.archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Pa
  ssh: i-> Get:12 http://azure.archive.ubuntu.com/ubuntu bionic-updates/restricted Translat
  packages [389 kB]
  -/ssh/Get:13 http://azure.archive.ubuntu.com/ubuntu bionic-updates/restricted Translation-en
  [18 kB]

```

Can't connect?

- Test your connection
- Troubleshoot SSH connectivity issues

Fig: used commands

12.Install different packages for docker

13.sudo apt install apt-transport-https ca-certificates curl software-properties-common

```
~curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add
```

```
~sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic
stable"
```

```

jungkook123@vm-doc: ~
The following NEW packages will be installed:
  apt-transport-https ca-certificates curl software-properties-common
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 4348 B of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://azure.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 apt-t
ransport-https all 1:6.14 [4348 B]
Fetched 4348 B in 0s (122 kB/s)
Selecting previously unselected package apt-transport-https.
Reading database... 78947
Preparing to unpack .../apt-transport-https_1:6.14_all.deb ...
Unpacking apt-transport-https (1:6.14)...
Setting up apt-transport-https (1:6.14)...
jungkook123@vm-doc:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg |
  sudo apt-key add -
Private key Jungkook123@vm-doc:~$ sudo add-apt-repository "deb [arch=amd64] https://download
.docke
r.com/linux/ubuntu bionic stable
Hit:1 http://azure.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu bionic-backports InRelease
Get:4 https://download.docker.com/linux/ubuntu bionic InRelease [64.4 kB]
Get:5 https://download.docker.com/linux/ubuntu bionic/stable amd64 Packages [18.4 kB]
```

Can't connect?

- Test your connection
- Troubleshoot SSH connectivity issues

Fig: used commands

~sudo apt-get update

~Command for installing docker

~sudo apt-get install docker-ce

~command pull image for docker

~sudo docker pull nginx:1.17.0

~this command will create container with name sampleapp

~sudo docker run --name sampleapp -p 80:80 -d nginx:1.17.0

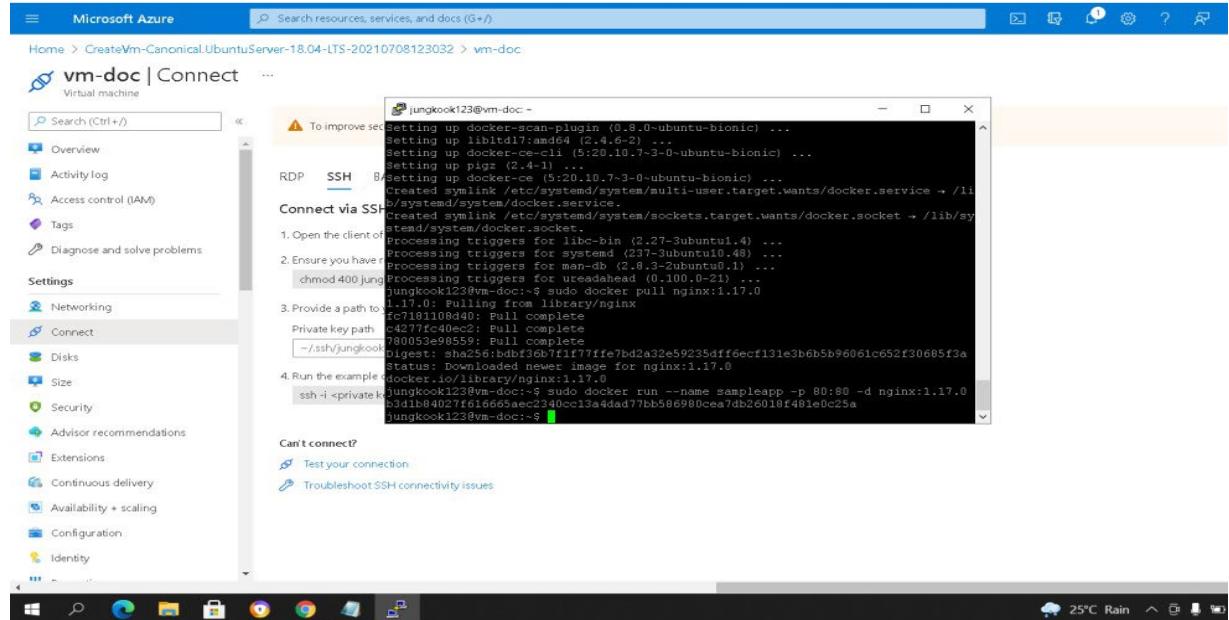
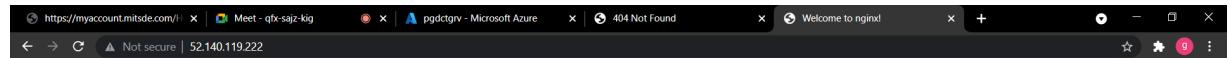


Fig: used commands

14.Then just grab public ip and browse it .

15.You will see the page of nginx server



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.



Fig: webpage of nginx server

16.Create Container instances

17.Search container instances in search bar

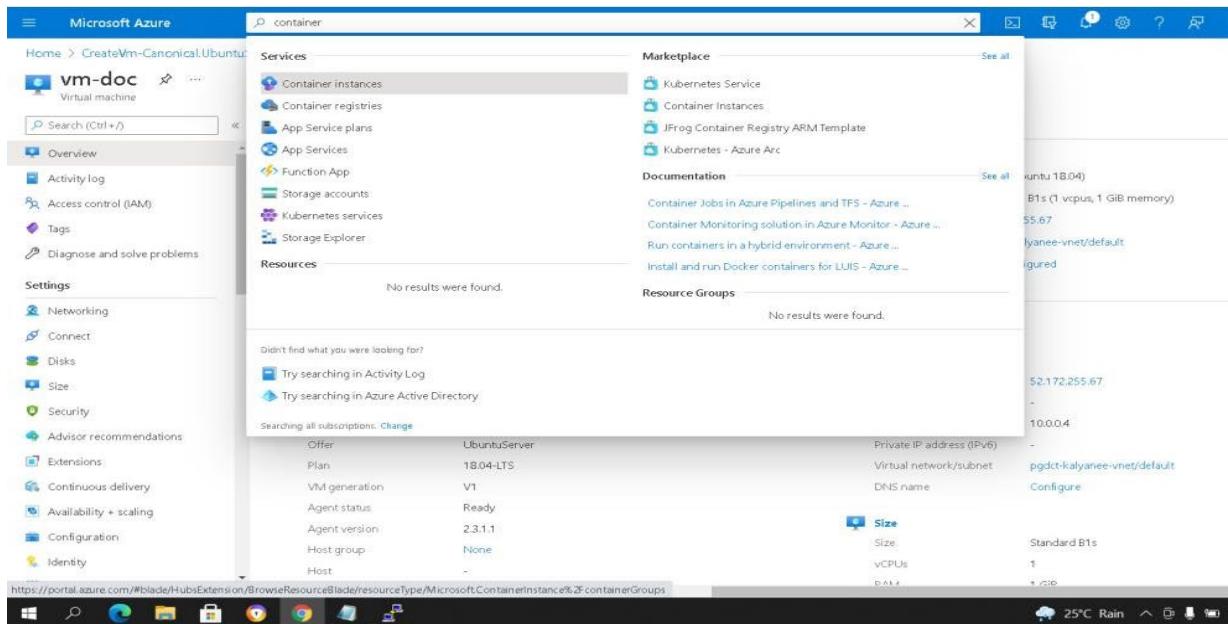


Fig: search container instance

18.Add resource group>>container name>>region

19.Select image source as docker hub

20.Image type public>>image nginx>>os type-Linux

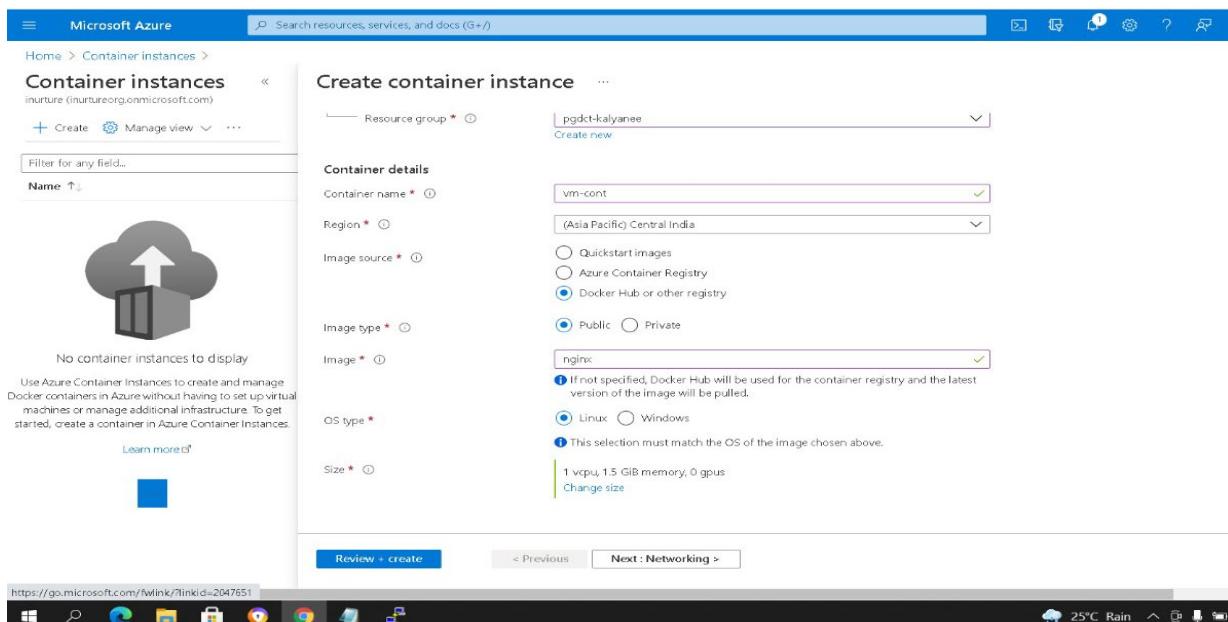


Fig: configure container instance

21.And set everything as default.

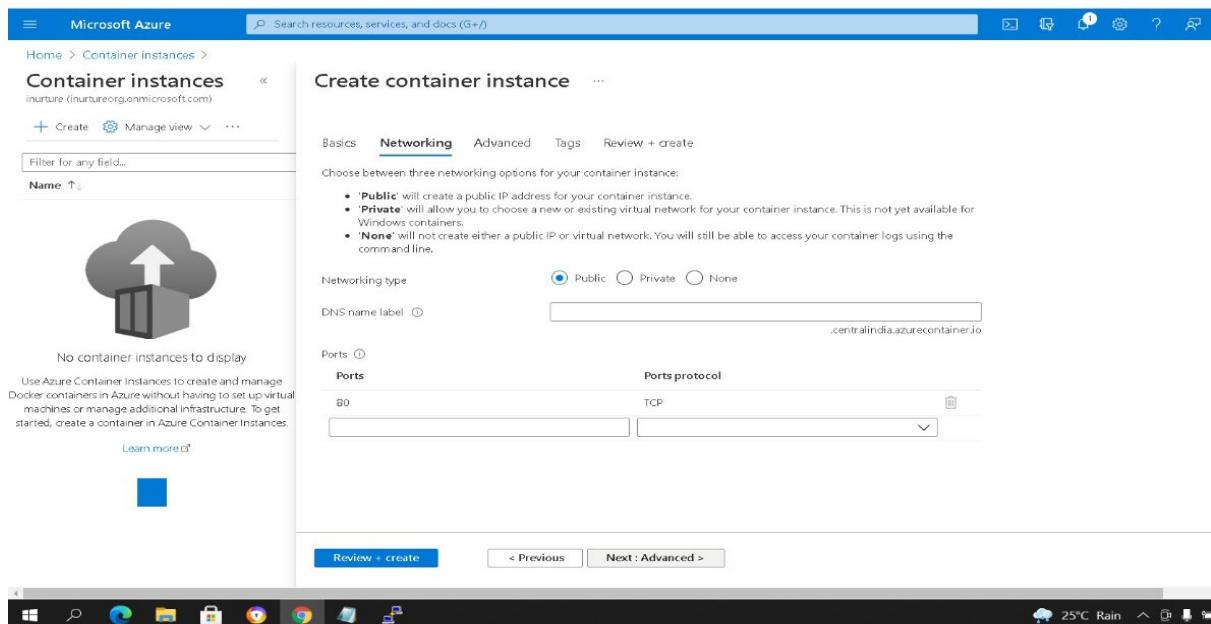


Fig: choose networking type

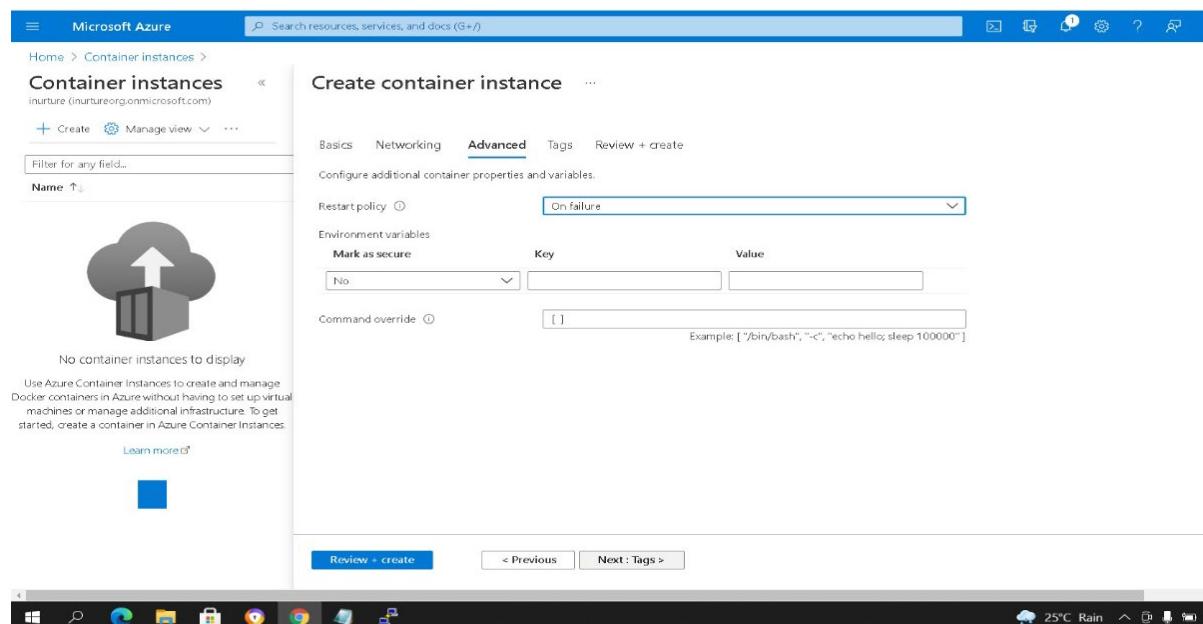


Fig: default advance

22. Here we are created container instance successfully.

Microsoft Azure

Home > Microsoft.ContainerInstances-20210708125453 >

vm-cont Container instances

Search (Ctrl+)

Start | Restart | Stop | Delete | Refresh

Overview

Activity log | Access control (IAM) | Tags

Essentials

Resource group (change) : pgdct-kalyanee Status : Running OS type : Linux

