
Capstone-Project 1

By: Ankit Ranjan

Problem Statement:

You work in XYZ Corporation. Your company is facing some issues and wish for You to use Azure cloud expertise to solve them.

Issues:

1. They wish to have a centralized store to store all their developer tools in. This store should be such that developers can mount it on their file system.
2. They wish to store large volumes of image data. They wish to have low latency access to frequently accessed images i.e., images that have been accessed in the last 14 days. If an image is not accessed within the last 14 days they wish to archive them.
3. They wish to lower the latency of their website. They have noticed that users who are far away from their web server have complained that images take a lot of time to load.
4. They wish to serve another website on Azure's VMs.
5. You also want to have two VMs in different networks. They wish for you to deploy those VMs and enable communication between them.
6. They wish to use Azure to resolve their site with domain 'simple-site.tk' to its IP address.
7. They wish for both the VMs serving their website to be more reliable so that if one VM fails the traffic is automatically routed to the other one.
8. They wish for you to find a way to assign and manage credentials for Azure for all 10 employees in the company.
9. Finally they have two applications that need to pass messages between one another on an on-demand basis i.e., an application will send the message and other applications will receive and process it when it can. You need to set up a service in such a way that these applications can do so (you are provided with the code). All you need to do is make changes to the config file.

You need to use services from Azure cloud to help your company resolve all these issues.


Issue 1: They wish to have a centralized store to store all their developer tools in. This store should be such that developers can mount it on their file system.


Solution:


Step 1: Create and open your storage account


Step 2: Click on Files

Services

**Blobs**
REST-based object storage for unstructured data
[Learn more](#)


**Files**
File shares that use the standard SMB 3.0 protocol
[Learn more](#)


**Tables**
Tabular data storage
[Learn more](#)

**Queues**
Effectively scale apps according to traffic
[Learn more](#)

Tools and SDKs

Step 3: Click on +FileShare

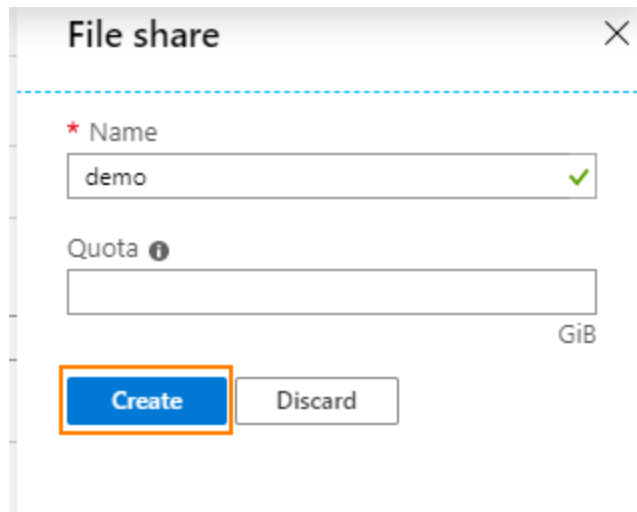
 File share

 Refresh

Storage account: [vmwestrgdiag](#)

NAME	MODIFIED	QUOTA
You don't have any file shares yet. Click '+ File share' to get started.		

Step 4: Enter the details and click on Create

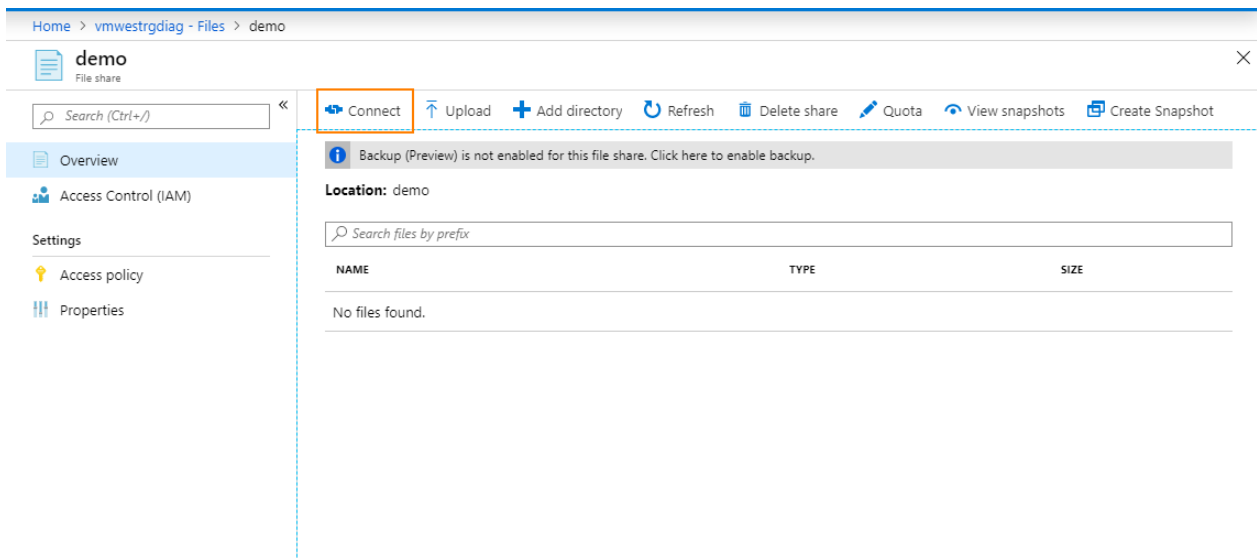


The image shows a 'File share' dialog box with a close button (X) in the top right corner. Below the title bar, there is a dashed blue line. The first section is labeled '* Name' and contains a text input field with the value 'demo' and a green checkmark icon to its right. Below this is a section labeled 'Quota' with an information icon (i) and a text input field. To the right of the input field, the unit 'GiB' is displayed. At the bottom of the dialog, there are two buttons: 'Create' (highlighted with an orange border) and 'Discard'.

Step 5: Mount the File Share on your PC

Step 5.1: Open the File Share in Azure Portal

Step 5.2: Click on Connect



The image shows the Azure Portal interface for a file share named 'demo'. The breadcrumb navigation at the top reads 'Home > vmwestrgdiag - Files > demo'. The left sidebar contains a search bar and a list of options: 'Overview' (selected), 'Access Control (IAM)', 'Settings', 'Access policy', and 'Properties'. The main content area has a toolbar with buttons: 'Connect' (highlighted with an orange border), 'Upload', 'Add directory', 'Refresh', 'Delete share', 'Quota', 'View snapshots', and 'Create Snapshot'. Below the toolbar, there is a message: 'Backup (Preview) is not enabled for this file share. Click here to enable backup.' The 'Location' is set to 'demo'. There is a search bar for files with the placeholder 'Search files by prefix'. Below this is a table with columns 'NAME', 'TYPE', and 'SIZE'. The table is currently empty, displaying 'No files found.'

Step 5.3: Copy the command and check you have URL, username and password

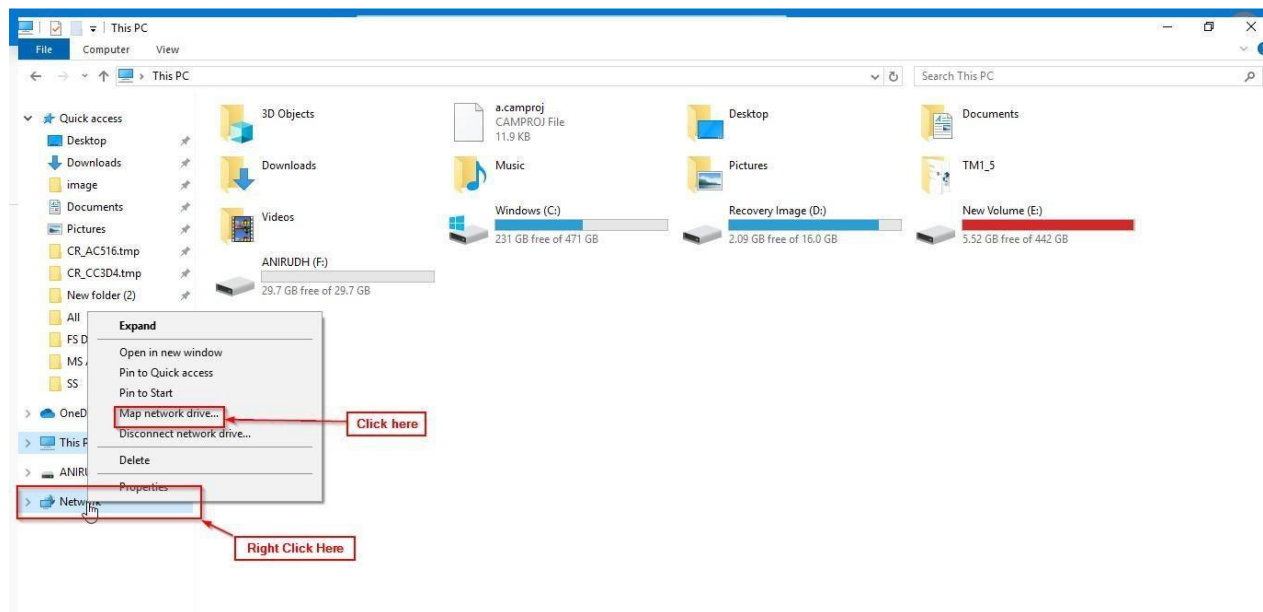
```
Test-NetConnection -ComputerName fsdemo1234.file.core.windows.net -Port 445
# Save the password so the drive will persist on reboot
Invoke-Expression -Command "cmdkey /add:fsdemo1234.file.core.windows.net /user:Azure\fsdemo1234
/pass:5b+fZUE1z5o6CDRC/ViFYptpKshhKnq4yJoMCPw8mHJIz2wPCDwofeKdhCGsjQ/g7b7KWLRnLtdCC1+kfX6hpw=="
# Mount the drive
New-PSDrive -Name Z -PSProvider FileSystem -Root '\\fsdemo1234.file.core.windows.net\demo'
```

Username

Password

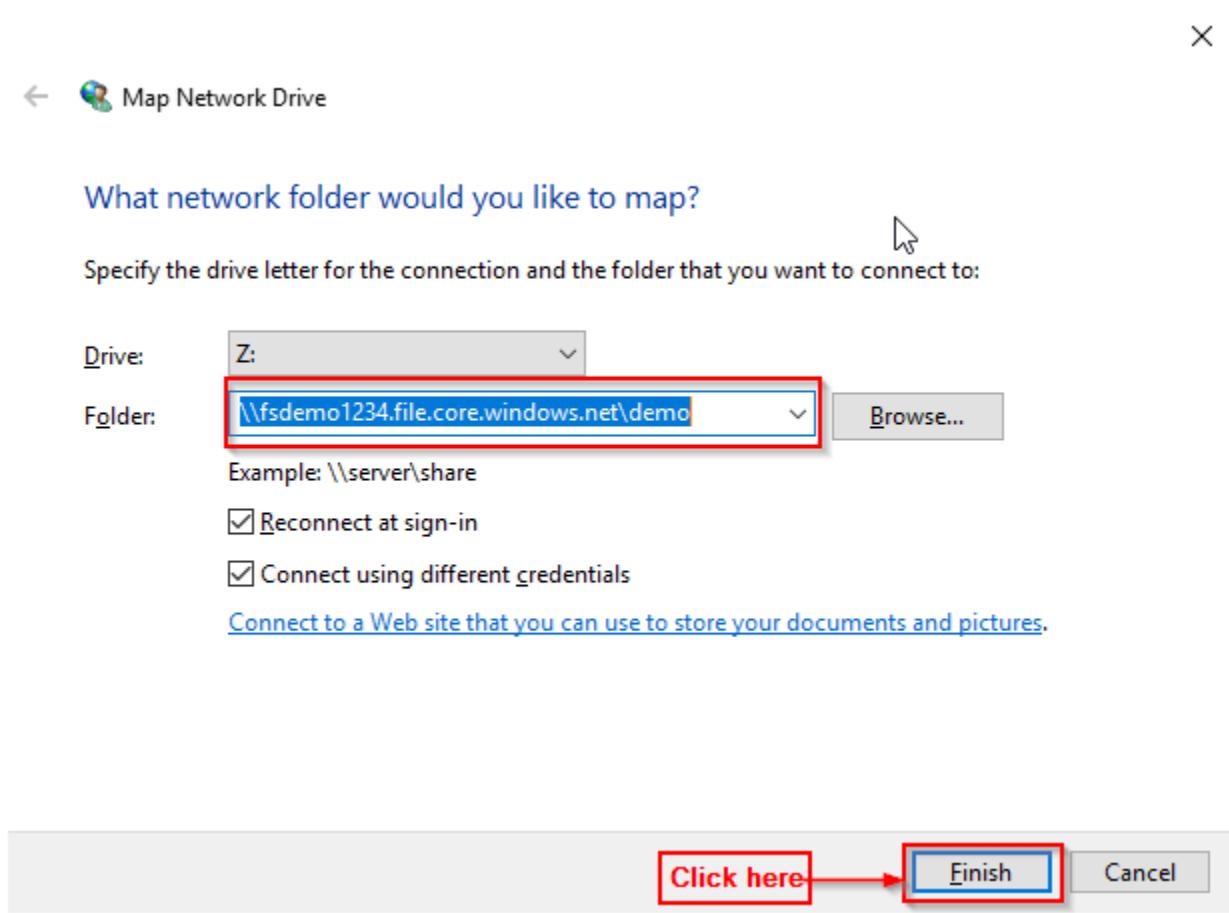
Folder URL

Step 5.4: Open Windows Explorer



Step 5.5: Right Click on Network and Select Map network drive

Step 5.6: In the folder section, paste the URL copied from the command



Map Network Drive

What network folder would you like to map?