Location : Central India IP address (Public) : 20.198.99.208

Subscription (change) : Azure_subscription FQDN : ---

Subscription ID : 7a229d68-2778-43b5-ac57-37077d7b34d5 Container count : 1

Tags (change) : Click here to add tags

CPU

Memory

Copy to clipboard

12 PM 12:15 PM 12:30 PM 12:45 PM UTC +05:30

12 PM 12:15 PM 12:30 PM 12:45 PM UTC +05:30

25°C Rain

Fig: grab public ip

23.Just grab public ip and browse it you will see nginx web page.

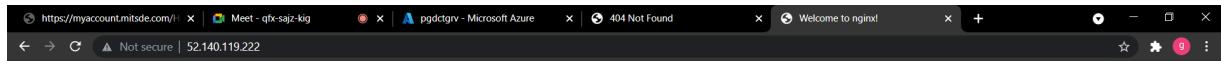
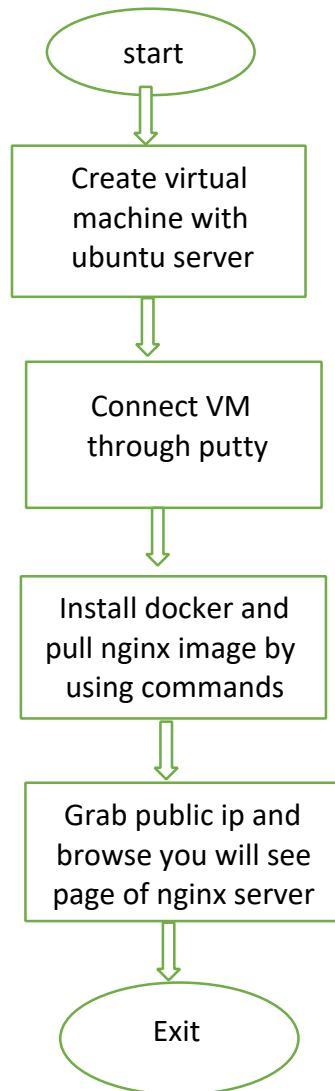
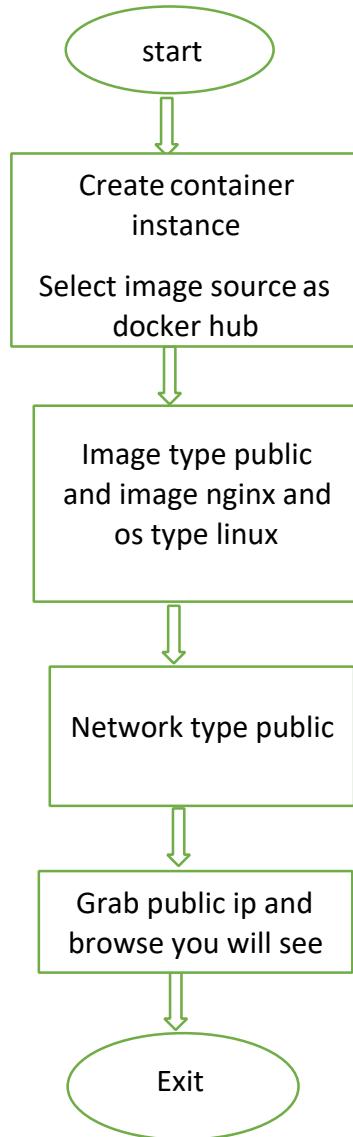


Fig: webpage of nginx server

Flowchart: To install docker on Vm and pull image using docker



To create container instance using docker



Result: We have created virtual machine then connect using putty and install docker engine using commands. We have created container instance using docker hub.

Conclusion: We have successfully Created of Virtual Machine and installed Docker Engine and done with Image Management. We have successfully Launched container instance.

Experiment No.: 6

Name of Experiment: Working with container Instance, Kubernetes cluster, network configuration

Prerequisites: Azure Portal, putty.

Description: **Azure Kubernetes Service:** Azure Kubernetes Service (AKS) simplifies deploying a managed Kubernetes cluster in Azure by offloading the operational overhead to Azure. As a hosted

Kubernetes service, Azure handles critical tasks, like health monitoring and maintenance. Since Kubernetes masters are managed by Azure, you only manage and maintain the agent nodes. Thus, AKS is free; you only pay for the agent nodes within your clusters, not for the masters.

A Kubernetes cluster is a **set of node machines for running containerized applications**. If you're running Kubernetes, you're running a cluster. At a minimum, a cluster contains a control plane and one or more compute machines, or nodes.

Azure Container Registry is a private registry service for building, storing, and managing container images and related artifacts. Use Docker commands to push a container image into the registry, and finally pull and run the image from your registry.

Algorithm:

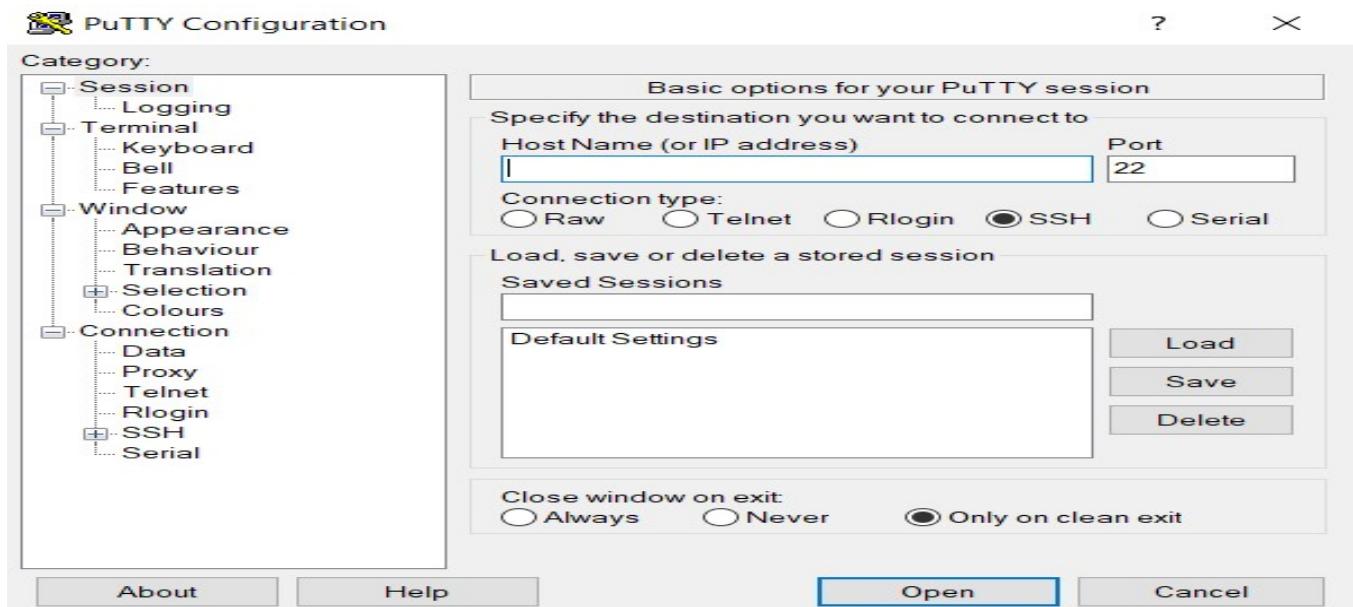
1. Go to container registry.
2. Give the name of container registry and select SKU as basic.

The screenshot shows the 'Create container registry' wizard on the Microsoft Azure portal. The 'Basics' tab is selected. The 'Subscription' dropdown is set to 'Azure_subscription'. The 'Resource group' dropdown is set to '(New) pgdctgrv' with a 'Create new' link. The 'Registry name' field is filled with 'regrv'. The 'Location' dropdown is set to 'Central India'. The 'Availability zones' section has a checkbox for 'Enabled' which is unchecked. A note below states: 'Availability zones are enabled on premium registries and in regions that support availability zones.' The 'SKU' dropdown is set to 'Standard'. At the bottom, there are 'Review + create' and 'Next: Networking >' buttons.

3. Go with default setting and create container registry.

4. Now create the virtual machine with ubuntu server.
5. Select the port http and ssh.

6. Now connect the virtual machine through ssh using putty.



7. Install docker using the commands.