Specify the drive letter for the connection and the folder that you want to connect to:

Drive: Z: ▼

Folder: \\fsdemo1234.file.core.windows.net\demo ▼ Browse...

Example: \\server\share

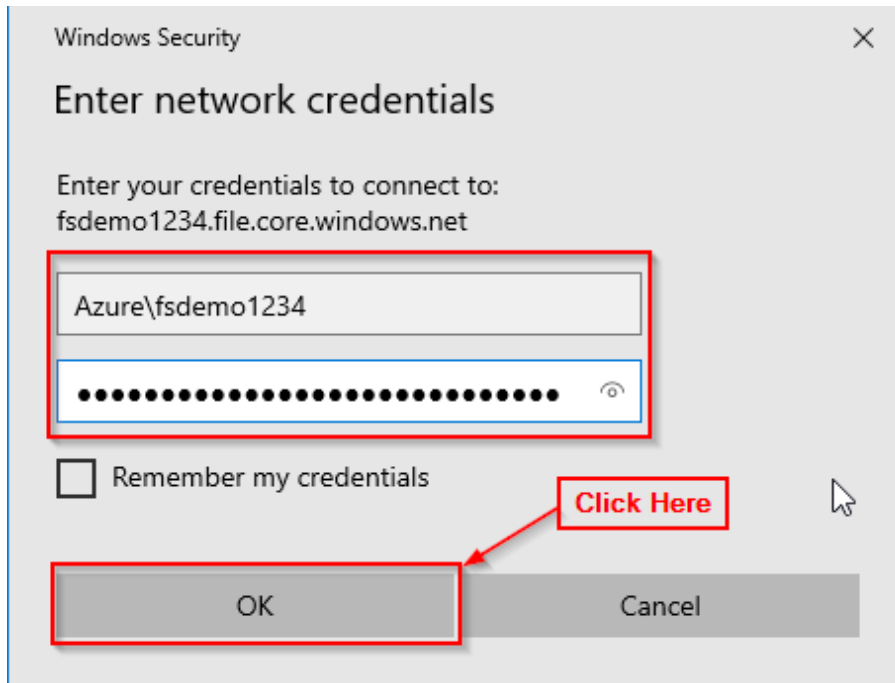
☒ Reconnect at sign-in

☒ Connect using different credentials

[Connect to a Web site that you can use to store your documents and pictures.](#)

Click here → Finish Cancel

Step 5.7: Enter username and password from the command and click OK



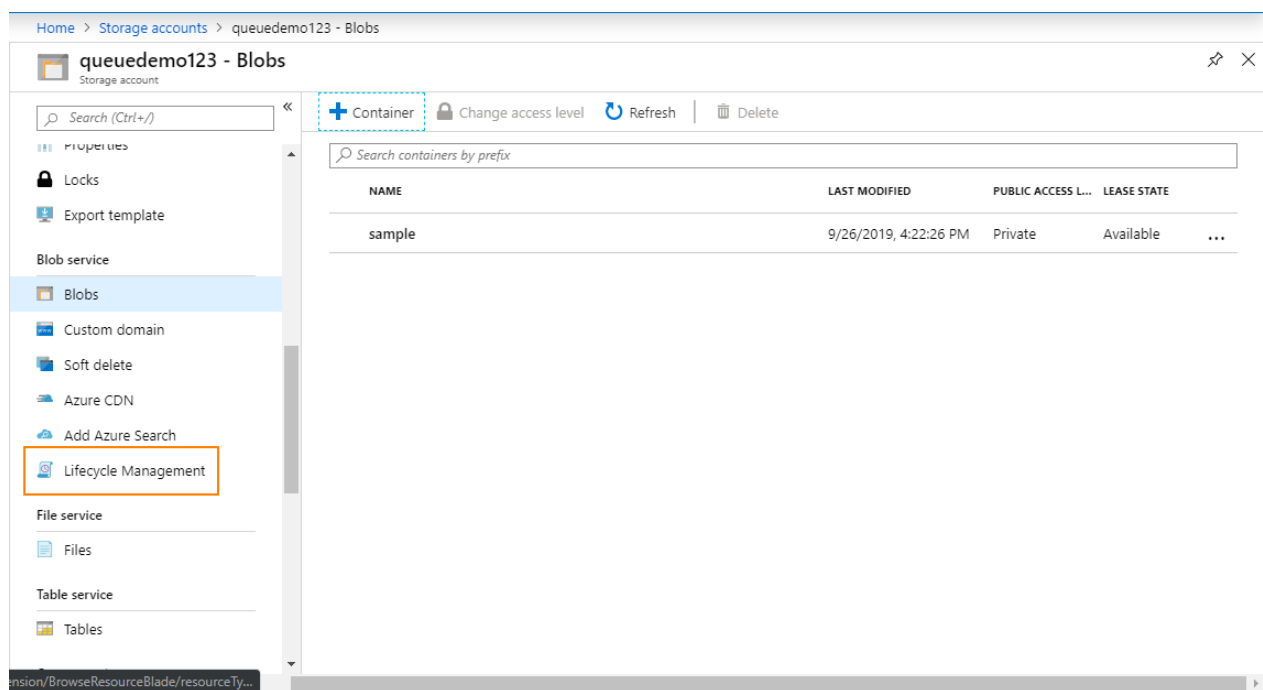
Issue 2: They wish to store large volumes of image data. They wish to have low latency access to frequently accessed images i.e. images that have been accessed in the last 14 days. If an image is not accessed within the last 14 days they wish to archive them.

Solution:

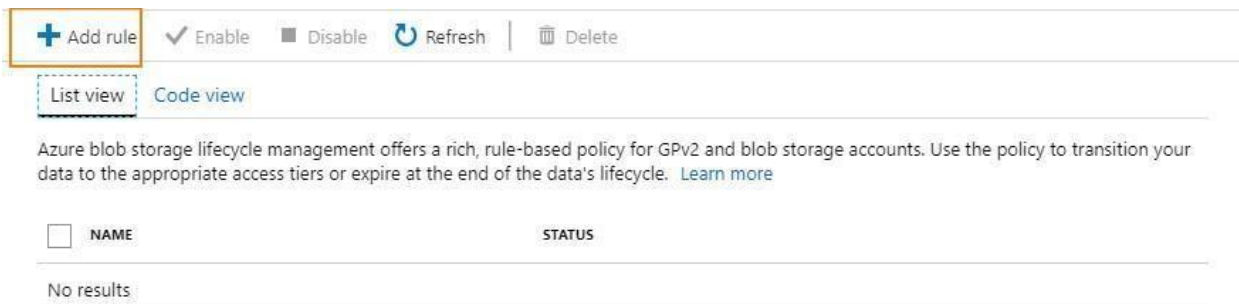
Step 1: Create a Blob container

Step 2: Add images to that Blob

Step 3: Under Blob Service in the sidebar, select Lifecycle Management



Step 4: Select Add rule



Step 5: Under the action set tab fill in the following details:

Step 5.1: Add rule name

Step 5.2: Select Move blob to archive storage

Step 5.3: In Days after last modification fill 14

Step 5.4: Click on Review + add

Action set Filter set Review + add

Each rule definition includes an action set and a filter set. The action set applies the tier or delete actions to the filtered set of objects. The filter set limits rule actions to a certain set of objects within a container or objects names.

* Rule name

Blobs

☐ Move blob to cool storage
Days after last modification

☒ Move blob to archive storage
Days after last modification

☐ Delete blob
Days after last modification

Review + add < Previous Next : Filter set >

Step 5.5: Click on Add

Home > Storage accounts > queuedemo123 - Lifecycle Management > Add a rule

Add a rule

✓ Validation passed

Action set Filter set Review + add

Action set	
Rule name	Sample
Status	Enabled
Blobs	Move to archive storage after 14 days after blob last modification.
Snapshots	None
Filter set	
Prefix match	None

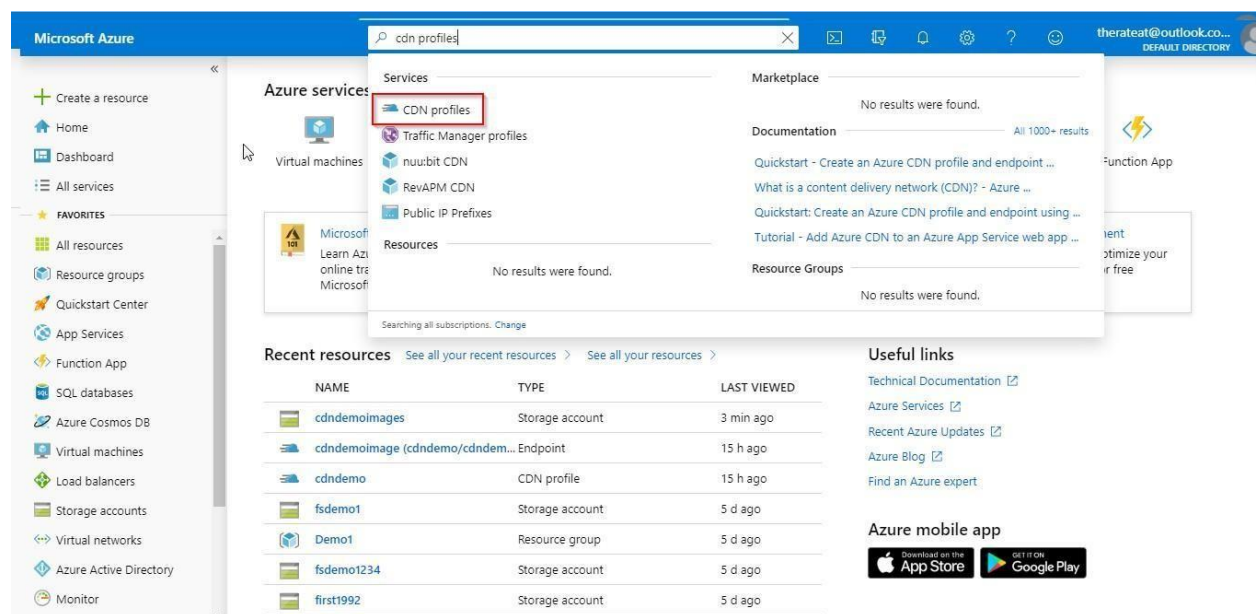
Add < Previous Next >

Issue 3: They wish to lower the latency of their website. They have noticed that users who are far away from their web server have complained that images take a lot of time to load.