```
root@pgdctgrv:/home/grv
Using username "grv".
grv@52.140.119.222's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1055-azure x86_64)

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

System information as of Wed Jul 21 06:46:10 UTC 2021

System load: 0.27      Processes:          133
Usage of /: 4.6% of 28.90GB   Users logged in:    0
Memory usage: 5%           IP address for eth0: 10.1.0.4
Swap usage: 0%

0 updates can be applied immediately.

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.

grv@pgdctgrv:~$ sudo su
root@pgdctgrv:/home/grv# sudo apt-get update
Hit:1 http://azure.archive.ubuntu.com/ubuntu bionic InRelease [80.7 kB]
Get:2 http://azure.archive.ubuntu.com/ubuntu bionic-updates InRelease [74.6 kB]
Get:3 http://azure.archive.ubuntu.com/ubuntu bionic-backports InRelease [8570 kB]
Get:4 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:5 http://azure.archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]
Get:6 http://azure.archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]
Get:7 http://azure.archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [151 kB]
Get:8 http://azure.archive.ubuntu.com/ubuntu bionic/multiverse Translation-en [108 kB]
Get:9 http://azure.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [2161 kB]
Get:10 http://azure.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1740 kB]
Get:11 http://azure.archive.ubuntu.com/ubuntu bionic-updates/universe Translation-en [372 kB]
Get:12 http://azure.archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [30.5 kB]
Get:13 http://azure.archive.ubuntu.com/ubuntu bionic-updates/multiverse Translation-en [7120 B]
Get:14 http://azure.archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [10.0 kB]
Get:15 http://azure.archive.ubuntu.com/ubuntu bionic-backports/main Translation-en [4764 B]
Get:16 http://azure.archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [10.3 kB]
Get:17 http://azure.archive.ubuntu.com/ubuntu bionic-backports/universe Translation-en [4588 B]
Get:18 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [1132 kB]
Get:19 http://security.ubuntu.com/ubuntu bionic-security/universe Translation-en [257 kB]
```

8. Now enter the following command:

```
curl -sL
https://packages.microsoft.com/keys/microsoft.asc
| \ gpg --dearmor | \
```

```
sudo tee /etc/apt/trusted.gpg.d/microsoft.asc.gpg> /dev/null
```

```
AZ_REPO=$(lsb_release -cs)
```

```
echo "deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ $AZ_REPO
main" | \ sudo tee /etc/apt/sources.list.d/azure-cli.list
```

```

Setting up containerd.io (1.4.9-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service →
 /lib/systemd/system/containerd.service.
Setting up docker-ce-rootless-extras (5:20.10.8-3~ubuntu-bionic) ...
Setting up docker-scan-plugin (0.8.0-ubuntu-bionic) ...
Setting up libltdl7:amd64 (2.4.6-2) ...
Setting up docker-ce-cll (5:20.10.8-3~ubuntu-bionic) ...
Setting up pigg (2.4-1) ...
Setting up docker-ce (5:20.10.8-3~ubuntu-bionic) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/
systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/
systemd/system/docker.socket.
Processing triggers for liblc-bin (2.27-3ubuntu1.4) ...
Processing triggers for systemd (237-3ubuntu10.49) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
sarzoo@aarzoovm:~$ curl -sL https://packages.microsoft.com/keys/microsoft.asc | \
\
: command not found
sarzoo@aarzoovm:~$ curl -sL https://packages.microsoft.com/keys/microsoft.asc | \
\
: command not found
sarzoo@aarzoovm:~$ curl -sL https://packages.microsoft.com/keys/microsoft.asc | \
gpg --dearmor | \
> sudo tee /etc/apt/trusted.gpg.d/microsoft.asc.gpg > /dev/null
sarzoo@aarzoovm:~$ 
sarzoo@aarzoovm:~$ sudo tee /etc/apt/sources.list.d/azure-cli.list

# [1]+  Stopped                  sudo tee /etc/apt/sources.list.d/azure-cli.list
sarzoo@aarzoovm:~$ AS_REPO=$(lsb_release -cs)
sarzoo@aarzoovm:~$ echo "deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/
$AS_REPO main" | \
> sudo tee /etc/apt/sources.list.d/azure-cli.list
deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ bionic main
[sarzoo@aarzoovm:~$]

sudo apt-get update sudo apt-get install azure-cli sudoaz login sudoazacr login --
name grvregsudo docker pull nginx:1.17.0

{
    "homeTenantId": "1405a654-c396-4773-a008-1dfafbd72796",
    "id": "7a229d68-2770-43b5-ac57-37077d7b34d5",
    "isDefault": true,
    "managedByTenants": [],
    "name": "Azure_subscription",
    "state": "Enabled",
    "tenantId": "1405a654-c396-4773-a008-1dfafbd72796",
    "user": {
        "name": "aarzoopatel@inurtureorg.onmicrosoft.com",
        "type": "user"
    }
}
sarzoo@aarzoovm:~$ sudo az acr login --name aarzooreg
Login Succeeded
sarzoo@aarzoovm:~$ sudo docker tag nginx:1.17.0 aarzooreg.azurecr.io/nginx:1.17.0
Error response from daemon: No such image: nginx:1.17.0
sarzoo@aarzoovm:~$ sudo docker tag nginx:1.17.0 aarzooreg.azurecr.io/ nginx:1.17.0
"docker tag" requires exactly 2 arguments.
See 'docker tag --help'.
Usage: docker tag SOURCE_IMAGE[:TAG] TARGET_IMAGE[:TAG]

Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
sarzoo@aarzoovm:~$ sudo docker tag nginx:1.17.0 aarzooreg.azurecr.io/nginx:1.17.0
Error response from daemon: No such image: nginx:1.17.0
sarzoo@aarzoovm:~$ sudo docker tag nginx aarzooreg.azurecr.io/nginx
Error response from daemon: No such image: nginx:latest
sarzoo@aarzoovm:~$ sudo docker pull nginx:1.17.0
1.17.0: Pulling from library/nginx
fc7181108d40: Full complete
c4277fc40ec2: Full complete
780053e98559: Full complete
Digest: sha256:bdbbf36b7f1f77ffe7bd2a32e59235dff6ecf131e3b6b5b96061c652f30685f3a
Status: Downloaded newer image for nginx:1.17.0
sarzoo@aarzoovm:~$ sudo docker tag nginx:1.17.0 aarzooreg.azurecr.io/nginx:1.17.0
sarzoo@aarzoovm:~$
```

sudo docker tag nginx:1.17.0 grvreg.azurecr.io/nginx:1.17.0 sudo docker push
grvreg.azurecr.io/nginx:1.17.0

```

Digest: sha256:bdbf36b7f1f77ffe7bd2a32e59235dff6ecf131e3b6b5b96061c652f30685f3a
Status: Downloaded newer image for nginx:1.17.0
docker.io/library/nginx:1.17.0
aarzoo@aarzoovm:~$ sudo docker tag nginx:1.17.0 aarzooreg.azurecr.io/nginx:1.17.0
aarzoo@aarzoovm:~$ sudo docker push aarzooreg.azurecr.io/nginx:1.17.0
The push refers to repository [aarzooreg.azurecr.io/nginx]
d7acf794921f: Pushed
d9569ca04881: Pushed
cf5b3c6798f7: Pushed
1.17.0: digest: sha256:079aa93463d2566b7a81cbdf856afc6d4d2a6f9100ca3bcbe24ade92c9a7fe size: 948

```

9. Now go to container instance.

10. Select image source as container registry>>now open power shell and enable container registry using following command:

11. Create the container instance.

12. Go to Kubernetes services.
13. Select availability zone as none and Kubernetes version as 1.19.11.
14. Select method as manual and node count as 1.

Create Kubernetes cluster

Availability zone: None
High availability is recommended for standard configuration.

Kubernetes version: 1.19.11

Primary node pool
The number and size of nodes in the primary node pool in your cluster. For production workloads, at least 3 nodes are recommended for resilience. For development or test workloads, only one node is required. If you would like to add additional node pools or to see additional configuration options for this node pool, go to the 'Node pools' tab above. You will be able to add additional node pools after creating your cluster. Learn more about node pools in Azure Kubernetes Service.

Node size: Standard DS2s
2 vCPUs, 4 GiB memory
Standard DS2s is recommended for standard configuration.

Scale method: Manual
Autoscale
Autoscaling is recommended for standard configuration.

Node count: 1

Review + create

15. In authentication field tick AKS-managed Azure Active Directory.
16. Keep everything default and create the Kubernetes cluster.

kuber | Services and ingresses

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Security

Kubernetes resources

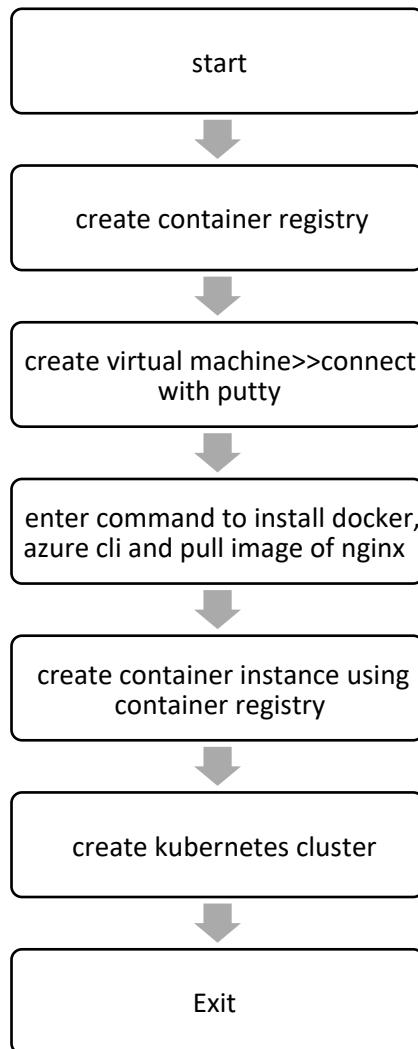
- Namespaces
- Workloads
- Services and ingresses**
- Storage Configuration

Settings

- Node pools
- Cluster configuration
- Networking
- Deployment center (preview)

Name	Namespace	Status	Type	Cluster IP	External IP	Ports	Age
kubernetes	default	Ok	ClusterIP	10.0.0.1		443/TCP	42 minutes
healthmodel-replicaset-service	kube-system	Ok	ClusterIP	10.0.250.221		25227/TCP	41 minutes
kube-dns	kube-system	Ok	ClusterIP	10.0.0.10		53/UDP,53/TCP	41 minutes
metrics-server	kube-system	Ok	ClusterIP	10.0.143.151		443/TCP	41 minutes
my-service	default	Ok	LoadBalancer	10.0.194.159	20.90.241.46	80:31509/TCP	34 minutes

Flowchart:



Result: It has been seen that using container registry we can create container instance and can also push the image in container registry using azure cli and have created Kubernetes cluster.

Conclusion: we have successfully created the container instance using container registry and Kubernetes cluster.

Experiment no :7

AIM: Azure Networking, Address space, & attaching secondary NIC to Virtual Machine.

Prerequisites: Azure portal.

Description:

Azure Networking: The networking services in Azure provide a variety of networking capabilities that can be used together or separately. Some of the services are mentioned below:

- **Connectivity services:** Connect Azure resources and on-premises resources using any or a combination of these networking services in Azure - Virtual Network (VNet), Virtual WAN, ExpressRoute, VPN Gateway, Virtual network NAT Gateway, Azure DNS, Peering service, and Azure Bastion.
- **Application protection services:** Protect your applications using any or a combination of these networking services in Azure - Load Balancer, Private Link, DDoS protection, Firewall, Network Security Groups, Web Application Firewall, and Virtual Network Endpoints.
- **Application delivery services:** Deliver applications in the Azure network using any or a combination of these networking services in Azure - Content Delivery Network (CDN), Azure Front Door Service, Traffic Manager, Application Gateway, Internet Analyzer, and Load Balancer.
- **Network monitoring:** Monitor your network resources using any or a combination of these networking services in Azure - Network Watcher, ExpressRoute Monitor, Azure Monitor, or VNet Terminal Access Point (TAP).

Public IP address: Public IP addresses allow Internet resources to communicate inbound to Azure resources. Public IP addresses enable Azure resources to communicate to Internet and public-facing Azure services. The address is dedicated to the resource, until it's unassigned by you.

Address Space: Address space is the amount of memory allocated for all possible addresses for a computational entity, such as a device, a file, a server, or a networked computer. Address space may refer to a range of either physical or virtual addresses accessible to a processor or reserved for a process.

Network Interface: A Network Interface (NIC) is an interconnection between a Virtual Machine and the underlying software network. An Azure Virtual Machine (VM) has one or more network

interfaces (NIC) attached to it. Any NIC can have one or more static or dynamic public and private IP addresses assigned to it.

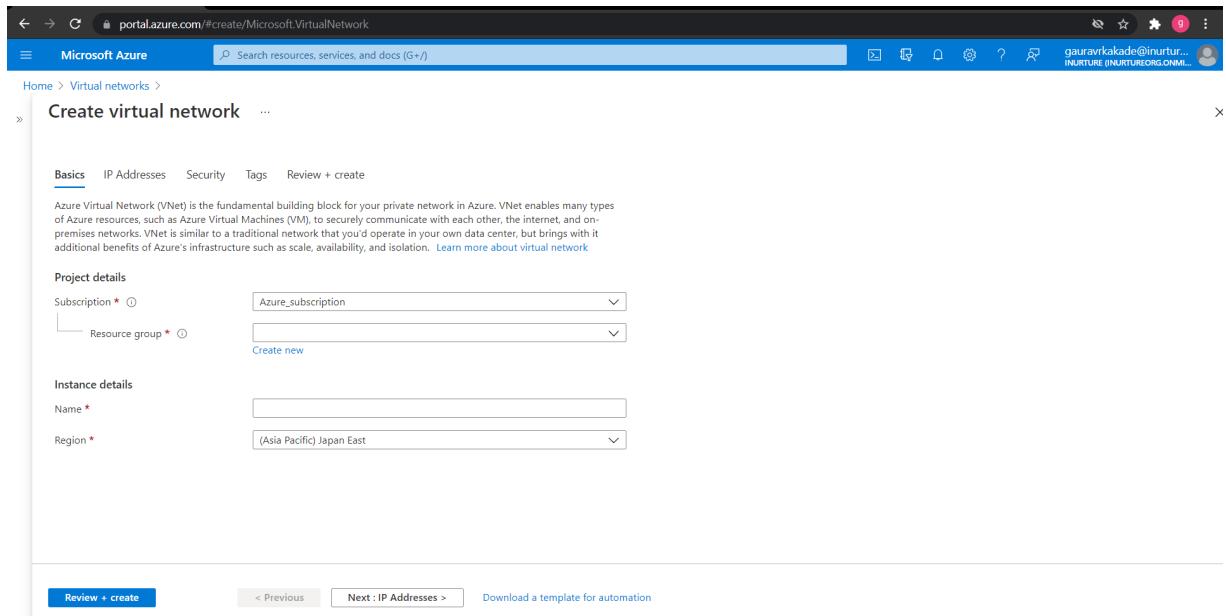
~ Configuring the network interface:

Virtual network & subnets: we can attach a network interface to a VNet and Subnet, and once we deployed a NIC into a VNet, we can't change it.

IP configuration: Public and private IP addresses will be assigned at the NIC level. Primary & secondary IP configurations.

ALGORITHM: We will see how to attach a secondary NIC to a virtual machine.

1.Create a virtual Network.



The screenshot shows the 'Create virtual network' wizard in the Microsoft Azure portal. The 'Basics' step is selected. In the 'Project details' section, 'Subscription' is set to 'Azure_subscription' and 'Resource group' is set to 'Create new'. In the 'Instance details' section, 'Name' is empty and 'Region' is set to '(Asia Pacific) Japan East'. At the bottom, there are buttons for 'Review + create', '< Previous', 'Next : IP Addresses >', and 'Download a template for automation'.

2.Create a Virtual Machine.

3. To see the default NIC go to networking and then go to network interface.
4. Now to attach the secondary NIC to virtual machine, we have to stop the virtual machine.
5. Go to overview of virtual machine and stop the virtual machine.
6. Create a new subnet.

7. Now go to the VM and then to Networking.
8. Click on attach network interface and create new interface.

pgdctgrv | Networking

pgdctgrv621

IP configuration: ipconfig1 (Primary)

Network Interface: pgdctgrv621 **Effective security rules** **Troubleshoot VM connection issues** **Topology**

Virtual network/subnet: grv/default NIC Public IP: 104.41.160.131 NIC Private IP: 172.20.0.4 Accelerated networking: Disabled

Inbound port rules	Outbound port rules	Application security groups	Load balancing			
Network security group pgdctgrv-nsg (attached to network interface: pgdctgrv621) Impacts 0 subnets, 1 network interfaces						