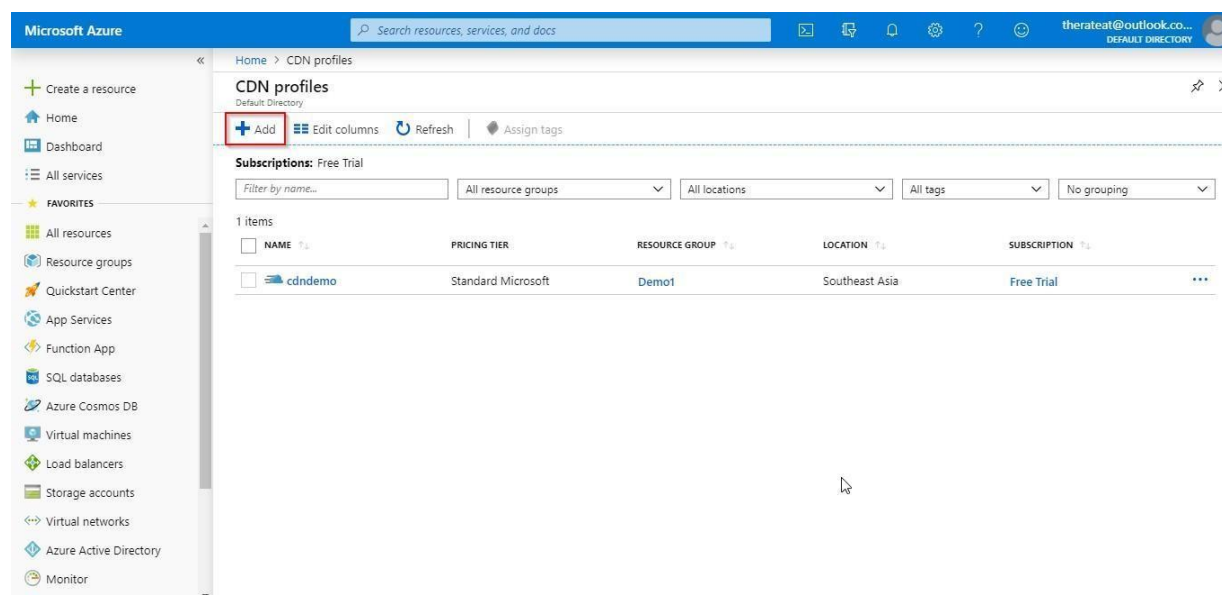
Solution:

Step 1: Create a storage account for Blob Storage and upload a few images

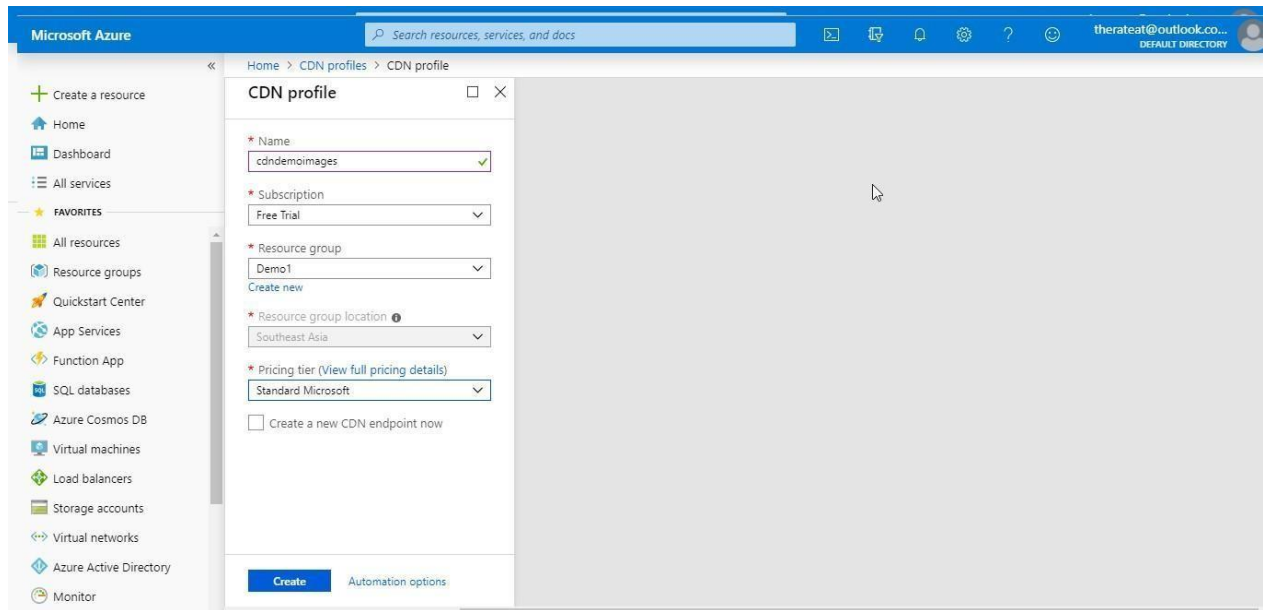
Step 2: In the Azure Portal's search for CDN Profiles and open it



Step 3: Click on + Add



Step 4: Fill the details and click on Create

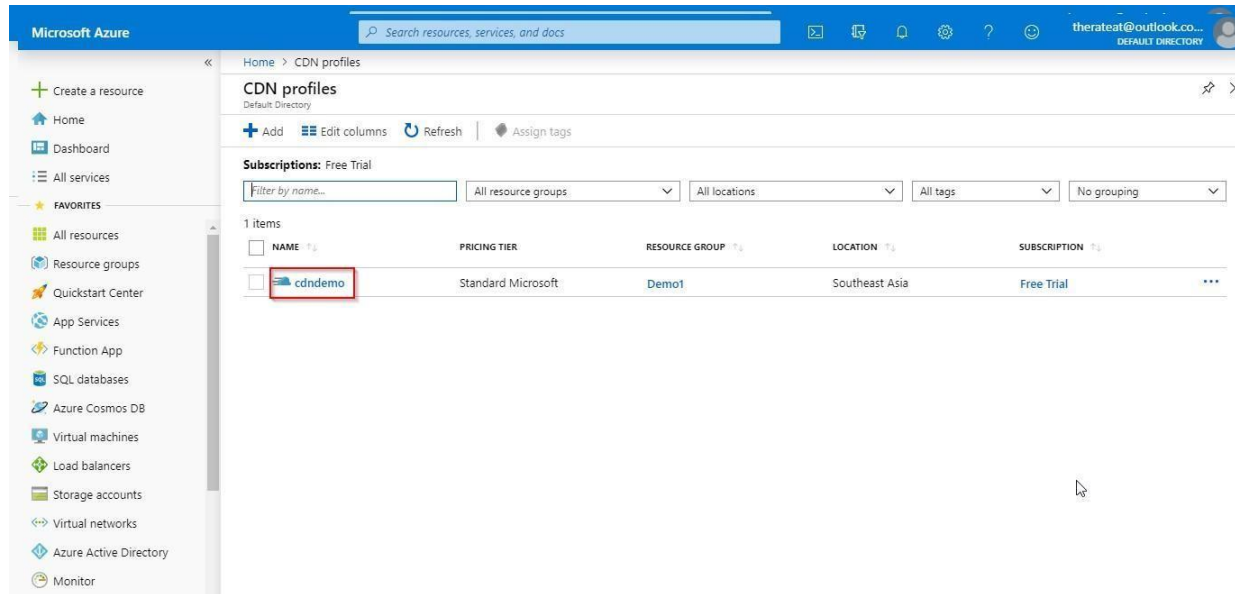


The screenshot shows the Microsoft Azure portal interface. The left sidebar contains navigation links: 'Create a resource', 'Home', 'Dashboard', 'All services', and a 'FAVORITES' section with links to 'All resources', 'Resource groups', 'Quickstart Center', 'App Services', 'Function App', 'SQL databases', 'Azure Cosmos DB', 'Virtual machines', 'Load balancers', 'Storage accounts', 'Virtual networks', 'Azure Active Directory', and 'Monitor'. The main content area is titled 'CDN profile' and contains the following fields:

- Name:** A text input field containing 'cdndemoimages' with a green checkmark.
- Subscription:** A dropdown menu showing 'Free Trial'.
- Resource group:** A dropdown menu showing 'Demo1' with a 'Create new' link below it.
- Resource group location:** A dropdown menu showing 'Southeast Asia'.
- Pricing tier (View full pricing details):** A dropdown menu showing 'Standard Microsoft'.
- Create a new CDN endpoint now:** An unchecked checkbox.

At the bottom of the form, there is a blue 'Create' button and a link for 'Automation options'.

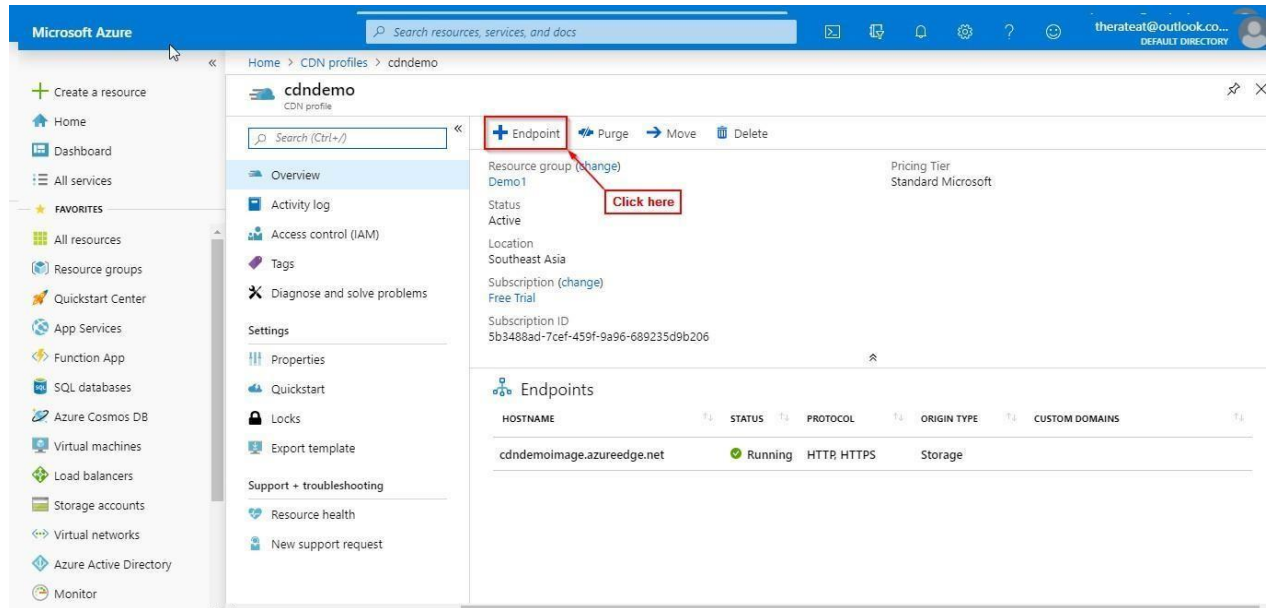
Step 5: Click on the CDN Profile just created



The screenshot shows the Microsoft Azure portal interface. The left sidebar contains navigation options like 'Create a resource', 'Home', 'Dashboard', 'All services', and 'FAVORITES'. The main content area is titled 'CDN profiles' and shows a table with one item, 'cdndemo', which is highlighted with a red box. The table columns are NAME, PRICING TIER, RESOURCE GROUP, LOCATION, and SUBSCRIPTION.