Add inbound port rule						
Priority	Name	Port	Protocol	Source	Destination	Action
300	RDP	3389	TCP	Any	Any	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

- Enter the name of networking interface and keep everything default and then add the subnet that you have created and click on network interface.

Create network interface

Basics

Create a network interface and attach it to a virtual machine. A network interface enables a virtual machine to communicate with internet, Azure, and on-premises resources. [Learn more about network interface](#)

Project details

Subscription *: Azure_subscription

Resource group *: pgdctgrv

Instance details

Name *:

Region *: (Asia Pacific) Japan East

Virtual network: grv

Subnet *: grv (172.20.1.0/24)

Private IP address assignment: Dynamic

Network security group: None

Review + create < Previous Next : Tags > Download a template for automation

- Now in networking you can see the two-network interface.

pgdctgrv | Networking

Virtual machine

Search (Ctrl+ /)

Attach network interface Detach network interface

pgdctgrv621

IP configuration: ipconfig1 (Primary)

Network Interface: pgdctgrv621 **Effective security rules:** Troubleshoot VM connection issues Topology

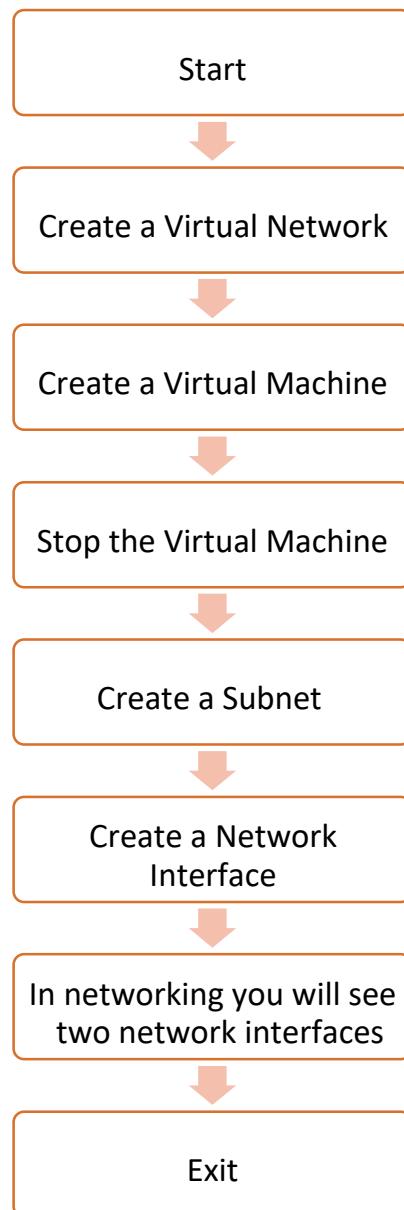
Virtual network/subnet: grv/default NIC Public IP: 104.41.160.131 NIC Private IP: 172.20.0.4 Accelerated networking: Disabled

Inbound port rules Outbound port rules Application security groups Load balancing

Add inbound port rule

Priority	Name	Port	Protocol	Source	Destination	Action
300	RDP	3389	TCP	Any	Any	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

FLOWCHART:



Result: we have seen the concept of azure networking, address space. By default, a network interface card is always associated with a virtual machine but if we want to attach a secondary NIC we can also do that.

Conclusion: we have successfully attached the network interface card.

Experiment No.: 8

Name of Experiment: Creation of VNET & Security rule management of Virtual Machine level and subnet level.

Prerequisites: Azure portal, RDP.

Description:

Virtual Network: Azure Virtual Network (VNET) is the fundamental building block for your private network in Azure. VNET enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks. VNET is similar to a traditional network that you'd operate in your own data centre, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation.

Security Group: You can use an Azure network security group to filter network traffic to and from Azure resources in an Azure virtual network. A network security group contains security rules that allow or deny inbound network traffic to, or outbound network traffic from, several types of Azure resources. For each rule, you can specify source and destination, port, and protocol.

Algorithm:

Steps to create virtual network (VNET):

1. Go to virtual network.
2. Create new resource group.
3. Give the name of virtual network and select the region.

The screenshot shows the Azure portal interface for creating a new virtual network. The URL in the address bar is portal.azure.com/#create/Microsoft.VirtualNetwork. The page title is "Create virtual network". The "Basics" tab is active. In the "Project details" section, the "Subscription" dropdown is set to "Azure_subscription" and the "Resource group" dropdown has "Create new" selected. Under "Instance details", the "Name" field is empty and the "Region" dropdown is set to "(Asia Pacific) Japan East". At the bottom, there are navigation buttons: "Review + create", "< Previous", "Next : IP Addresses >", and "Download a template for automation". The status bar at the bottom right shows the date as 13-07-2023, the time as 10:30 AM, and the location as "33°C Haze".

4. Go with the default IP address and subnet.
5. In security and tags keep everything default.

The virtual network's address space, specified as one or more address prefixes in CIDR notation (e.g. 192.168.1.0/24).

IPv4 address space

10.1.0.0/16 10.1.0.0 - 10.1.255.255 (65536 addresses)

Add IPv6 address space ⓘ

The subnet's address range in CIDR notation (e.g. 192.168.1.0/24). It must be contained by the address space of the virtual network.

+ Add subnet

Subnet name	Subnet address range	NAT gateway
<input type="checkbox"/> default	10.1.0.0/24	-

Review + create **< Previous** **Next : Security >** Download a template for automation

6. Create the virtual network.

Steps to create security group and attach it to virtual machine:

1. Create a virtual machine.
2. In networking select network security group as none.

The screenshot shows the Azure portal interface for creating a virtual machine. The top navigation bar includes links for Home, Virtual machines, Create a virtual machine, Basics, Disks, Networking (which is selected), Management, Advanced, Tags, and Review + create. The main content area is titled 'Networking' and contains fields for 'Virtual network' (set to 'Create new'), 'Public IP' (set to 'None'), and 'NIC network security group' (set to 'None'). A note states that accelerated networking is not supported for the selected VM size. Below this is a 'Load balancing' section with a note about placing the VM in an existing Azure load balancing solution. At the bottom are 'Review + create' and 'Next : Management >' buttons.

3. Keep everything default and create the virtual machine.
4. Now try to connect with the virtual machine through RDP, it will connect because in azure everything is open in security group.

The screenshot shows the Azure portal interface for connecting to a virtual machine named 'pgdctgrv'. The left sidebar lists 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', and 'Diagnose and solve problems'. Under 'Settings', 'Networking' is selected. The main pane shows the 'Connect' section with 'RDP' selected. A 'Windows Security' dialog box is open, prompting for credentials to connect to the IP address 104.41.160.131. The 'IP address' field is filled with 'Public IP address (104.41.160.131)' and the 'Port number' is set to 3389. The 'Download RDP File' button is visible. The dialog also includes 'Remember me' and 'More choices' options, and buttons for 'OK' and 'Cancel'.

5. Now install IIS web server in virtual machine and browse the IP address of virtual machine and you will see the webpage of IIS webserver; this is because everything is open in security group by default.



6. Now to add security group, go to virtual machine.
7. Network security group>> select the resource group>>enter the name of security group and create.

Microsoft Azure

Search resources, services, and docs (G+ /)

Home > Network security groups >

Create network security group ...

Basics Tags Review + create

Project details

Subscription * Azure_subscription

Resource group * pgdctgrv Create new

Instance details

Name *

Region * (Asia Pacific) Japan East

Review + create < Previous Next : Tags > Download a template for automation

8. Now in inbound security rule>>click on add>>choose the service as http.
9. Keep action allow and enter priority as 100>> add the security rule.

Add inbound security rule

Source: Any

Source port ranges: *

Destination: Any

Service: Custom

Destination port ranges: 8080

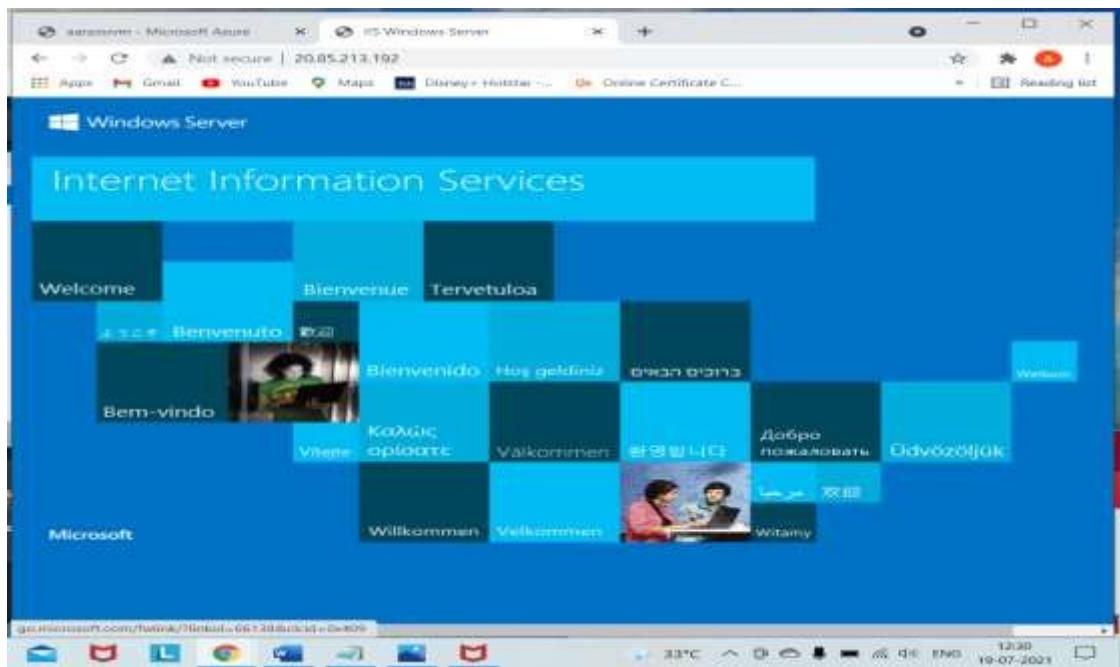
Protocol: Any

Action: Allow

Priority ↑↓	Name ↑↓	Port ↑↓	Protocol ↑↓
300	RDP	3389	TCP
65000	AllowVnetInBound	Any	Any
65001	AllowAzureLoadBalancer...	Any	Any
65500	DenyAllInBound	Any	Any

10. Now go to virtual machine>>stop the virtual machine.
11. Go to network interface>>network security group>>select your network security group and save.
12. You will see the security group has attached to your network interface.

13. Now copy the public IP address of virtual machine and browse it you will see your web server.



14. Also try to connect with RDP, it will not connect as we have not allowed the RDP security rule.
15. To connect through RDP>>create a inbound security rule with service RDP>>now try to connect through RDP, it will connect.

Priority	Name	Port	Protocol
300	RDP	3389	TCP
65000	AllowNetInBound	Any	Any
65001	AllowAzureLoadBalancer...	Any	Any
65500	DenyAllInBound	Any	Any

Add inbound security rule

Source: Any

Source port ranges: *

Destination: Any

Service: Custom

Destination port ranges: * 8080

Protocol: Any

Action: Allow

The screenshot shows the Microsoft Azure portal interface. On the left, there is a sidebar with various options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Networking, Connect, Windows Admin Center (preview), Disks, Size, Security, Advisor recommendations, Extensions, Continuous delivery, Availability + scaling, and Configuration. The 'Connect' option is currently selected. In the center, there is a 'Windows Security' dialog box titled 'Enter your credentials'. It asks for an IP address (grv) and a port number (3389). There is also a 'Remember me' checkbox and a 'Download RDP File' button. Below the main form, there are links for 'Can't connect?', 'Test your connection', 'Troubleshoot RDP connectivity issues', and 'Provide feedback'. At the bottom right of the dialog are 'OK' and 'Cancel' buttons. The status bar at the bottom shows the date and time as 17:27 on 13-Aug-21.

Steps to create security group and attach to a subnet:

1. Create a new security group.

The screenshot shows the Microsoft Azure portal interface. The URL in the address bar is 'portal.azure.com/#create/Microsoft.NetworkSecurityGroup'. The sidebar on the left shows 'Home > Network security groups > Create network security group'. The main area is the 'Create network security group' wizard, specifically the 'Basics' step. It has tabs for 'Basics', 'Tags', and 'Review + create'. Under 'Project details', 'Subscription' is set to 'Azure_subscription' and 'Resource group' is set to 'pgdctgrv'. Under 'Instance details', 'Name' is a required field (indicated by a red asterisk) and 'Region' is set to '(Asia Pacific) Japan East'. At the bottom, there are buttons for 'Review + create', '< Previous' and 'Next : Tags >', and a link to 'Download a template for automation'. The status bar at the bottom shows the date and time as 17:26 on 13-Aug-21.

2. Go to virtual network>>subnets>>add

Associate subnet

Virtual network: grv

Subnet: grv

OK

3. Go to inbound security rule.
4. Create a rule with service http>>choose action as deny and add.

Add inbound security rule

Source: Any

Service: Custom

Protocol: Any

Destination port ranges: 8080

Action: Allow

Add

5. As we have denied the Http at subnet level and has allow the http at virtual machine level, Now try to browse the IP address and see you will get your IIS web server or not.
6. It will not show the webserver.

7. Now, go to subnet level security group and allow the http traffic and again try to browse the IP, now it will show the webserver.

Microsoft Azure | portal.azure.com

Home > Network security groups > pgdctgrv-nsg

Search resources, services, and docs (G+)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

- Inbound security rules
- Outbound security rules
- Network interfaces
- Subnets
- Properties
- Locks
- Monitoring
- Alerts
- Diagnostic settings
- Logs
- NSG flow logs

Move Delete Refresh Give feedback

Essentials

Resource group (change) : pgdctgrv

Location : Japan East

Subscription (change) : Azure_subscription

Subscription ID : 7a229d68-2778-43b5-ac57-37077d7b34d5

Tags (change) : Click here to add tags

Custom security rules : 1 inbound, 0 outbound

Associated with : 0 subnets, 1 network interfaces

Inbound Security Rules

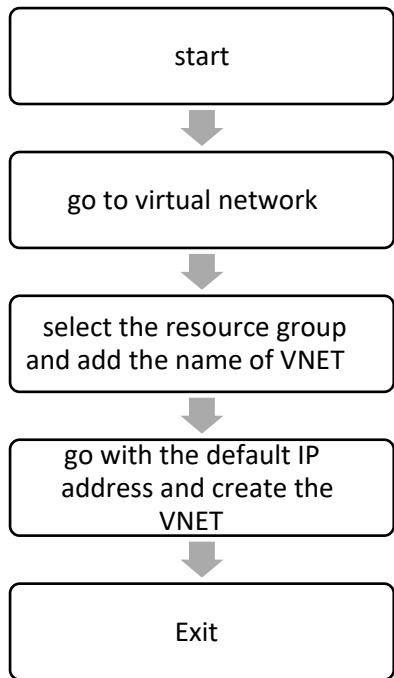
Priority ↑	Name ↑	Port ↑	Protocol ↑	Source ↑	Destination ↑	Action ↑
300	RDP	3389	TCP	Any	Any	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancer...	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

Outbound Security Rules

Name ↑	Port ↑	Protocol ↑	Source ↑	Destination ↑	Action ↑
AllowVnetOutBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
AllowInternetOutBound	Any	Any	Any	Internet	Allow
DenyAllOutBound	Any	Any	Any	Any	Deny



Flowchart:



Result: The virtual network has been created and after creating security group we have seen the effect of security group at virtual machine level and at subnet level.

Conclusion: we have successfully created the virtual network and security group, and has attached the security group at virtual machine level and subnet level.

Experiment No.: 9

Name of Experiment: Creation of VNET using ARM management.

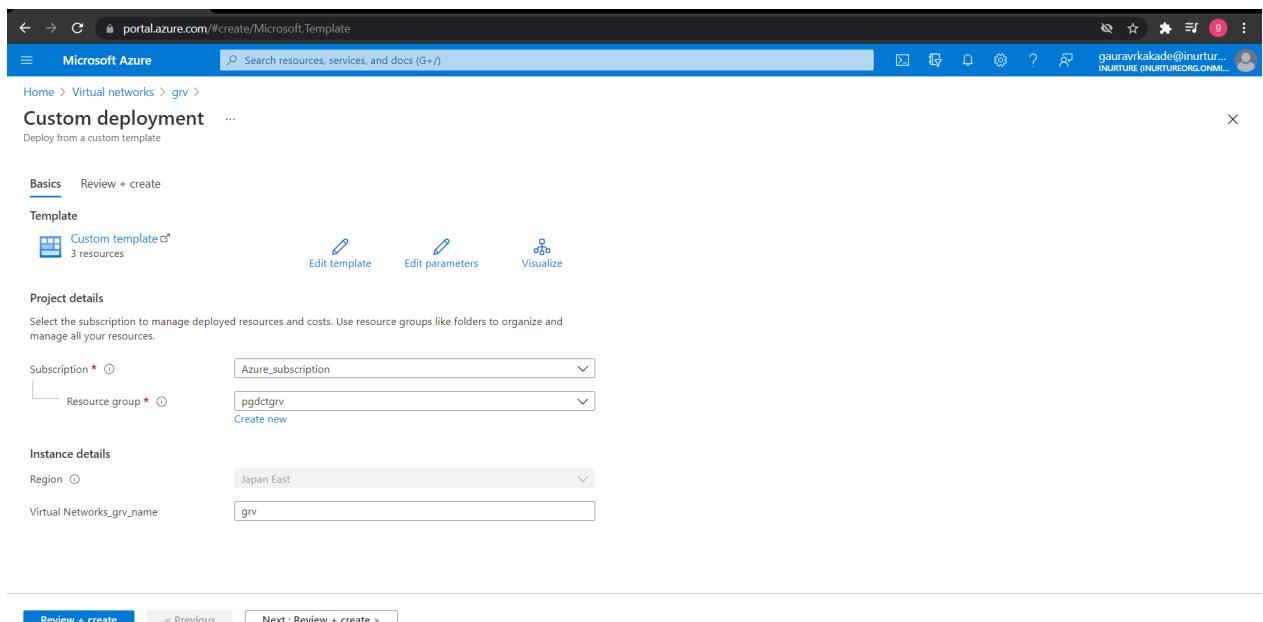
Prerequisites: Azure portal.

Description: ARM Templates are a way to declare the objects you want, the types, names and properties in a JSON file which can be checked into source control and managed like any other code file. ARM Templates are what really gives us the ability to roll out Azure “Infrastructure as code”.

An ARM template can either contain the contents of an entire resource group or it can contain one or more resources from a resource group. When a template is deployed, you have the option of either using ‘complete’ or ‘incremental’ mode.

Algorithm:

1. Go to deploy a custom template.
2. Click on build your own template with editor.



The screenshot shows the Azure portal interface for creating a new ARM template. At the top, the URL is `portal.azure.com/#create/Microsoft.Template`. The main title is "Custom deployment". Below it, there's a note: "Deploy from a custom template". The "Basics" tab is selected. Under "Template", there's a preview icon showing "Custom template" and "3 resources". To the right are buttons for "Edit template", "Edit parameters", and "Visualize". The "Project details" section asks to select a subscription and resource group. The chosen subscription is "Azure_subscription" and the resource group is "pgdctgrv". The "Instance details" section specifies the region as "Japan East" and the virtual network name as "grv". At the bottom, there are navigation buttons: "Review + create", "< Previous", and "Next : Review + create >".

3. There you will see a window where you can add your own code.
4. Other option is that, you can click on add resource.
5. Where you can see various resources.
6. Click on virtual network and give the name.