NAME	PRICING TIER	RESOURCE GROUP	LOCATION	SUBSCRIPTION
cdndemo	Standard Microsoft	Demo1	Southeast Asia	Free Trial

Step 6: Click on +Endpoint



Step 7: Enter the details:

Origin type : Storage

Origin hostname : URI of Blob storage created in previous step

Step 8: Click on add

The screenshot displays the Microsoft Azure portal interface. On the left, the navigation pane shows various services under 'FAVORITES'. The main area is titled 'Home > CDN profiles > cdndemo'. Below this, there's a search bar and a list of actions: '+ Endpoint', 'Purge', 'Move', and 'Delete'. The 'Overview' tab is selected, showing details for the 'cdndemo' CDN profile, including its resource group (Demo1), status (Active), location (Southeast Asia), and subscription ID. Below the overview, there's a table for 'Endpoints' with columns for 'HOSTNAME', 'STATUS', and 'PROTOCOL'. One endpoint is listed: 'cdndemoimage.azureedge.net' with a status of 'Running' and protocols 'HTTP, HTTPS'. On the right, the 'Add an endpoint' configuration pane is open. It contains several fields: 'Name' (cdndemoimg), 'Origin type' (Storage), 'Origin hostname' (cdndemoimages.blob.core.windows.net), 'Origin path' (/Path), 'Origin host header' (cdndemoimages.blob.core.windows.net), 'Protocol' (HTTP, HTTPS), and 'Origin port' (80, 443). The 'Optimized for' dropdown is set to 'General web delivery'. At the bottom of this pane, there's a blue 'Add' button and a link to 'Automation options'. A red box highlights the 'Add' button, and a red arrow points to it with the text 'Click Here'.

Issue 4: They wish to serve another website on

Azure's VMs.

Solution:

Step 1: Create a VM using Azure Portal (Make sure to enable HTTP and RDP protocol for connecting)

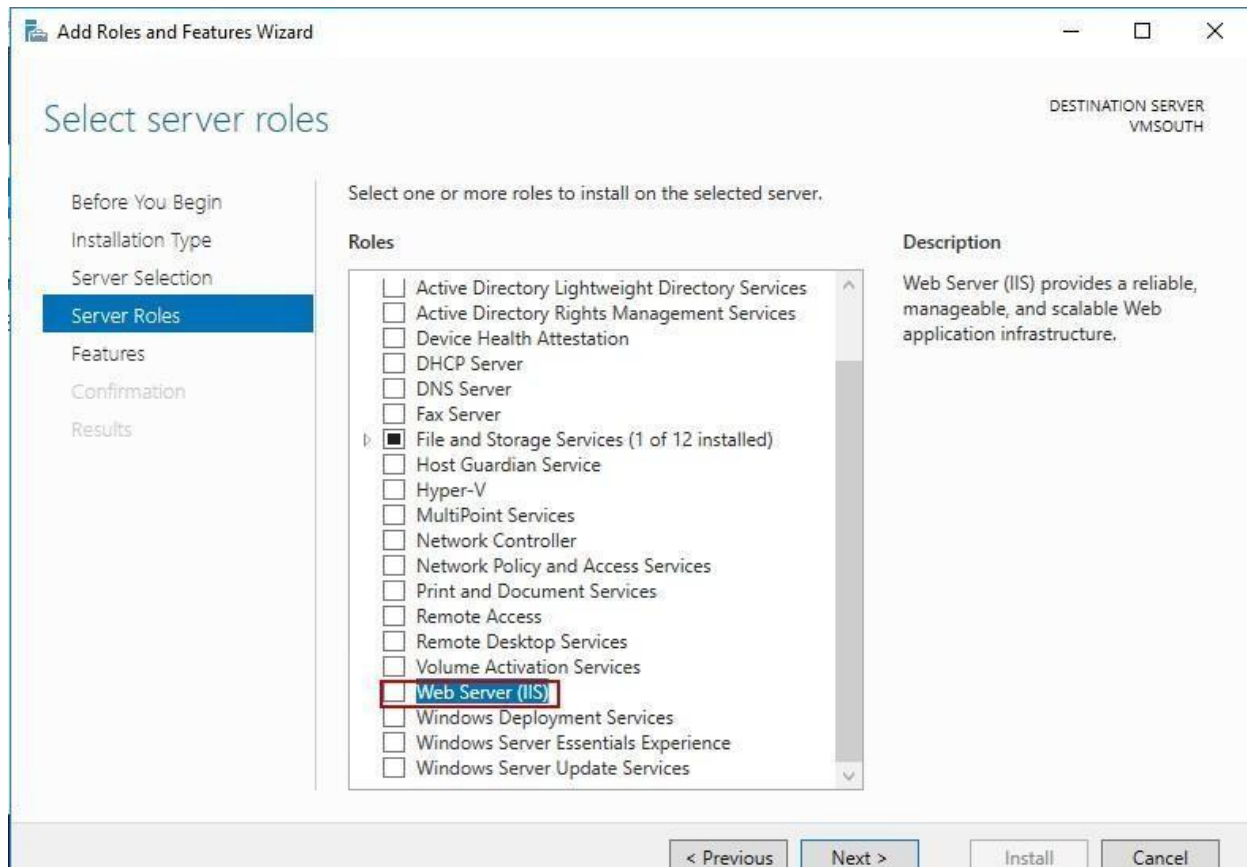
Step 2: Open the VM using RDP

Step 3: Open Server Manager

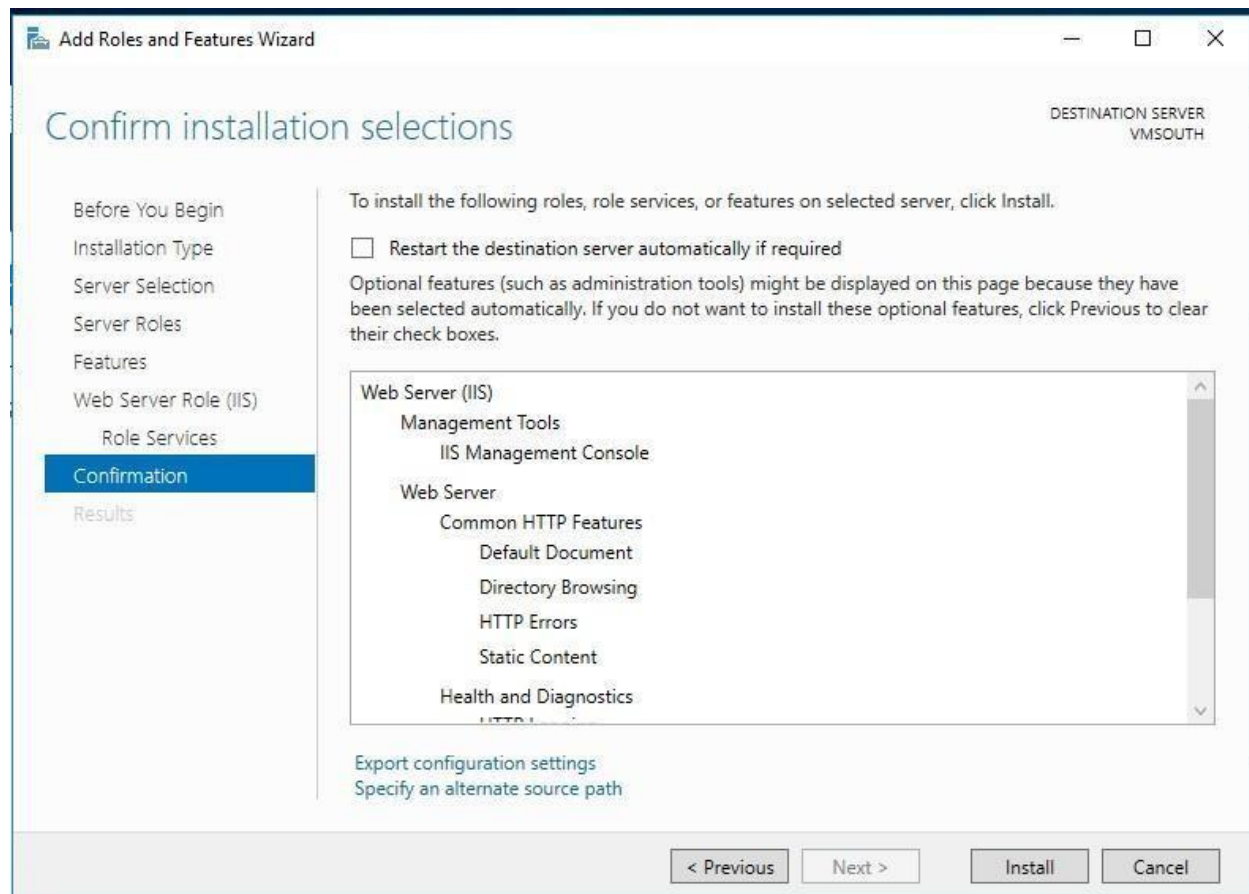
Step 4: Click on 'Add Roles and Features'



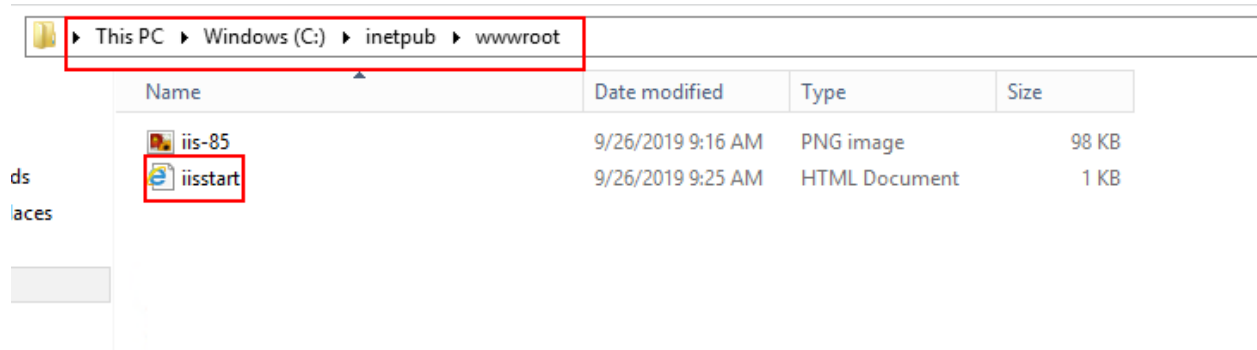
Step 5: In the wizard click on 'Next' 3 times and then in 'roles' select Web Server (IIS)



Step 6: Click on Next and Install



Step 7: After installation finishes Open 'C:\inetpub\wwwroot', this is the root folder of IIS

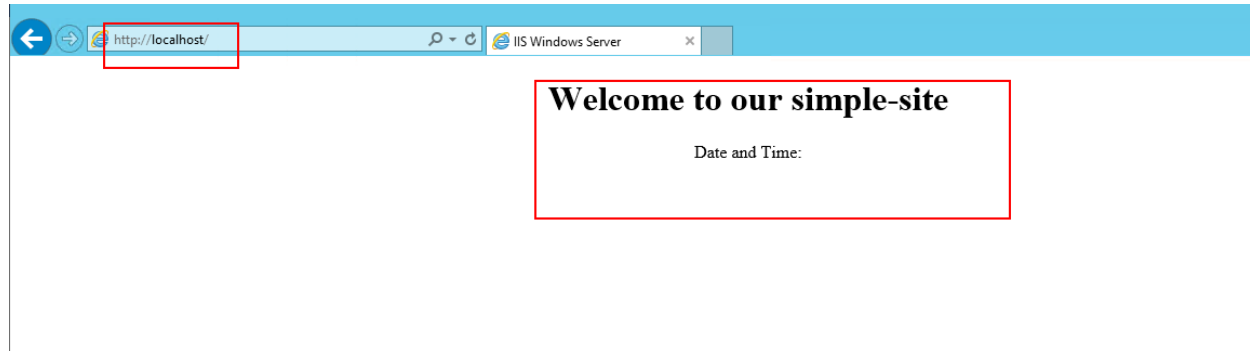


Step 8: Open iisstart.html in Notepad

Step 9: Make changes to the code by replacing the body tag with following code and save it