```

12
13     "type": "Microsoft.Network/virtualNetworks",
14     "apiVersion": "2020-11-01",
15     "name": "[parameters('virtualNetworks_grv_name')]",
16     "location": "japaneast",
17     "properties": {
18         "addressSpace": {
19             "addressPrefixes": [
20                 "172.20.0.0/16"
21             ]
22         },
23         "subnets": [
24             {
25                 "name": "default",

```

Add a resource to the template

Select a resource *

OK **Cancel**

Save **Discard**

7. Then you can see azure automatically add the code for virtual network.
8. Then you can save the template.

```

1     "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
2     "contentVersion": "1.0.0.0",
3     "parameters": {
4         "virtualNetworks_grv_name": {
5             "defaultValue": "grv",
6             "type": "string"
7         }
8     },
9     "variables": {},
10    "resources": [
11        {
12            "type": "Microsoft.Network/virtualNetworks",
13            "apiVersion": "2020-11-01",
14            "name": "[parameters('virtualNetworks_grv_name')]",
15            "location": "japaneast",
16            "properties": {
17                "addressSpace": {
18                    "addressPrefixes": [
19                        "172.20.0.0/16"
20                    ]
21                },
22                "subnets": [
23                    {
24                        "name": "default",

```

Save **Discard**

```

54     ],
55     "properties": {
56       "addressPrefix": "172.20.0.0/24",
57       "delegations": [],
58       "privateEndpointNetworkPolicies": "Enabled",
59       "privateLinkServiceNetworkPolicies": "Enabled"
60     }
61   },
62   {
63     "type": "Microsoft.Network/virtualNetworks/subnets",
64     "apiVersion": "2020-11-01",
65     "name": "[concat(parameters('virtualNetworks_grv_name'), '/', parameters('virtualNetworks_grv_name'))]",
66     "dependsOn": [
67       "[resourceId('Microsoft.Network/virtualNetworks', parameters('virtualNetworks_grv_name'))]"
68     ],
69     "properties": {
70       "addressPrefix": "172.20.1.0/24",
71       "serviceEndpoints": [],
72       "delegations": [],
73       "privateEndpointNetworkPolicies": "Enabled",
74       "privateLinkServiceNetworkPolicies": "Enabled"
75     }
76   }
77 ]
78

```

Save **Discard**

9. Select your resource group and region and then create.

Custom deployment ...

Deploy from a custom template

Basics **Review + create**

Template

- Custom template
- 3 resources
- Edit template
- Edit parameters
- Visualize

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * Azure_subscription

Resource group * pgdctgrv
Create new

Instance details

Region Japan East

Virtual Networks_grv_name grv

Review + create < Previous Next : Review + create >

10. You will see your virtual network is created.

Overview

Resource group (change) : pgdctgrv

Location : Japan East

Subscription (change) : Azure_subscription

Subscription ID : 7a229d68-2778-43b5-ac57-37077d7b34d5

Tags (change) : Click here to add tags

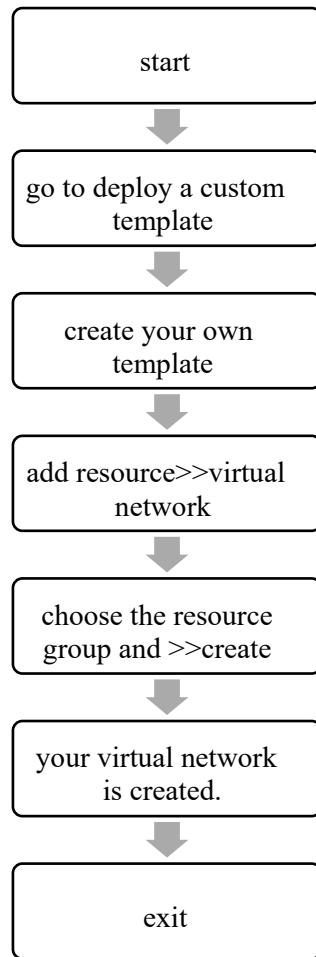
Address space : 172.20.0.0/16

DNS servers : Azure provided DNS service

Connected devices

Device ↑↓	Type ↑↓	IP Address ↑↓	Subnet ↑↓
pgdctgrv621	Network interface	172.20.0.4	default

Flowchart:



Result: It has been seen that we can also create various resources using ARM template, where we have to just select the resource and give name, the code is given by azure or we can also deploy our code and create the resource.

Conclusion: we have successfully created the virtual network using ARM template.

Experiment No. 10

Name of Experiment: Creating load balancer in azure- Basic and Standard.

Prerequisites: Azure portal, RDP, putty.

Description: Load balancing refers to evenly distributing load (incoming network traffic) across a group of backend resources or servers. Azure Load Balancer operates at layer 4 of the Open Systems Interconnection (OSI) model. It's the single point of contact for clients. Load balancer distributes inbound flows that arrive at the load balancer's front end to backend pool instances. These flows are according to configured loadbalancing rules and health probes. The backend pool instances can be Azure Virtual Machines or instances in a virtual machine scale set.

Load balancer supports both Standard and Basic SKUs. These SKUs differ in scenario scale, features, and pricing. Any scenario that's possible with Basic load balancer can be created with Standard load balancer.

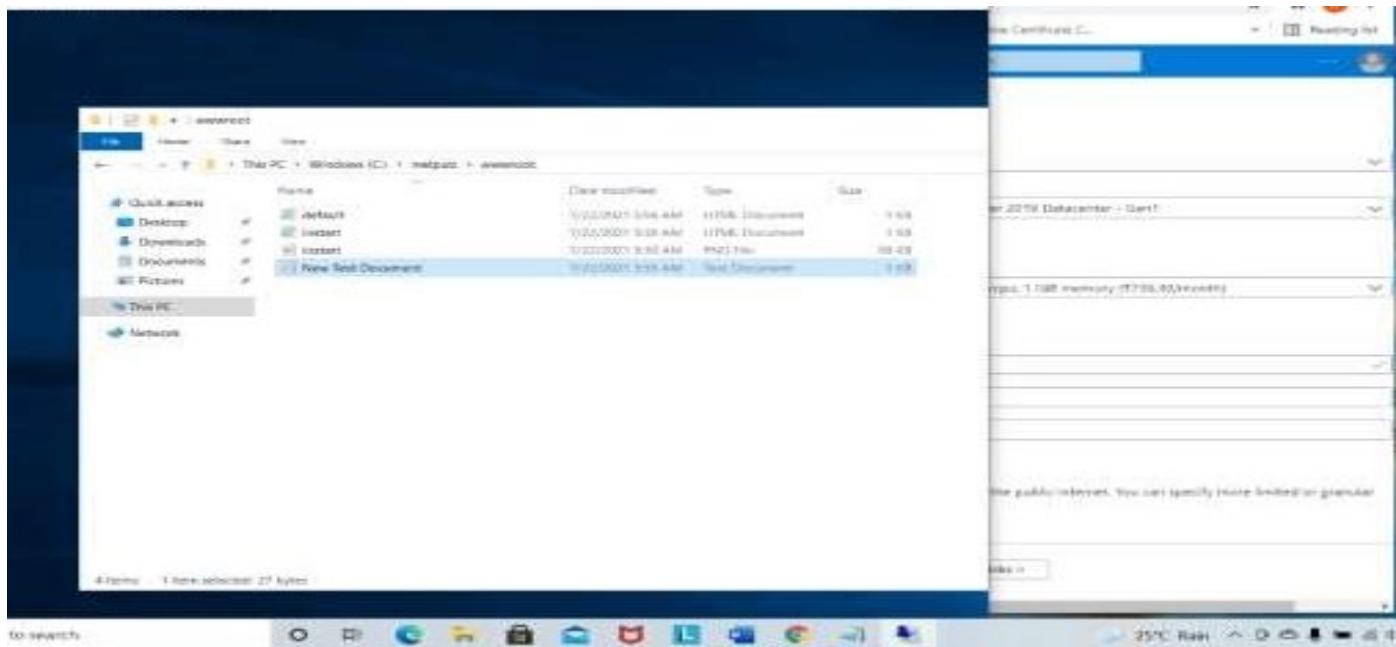
Algorithm:

Creating load balancer with basic SKUs:

1. Create two virtual machines.
2. Create the availability sets in both virtual machines.

The screenshot shows the Azure portal interface for creating an availability set. The URL is portal.azure.com/#create/Microsoft.AvailabilitySet. The process is at the 'Create availability set' step. In the 'Project details' section, a subscription 'Azure_subscription' and a new resource group 'pgdctgrv' are selected. In the 'Instance details' section, a name 'pgdctgrv' and region '(Europe) UK South' are specified, along with a fault domain count of 2. A note states: 'The maximum platform fault domain count in the selected subscription and location is 2.' At the bottom, there are 'Review + create' and 'Next : Advanced >' buttons.

3. Install IIS server on both the virtual machines.
4. Create a html file in c drive>>inetpub>>wwwroot>>default.html (do this in both the virtual machines).



5. Disable the public IP of both virtual machine, to disable the IP go to networking>>go to NIC>>IP configuration>>disassociate>>save.
6. Create the public IP address>>give the name of public IP and keep IP address assignment as dynamic and click on create.

Create public IP address

IP Version IPv4 IPv6 Both

SKU Standard Basic

Tier Regional Global

IPv4 IP Address Configuration

Name

IP address assignment Dynamic Static

Routing preference Microsoft network Internet

Idle timeout (minutes)

DNS name label

Create **Automation options**

7. Go to load balancer>>give the name of load balancer>>keep type as public and SKUs as basic.
8. Choose the public IP address that you have created and create the load balancer.