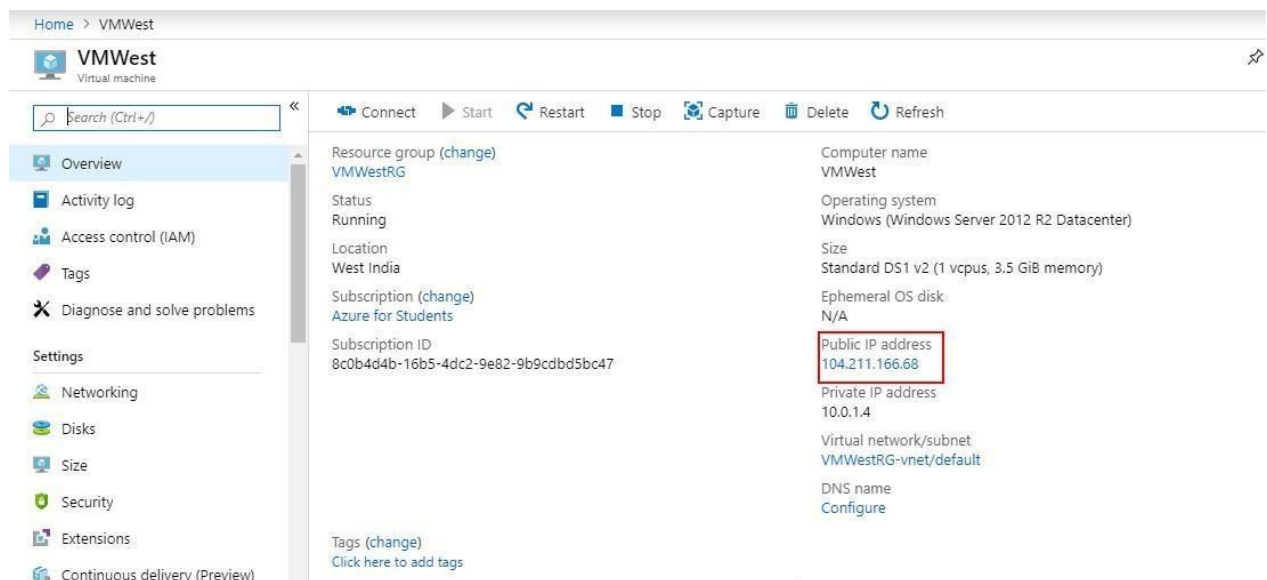
```
<body>
<div id="container">
<h1>Welcome to our simple-site</h1>
<span>Date and Time: </span>
<p id='time'></p>
</div>
<script>
    let el = document.getElementById('time');
    setInterval(() => el.innerHTML = (new Date()).toLocaleString(), 1000);
</script>
</body>
```

Step 10: Open localhost in Internet Explorer in VM to see if it works.



Step 11: Open the VM page in Azure Portal

Step 12: Copy the public IP Address of the VM and open it in the browser



Step 13: See that your simple-site is being served by the web server



Welcome to our simple-site

Date and Time:

9/26/2019, 4:42:13 PM

Issue 5: They also want to have two VMs in different networks. They wish for you to deploy those VMs and enable communication between them.

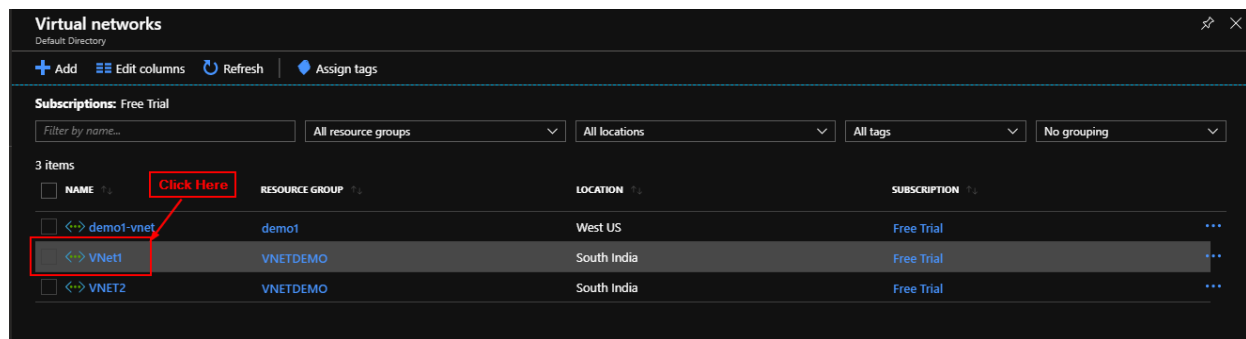
Solution:

Step 1: Create two VNets in the same region

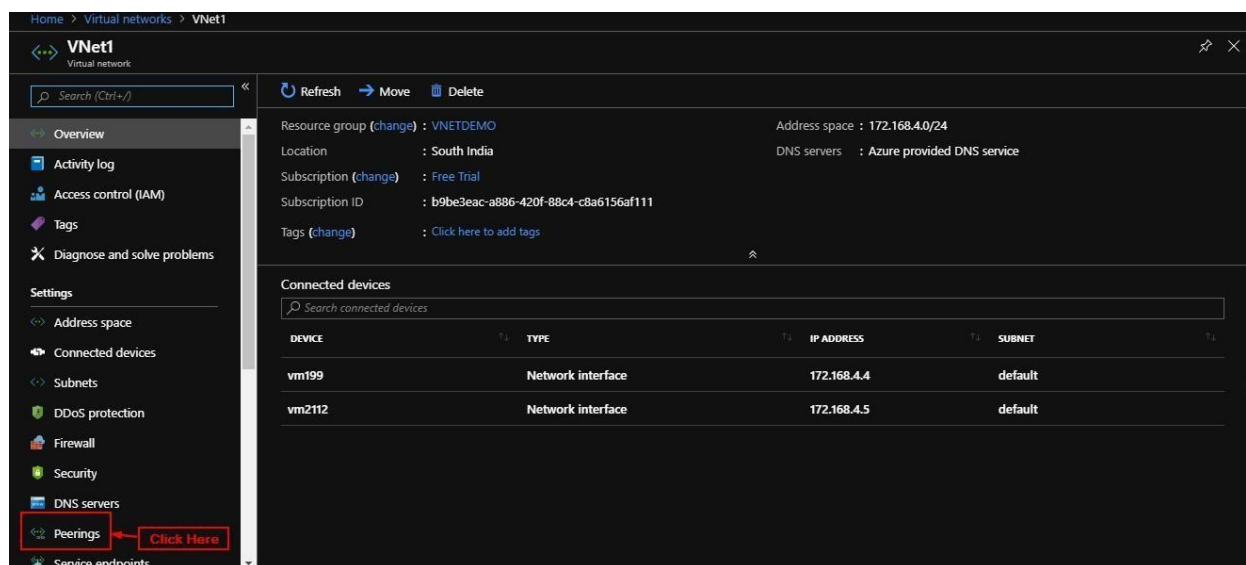
Step 2: Create two VMs, one in each separate VNet

Step 3: Open Virtual Network page in Azure Portal

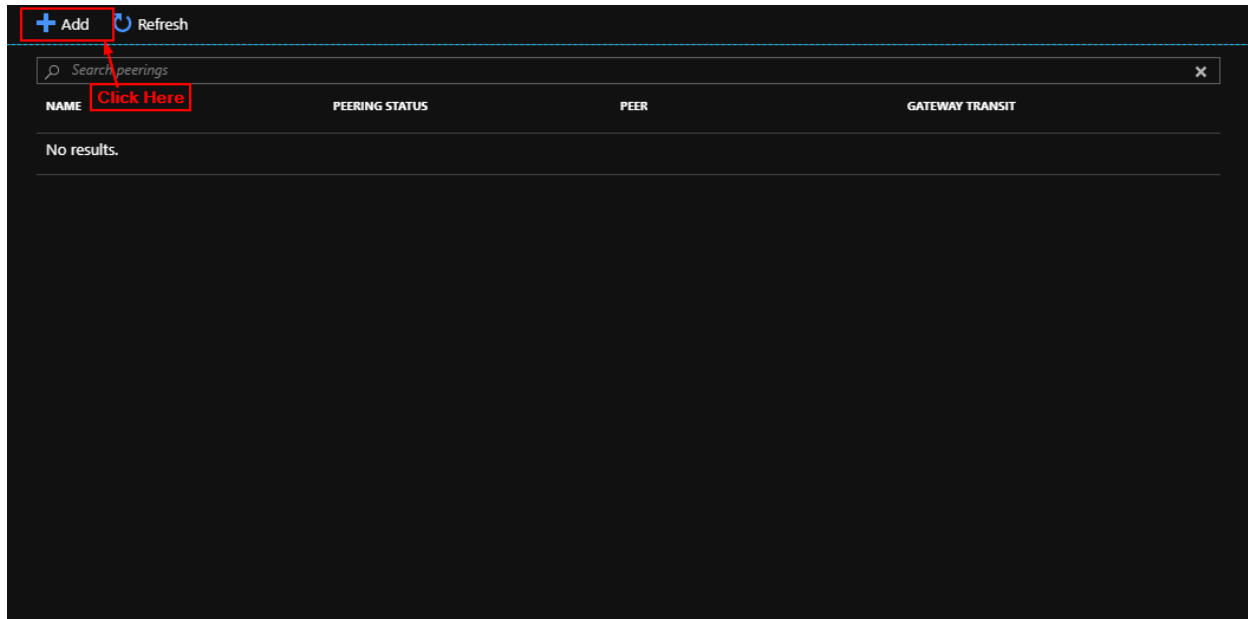
Step 4: Click and open one of the VNets created in Step 1



Step 5: In the sidebar click on Peerings



Step 6: Click on + Add



Step 7: Enter the details, make sure you add details for both peerings to enable bi-directional communication

The screenshot shows the 'Add peering' form in the Azure portal. The form is titled 'Add peering' and 'VNet1'. It contains several fields and sections. The 'Name of the peering from VNet1 to VNET2' field is filled with 'VNET1TOVNET2'. The 'Peer details' section shows 'Virtual network deployment model' with 'Resource manager' selected. The 'Subscription' dropdown is set to 'Free Trial'. The 'Virtual network' dropdown is set to 'VNET2 (VNETDEMO)'. The 'Name of the peering from VNET2 to VNet1' field is filled with 'VNETTOVNET1'. The 'Configuration' section is partially visible. At the bottom, there is an 'OK' button highlighted with a red box and a red arrow pointing to it from the text 'Click here'.

Step 8: Click on OK

Issue 6: They wish to use Azure to resolve their site with a domain of your choice to its IP address

Solution:

-- GET A FREE DOMAIN --

Step 1: Open and sign up at freenom.com

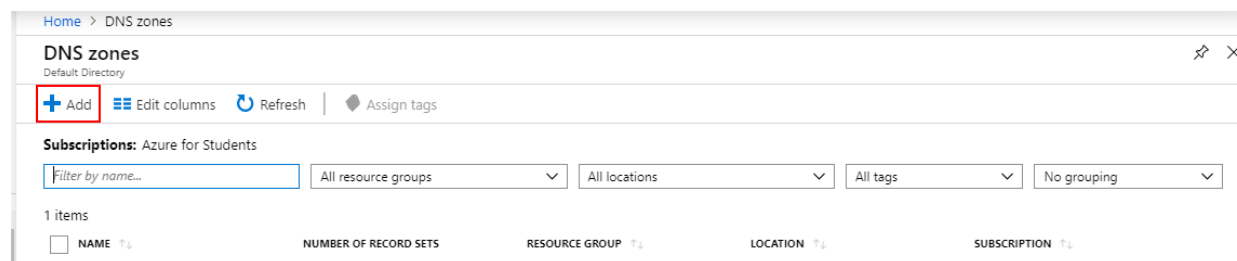
Step 2: Buy a free domain (for our purposes it can be anything)

simple-site.ml	26/09/2019	26/12/2019	ACTIVE	Free	Manage Domain
----------------	------------	------------	--------	------	---------------