Azure load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. Load balancers uses a hash-based distribution algorithm. By default, it uses a 5-tuple (source IP, source port, destination IP, destination port, protocol type) hash to map traffic to available servers. Load balancers can either be internet-facing where it is accessible via public IP addresses, or internal where it is only accessible from a virtual network. Azure load balancers also support Network Address Translation (NAT) to route traffic between public and private IP addresses. [Learn more](#).

Project details

Subscription * Azure_subscription

Resource group * (New) pgdctgrv [Create new](#)

Instance details

Name * load1

Region * (Asia Pacific) Australia Central

Type * Internal Public

SKU * Standard Basic

[Review + create](#) < Previous Next : Frontend IP configuration > Download a template for automation [Give feedback](#)

9. Now create the backend pool>>and add both the virtual machines.

Name * Backend pool name

Virtual network * australiacentral

Backend Pool Configuration NIC IP Address

IP Version IPv4 IPv6

Virtual machines

You can only attach virtual machines in australiacentral that have a standard SKU public IP configuration or no public IP configuration. All IP configurations must be on the same virtual network.

No virtual machine is found in australiacentral that matches the above criteria

[Add](#) [Give feedback](#)

10. Now create the load balancing rule>>give the name of the rule>>select the frontend IP address.
11. Add port 80>>add backend port as 80>>add your backend pool.
12. Create the health probe>>give the name of health probe>>give port 80 and create>>add the load balancing rule.

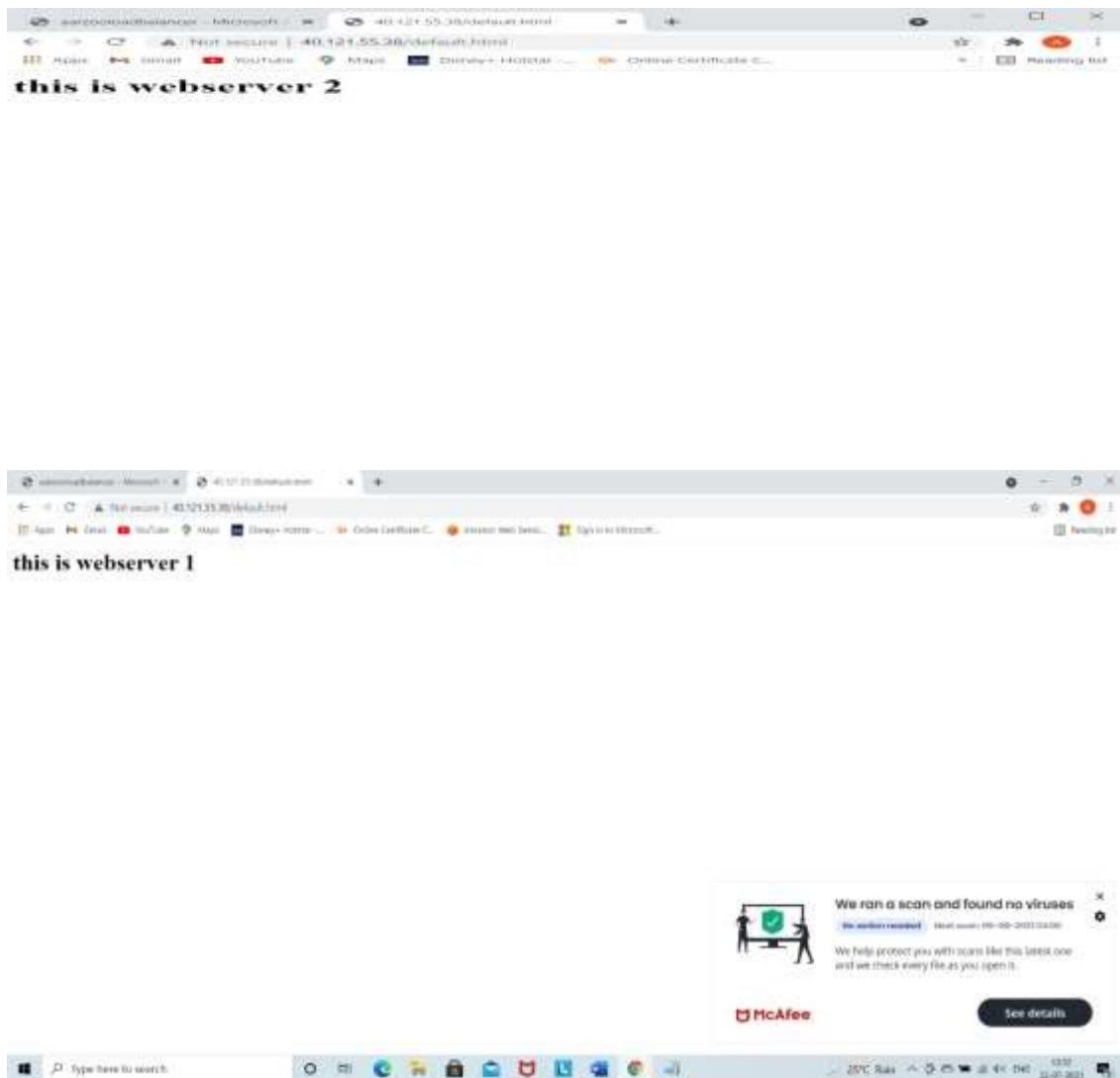
The screenshot shows the Microsoft Azure portal interface. The URL in the address bar is portal.azure.com/#@inurtureorg.onmicrosoft.com/resource/subscriptions/7a229d68-2778-43b5-ac57-37077d7b34d5/resourcegroups/pgdctgrv/providers/Microsoft.Network/loadBalancers/l.... The page title is "Microsoft Azure" and the search bar says "Search resources, services, and docs (G+/-)". The user is logged in as "gauravrkakade@inurture.org.in". The current path is "Home > Microsoft.LoadBalancer-20210810094639 > load1 > Add health probe". The form fields are as follows:

Name *	<input type="text" value="Health Probe Name"/>
Protocol *	<input type="text" value="TCP"/>
Port *	<input type="text" value="80"/>
Interval *	<input type="text" value="5"/> seconds
Unhealthy threshold *	<input type="text" value="2"/> consecutive failures
Used by	<input type="text" value="Not used"/>

At the bottom of the form, there are two buttons: "Add" (highlighted in blue) and "Give feedback".



13. Now copy the IP address of the load balancer and browse(40.121.22.38/default.html).
14. You will see you webpage1, after refreshing in sometime you will see your another webpage.



Creating load balancer using standard SKUs:

Here we will see a use case in which we will configure multiple backend pool and will see how to handle both backend pool through frontend IP configuration.

1. Create two virtual machines.
2. Disassociate public IP of both the virtual machines.
3. Create the load balancer with standard SKUs

Name *
Backend pool name

Virtual network * ⓘ

Backend Pool Configuration
 NIC
 IP Address

IP Version
 IPv4
 IPv6

Virtual machines
You can only attach virtual machines in australiacentral that have a standard SKU public IP configuration or no public IP configuration. All IP configurations must be on the same virtual network.

No virtual machine is found in australiacentral that matches the above criteria

4. Create the public IP address>>create the load balancer.
5. Create the backend pool and select both the virtual machines.

Add virtual machines to backend pool

You can only attach virtual machines that are in the same location and on the same virtual network as the loadbalancer. Virtual machines must have a standard SKU public IP or no public IP.

Virtual machine	Resource group	IP Configuration	Availability set	Tags
pgdctaarzoovm3	pgdctaarzoo	ipconfig1 (10.1.0.4)	-	-
pgdctaarzoovm4	pgdctaarzoo	ipconfig1 (10.1.0.5)	-	-

6. Create health probe and create load balancing rule.
7. Create NAT rule>>give the name of rule>>select the frontend IP address>>In service choose custom>>give port 4000>>select the target virtual machine>>add.

8. Now open command prompt and type the command **mstsc /v: IP address:4000**.
9. Connect both the virtual machines with RDP and install IIS server.
10. Now, Create a new linux virtual machine.
11. Connect with putty.
12. Install nginx server on machine: **sudo apt-get install nginx**.

13. Disassociate the public IP.
14. Now go to load balancer and create a new backend pool>>add the linux virtual machine.

portal.azure.com/#@inurtureorg.onmicrosoft.com/resource/subscriptions/7a229d68-2778-43b5-ac57-37077d7b34d5/resourcegroups/pgdctgrv/providers/Microsoft.Network/loadBalancers/l...

Microsoft Azure Search resources, services, and docs (G+)

Home > Microsoft.LoadBalancer-20210810094639 > load1 > Add backend pool ...

Name * Backend pool name

Virtual network * (dropdown menu)

Backend Pool Configuration
 NIC
 IP Address

IP Version
 IPv4
 IPv6

Virtual machines
 You can only attach virtual machines in australiacentral that have a standard SKU public IP configuration or no public IP configuration. All IP configurations must be on the same virtual network.

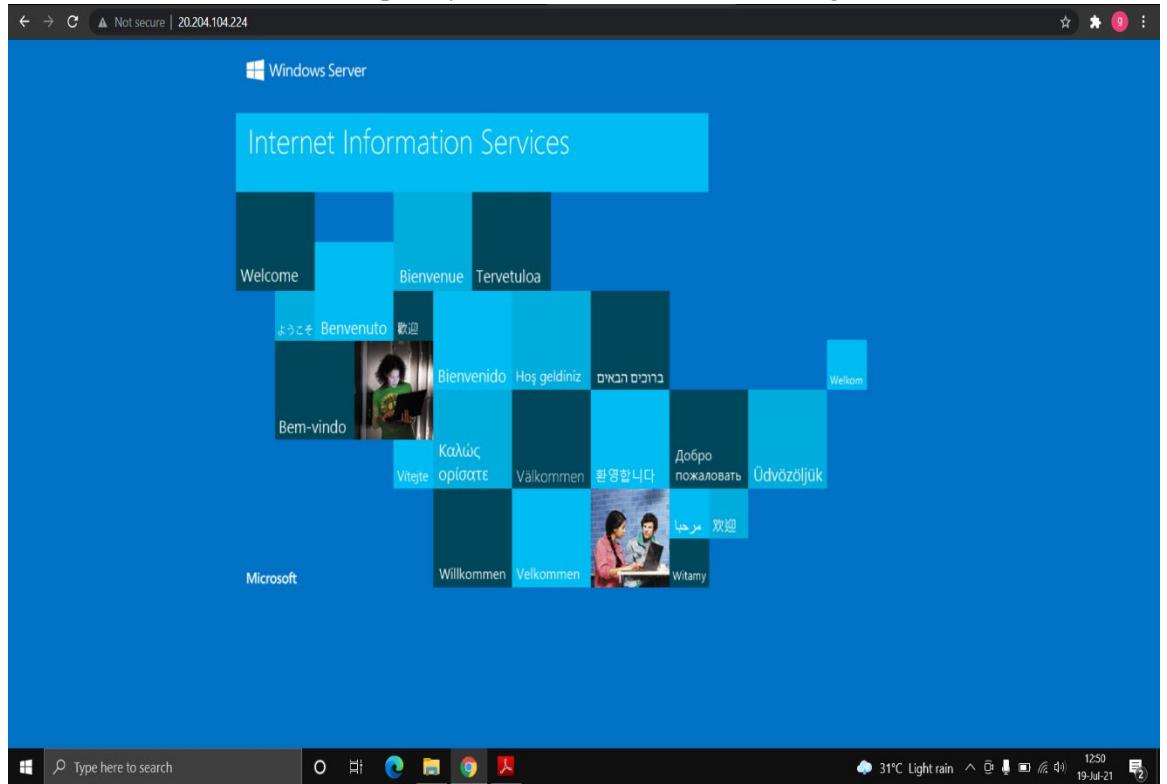
No virtual machine is found in australiacentral that matches the above criteria

Add Give feedback

Type here to search

31°C AQI 92 10-Aug-21

15. Add frontend IP address.
16. Create new load balancing rule and select the health probe.
17. Now browse the IP of both the pool, you will see the IIS server and nginx server.





Result: The load balancer has been created with basic and standard SKUs, and it has been seen that while creating load balancer with basic SKU the virtual machine must be in an availability set and with standard SKU the multiple virtual machine can be in load balancer backend pool without creating availability set. We have also seen the configuration of multiple backend pools.

Conclusion: we have successfully created the basic and standard load balancer and have also seen how to created multiple backend pools.