-- Configure Azure DNS --

Step 1: In the Azure Portal search for DNS Zones and open it

Step 2: Click on + Add

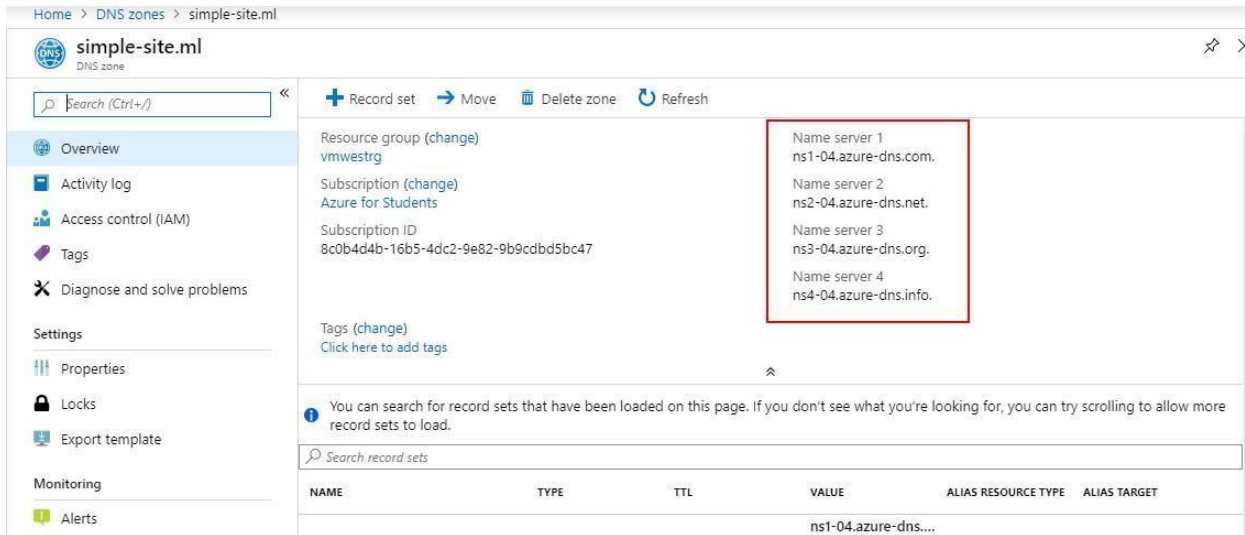


Step 3: Enter the following details and click on 'Review + create'

- Resource Group: Your Resource Group.
- Name: you domain name e.g. 'simple-site.ml'

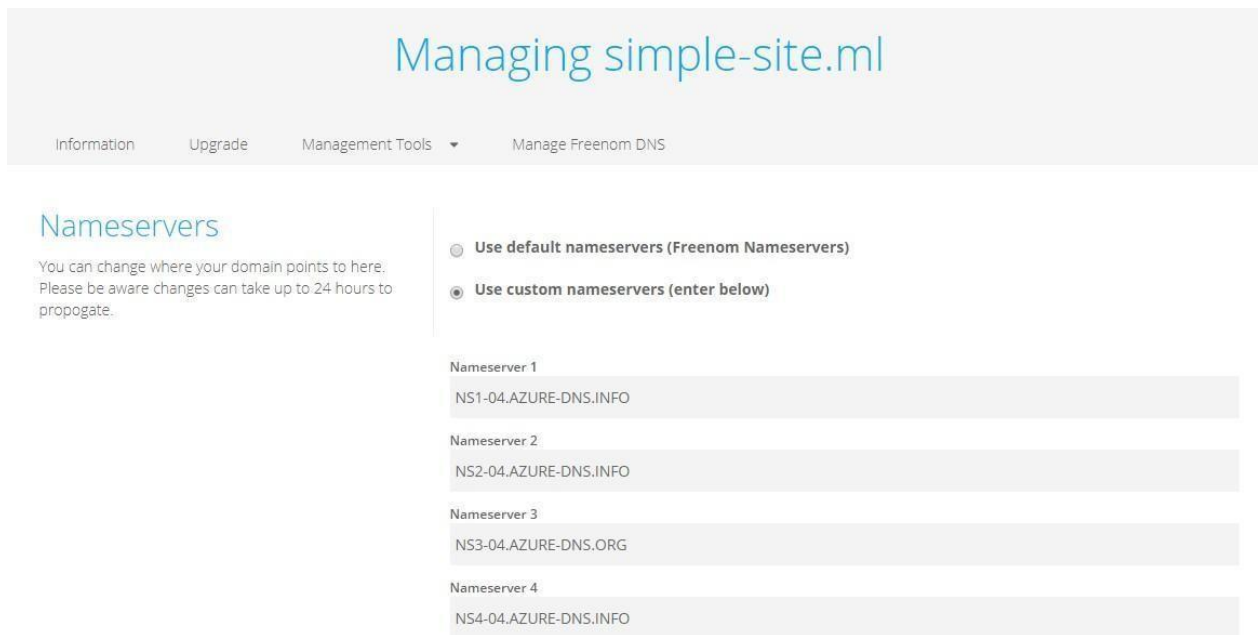
Step 4: Click on Create

Step 5: Open the DNS Zone and copy Name Server Addresses

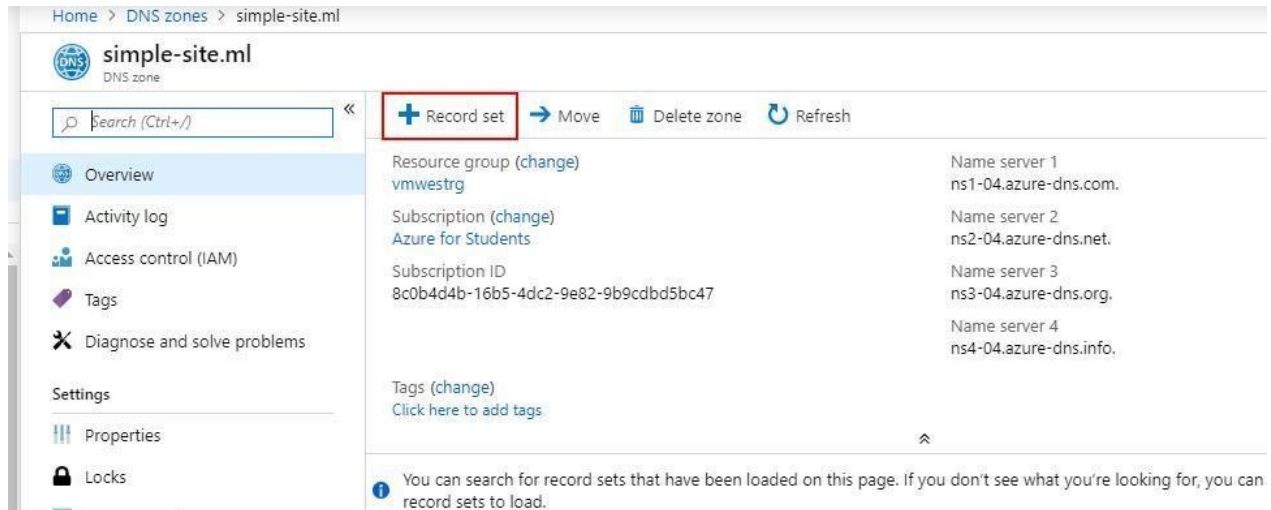


Step 6: Open your domain name provider's admin panel

Step 7: Change the name server addresses to the Azure Name Servers



Step 8: Open the DNS Zone and click on '+ Record Set'



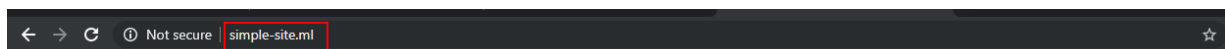
Step 9: Enter details:

- Name: www.
- Type: A

-IP Address as the Public IP of the VM on which app is deployed (Public IP of VM Created in solution of issue #4).

Step 10: Click on OK

Step 11: Now open the domain



Welcome to our simple-site

Date and Time:

9/26/2019, 4:52:24 PM

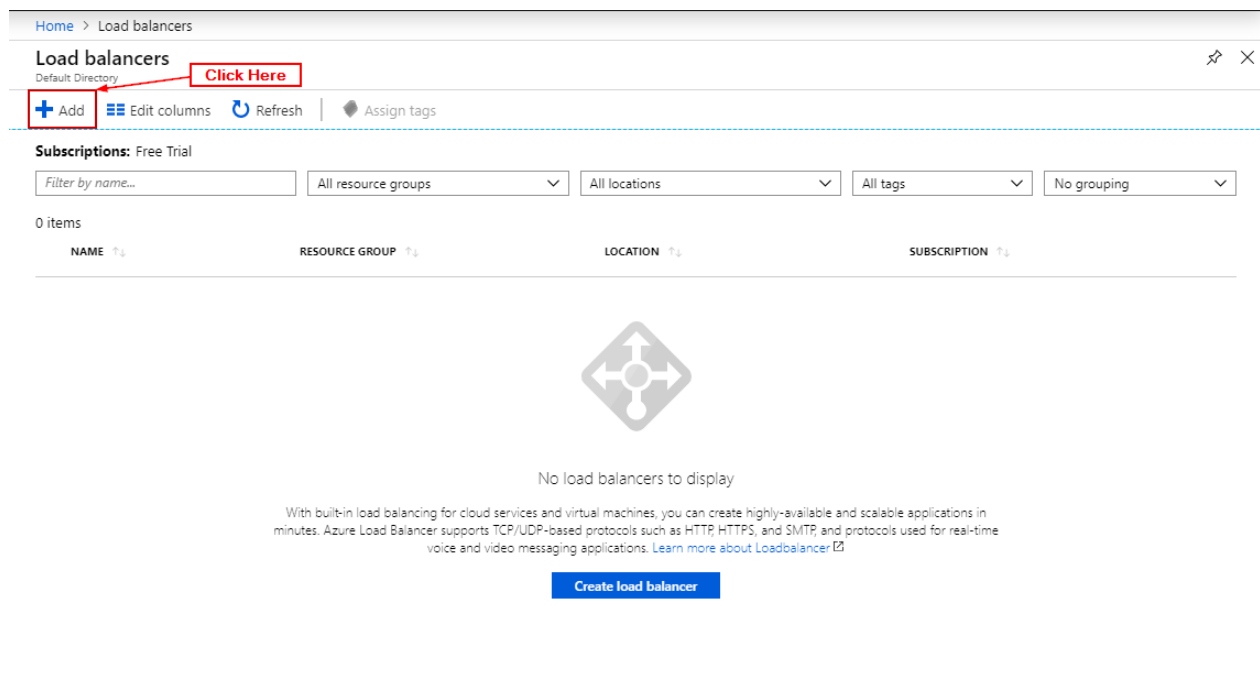
Issue 7: They wish for both the VMs serving their website to be more reliable so that if one VM fails the traffic is automatically routed to the other one.

Solution:

Step 1: Create two virtual machines within a single availability set and same virtual network

Step 2: Search Load Balancer in Azure Portal and click on it

Step 3: Click on + Add



Step 4: Add details and click Review + create

Home > Load balancers > Create load balancer

Create load balancer

* Subscription: Free Trial

* Resource group: LoadBalancerDEMO [Create new](#)

Instance details

* Name: load-balancer ✓

* Region: (Asia Pacific) South India

* Type: ☐ Internal ☒ Public

* SKU: ☒ Basic ☐ Standard

Public IP address

* Public IP address: ☒ Create new ☐ Use existing

* Public IP address name: load-balancer-ip ✓

Public IP address SKU: Basic

* Assignment: ☒ Dynamic ☐ Static

[Review + create](#) [< Previous](#) [Next : Tags >](#) [Download a template for automation](#)

Click Here

Step 5: Click on Create

Home > Load balancers > Create load balancer

Create load balancer

✓ Validation passed

[Basics](#) [Tags](#) [Review + create](#)

Basics

Subscription	Free Trial
Resource group	LoadBalancerDEMO
Name	load-balancer
Region	(Asia Pacific) South India
SKU	Basic
Type	Public
Public IP address	load-balancer-ip

Tags

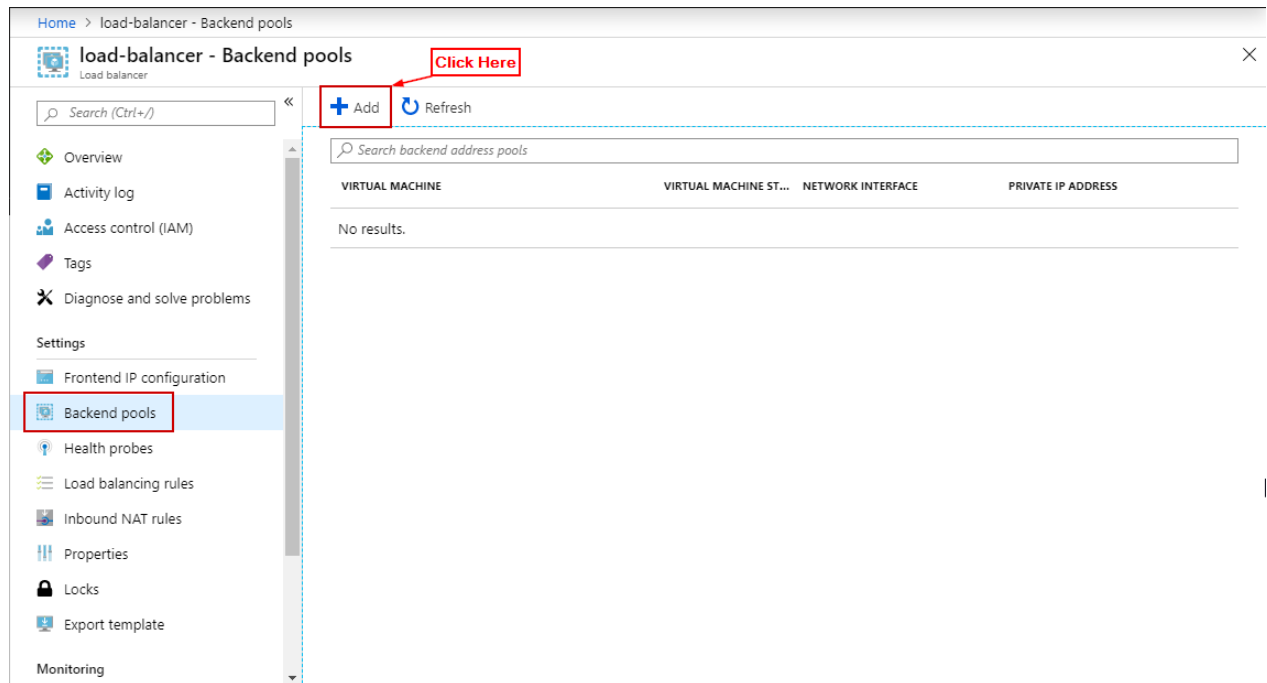
None

[Create](#) [< Previous](#) [Next >](#) [Download a template for automation](#)

Click Here

Step 6: Open the load balancer

Step 7: Click on backend pools and click on + Add



Step 8: Enter the details (availability Set and in target IP Configuration, add both

V

* Name
backend-pool ✓

IP version
IPv4 IPv6

Associated to ⓘ
Availability set ✓

Availability set ⓘ
LoadBalancerAS
number of virtual machines: 2 ✓

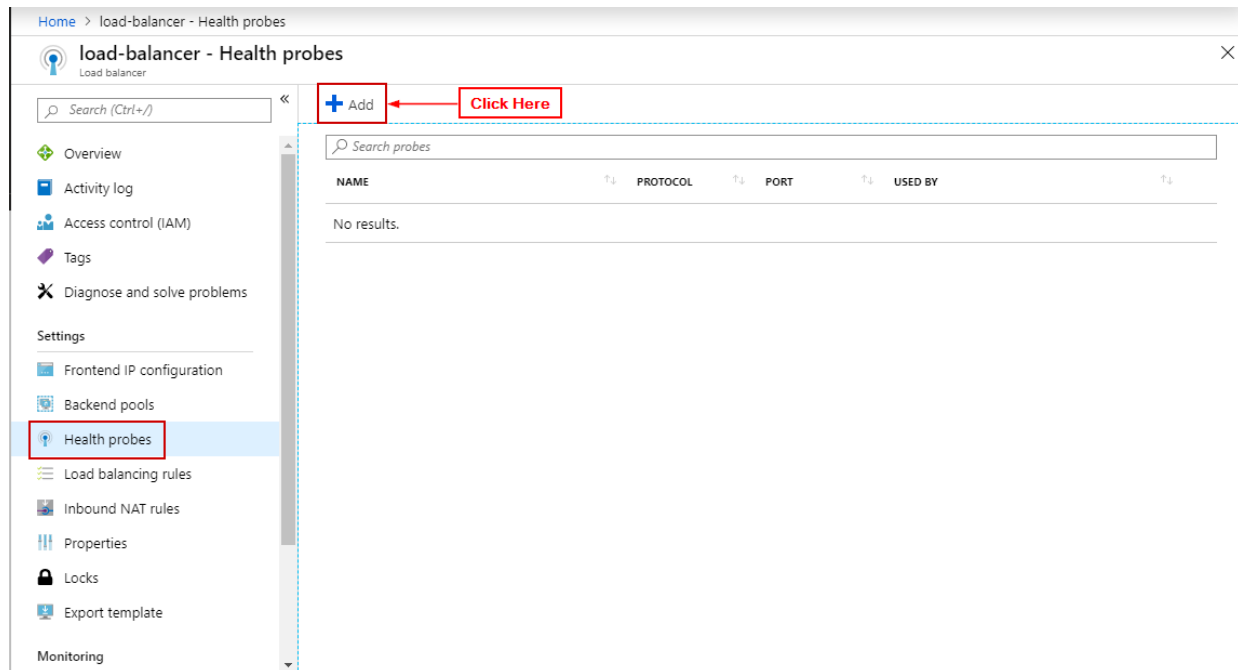
Target network IP configurations
Only VMs within the current availability set can be chosen. Once a VM is chosen, you can select a network IP configuration related to it.

Virtual machine: VM1 Network IP configuration: vm1232/ipconfig1 (10.0.0.4)	🗑️
* Target virtual machine ⓘ VM2 size: Standard_DS1_v2, network interfaces: 1	🗑️
* Network IP configuration ⓘ ipconfig1 (10.0.0.5)	✓

+ Add a target network IP configuration

OK ← Click Here

Step 9: In load balancer click on health probes and click on + Add



Step 10: Enter details click on OK

Home > load-balancer - Health probes > Add health probe

Add health probe

load-balancer

* Name ✓

IP version
IPv4

Protocol ⓘ
TCP ▼

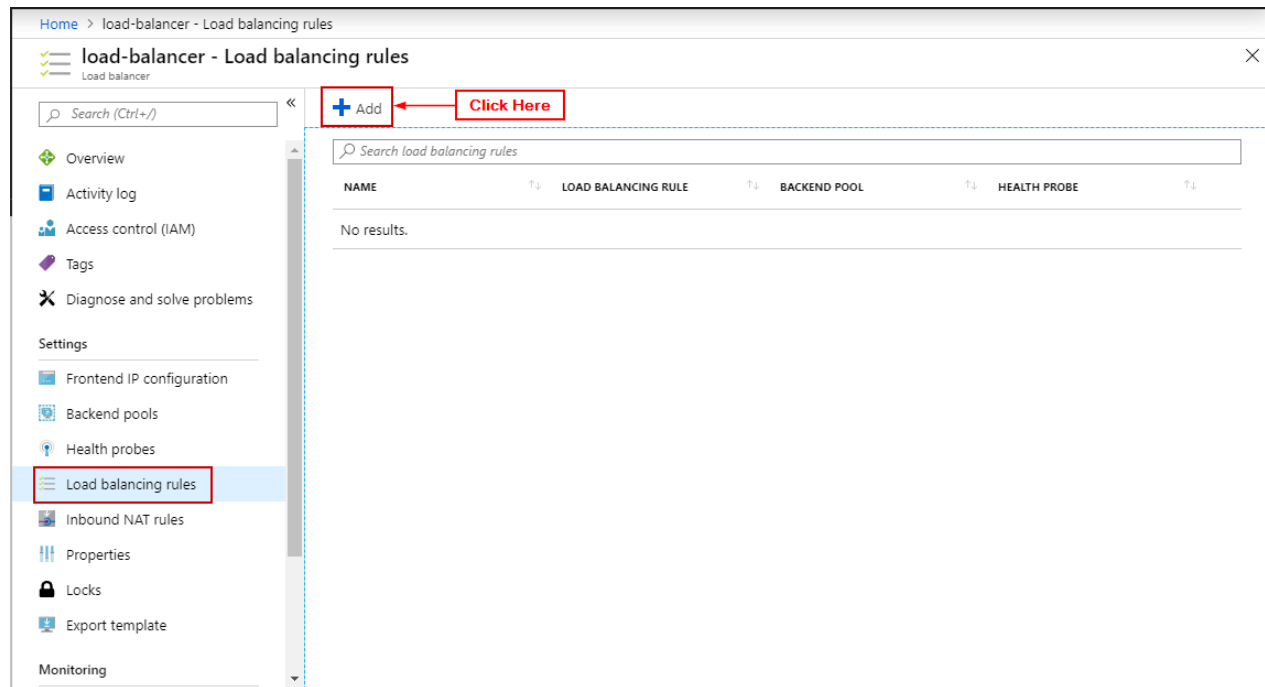
* Port ⓘ

* Interval ⓘ
 seconds

* Unhealthy threshold ⓘ
 consecutive failures

←

Step 11: In load balancer click on Load Balancing Rule and click on + Add



Step 12: Enter details click on OK

Home > load-balancer - Load balancing rules > Add load balancing rule

Add load balancing rule

load-balancer

* Name
load-balancing-rule ✓

* IP Version
☒ IPv4 ☐ IPv6

* Frontend IP address ⓘ
LoadBalancerFrontEnd ▼

Protocol
☒ TCP ☐ UDP

* Port
80

* Backend port ⓘ
80

Backend pool ⓘ
backend-pool (2 virtual machines) ▼

Health probe ⓘ
health-probe (TCP:80) ▼

OK ← **Click Here**

Step 13: Open load balancer and open the public IP Address

→ Move

🗑️ Delete

🔄 Refresh

Resource group (change)
LoadBalancerDEMO

Location
South India

Subscription (change)
Free Trial

Subscription ID
b9be3eac-a886-420f-88c4-c8a6156af111

SKU
Basic

Tags (change)
[Click here to add tags](#)

Backend pool
backend-pool (2 virtual machines)

Health probe
health-probe (Tcp:80)

Load balancing rule
load-balancing-rule (Tcp/80)

NAT rules
0 inbound

Public IP address
104.211.204.139 (LoadBalancerFrontEnd)

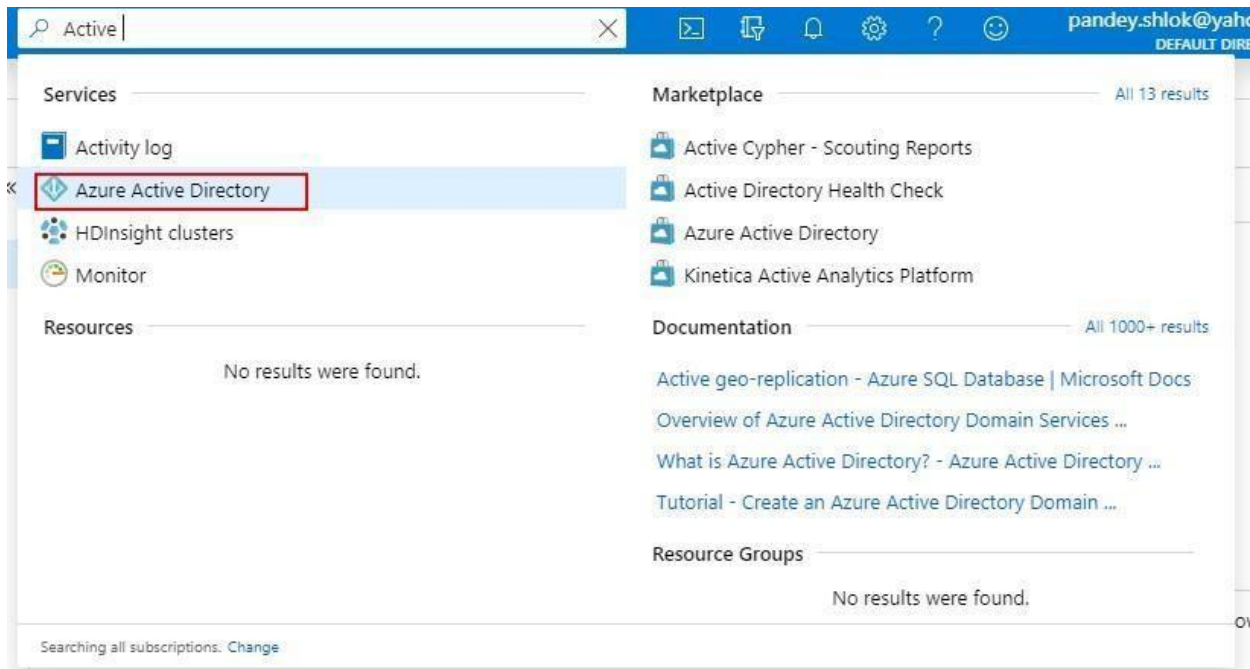
⌵

Open this IP

Issue 8: They wish for you to find a way to assign and manage credentials for Azure for all 10 employees in the company

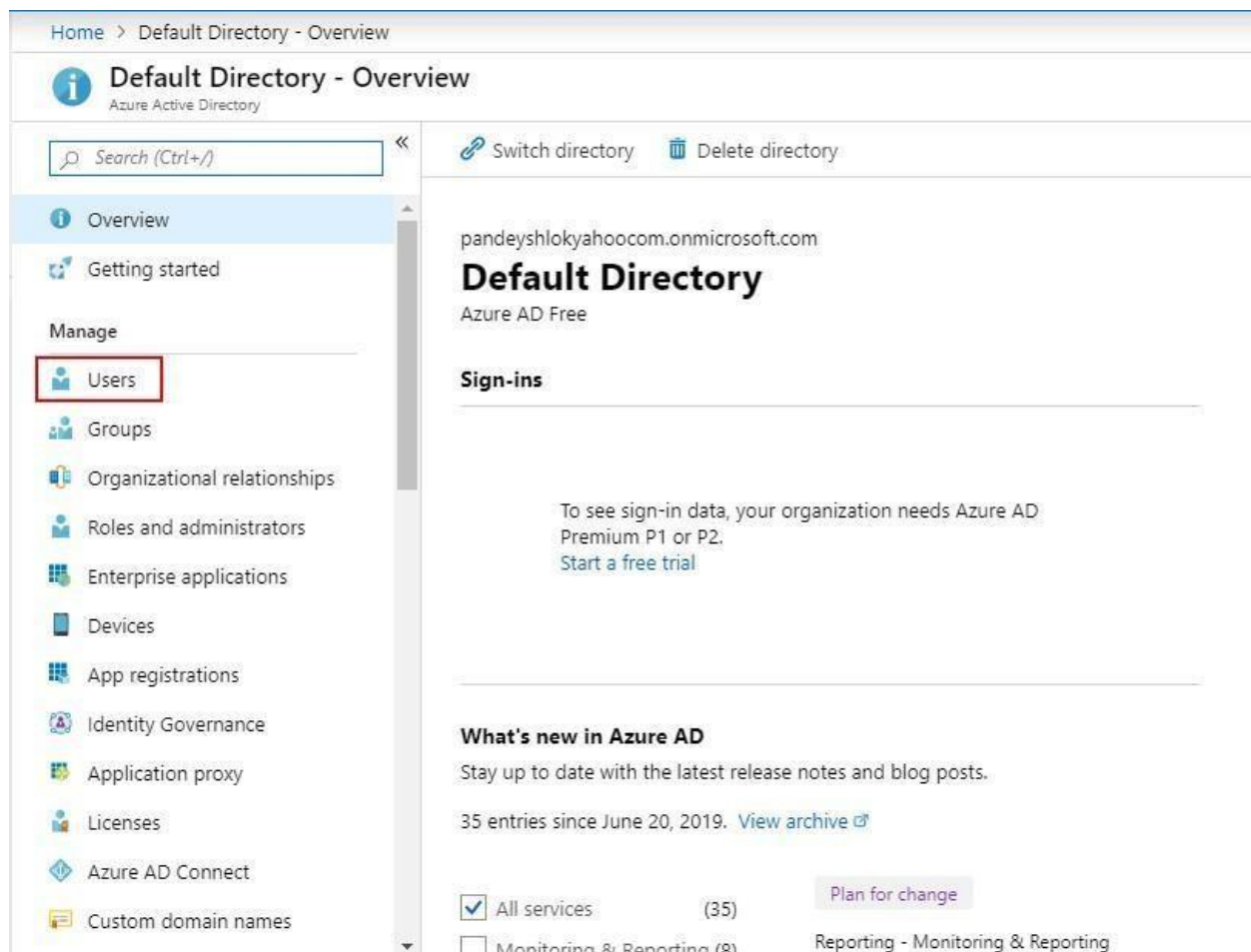
Solution:

Step 1: In Azure Portal search for Azure Active Directory

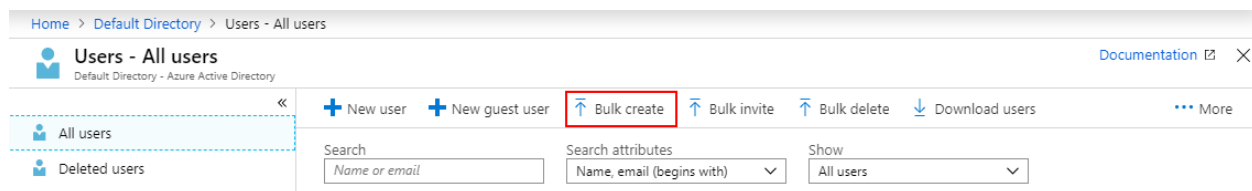


Step 2: Click on it and open it

Step 3: In the sidebar, click on Users



Step 4: Click on Bulk Create



Step 5: Click on Download CSV Template

Bulk create user (Preview)


Default Directory - Azure Active Directory

1. Download csv template (optional)

Download

2. Edit your csv file

3. Upload your csv file

Select a file 

[Learn more about bulk import users](#)

Submit

Step 6: Edit the CSV Template and add details of users

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
version:v1.0																	
Name (ex: User name) Initial password First name Last name Job title [optional] Department Block sign Usage loci Street address State or province Country or region Office phone City [city] ZIP or postal code Office phone Mobile phone [mobile]																	
John Doe	john@par	abcd@123	John	Doe	Sample User	Sample	No	India									
Jane Doe	Jane@par	abcd@123	Jane	Doe	Sample User	Sample	No	India									

Step 7: Upload the CSV File


Bulk create user (Preview) ×
Default Directory - Azure Active Directory

1. Download csv template (optional)

Download

2. Edit your csv file

3. Upload your csv file



File uploaded successfully

[Learn more about bulk import users](#)

Submit

Step 8: Click on Submit

Step 9: In the sidebar click on Bulk Operation Details to view the status

Refresh Help Columns

Got a second? We would love your feedback on Bulk operations →




File name: Type:

FILE NAME	UPLOAD TIME	COMPLETION TI...	STATUS	# SU...	# FA...	TOTAL R...	ADMIN UPLOAD...	TYPE
UserCreateTemplate...	9/26/2019, 5:...	Not completed	Processing	0	0	0	pandey.shlok...	user create ...

Step 10: After the operation is successful, open the User page to view the created users

+ New user + New guest user ↑ Bulk create ↑ Bulk invite ↑ Bulk delete ↓ Download users ... More

Search: Search attributes: Show:

NAME	USER NAME	USER TYPE	SOURCE
 Jane Doe	jane@pandeyshlokyahooom....	Member	Azure Active Directory
 John Doe	john@pandeyshlokyahooom....	Member	Azure Active Directory
 Vipul Shah	admin@pandeyshlokyahoooc...	Member	Microsoft Account

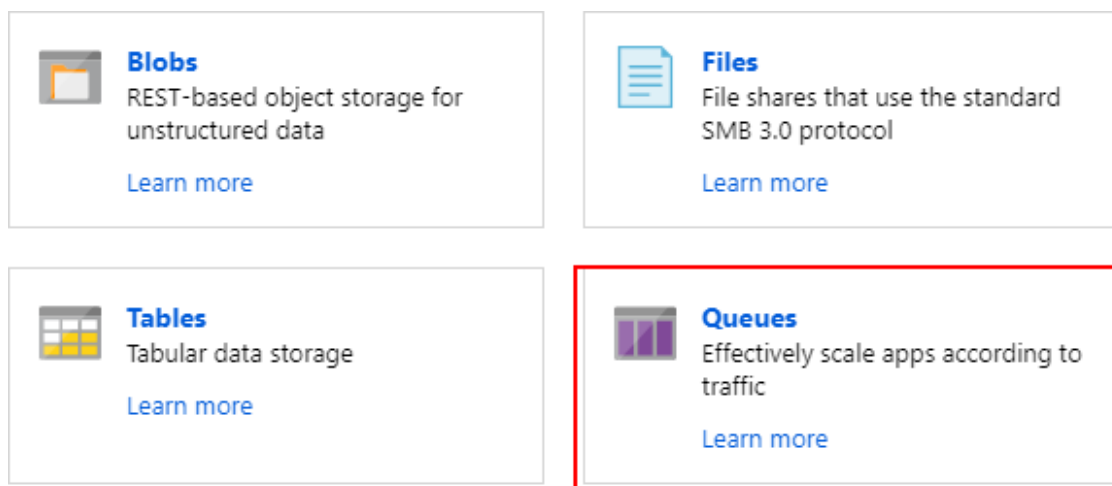
Issue 9: Finally they have two applications that need to pass messages between one another on an on-demand basis i.e. an application will send the message and another application will receive and process it when it can. You need to set up a service in such a way that these applications can do so (you are provided with the code). All you need to do is make changes to the config.js file.

Solution:

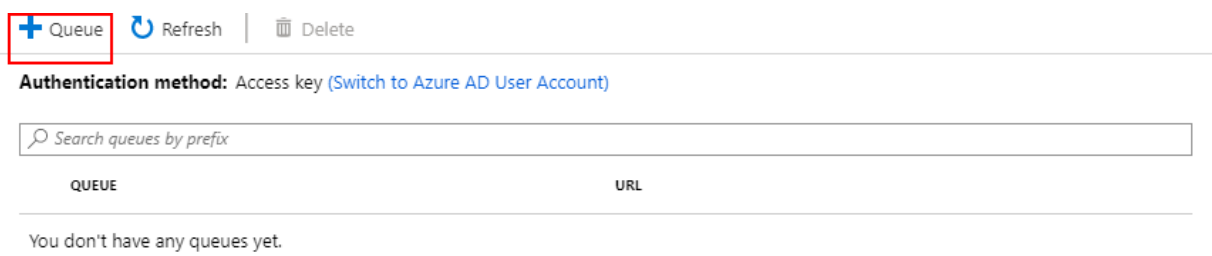
-- Make sure you have node js installed and run npm install in the project directory –

Step 1: Open Storage Account.

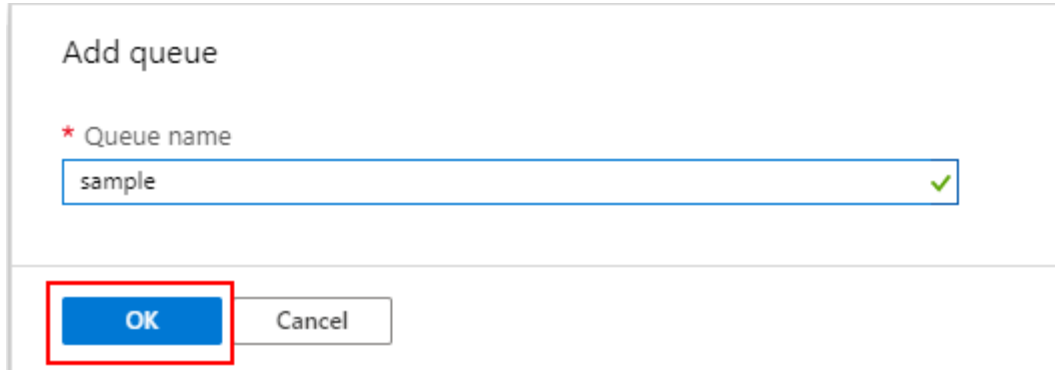
Step 2: Click on Queues.



Step 3: Click on + Queue



Step 4: Enter details and click OK



Add queue

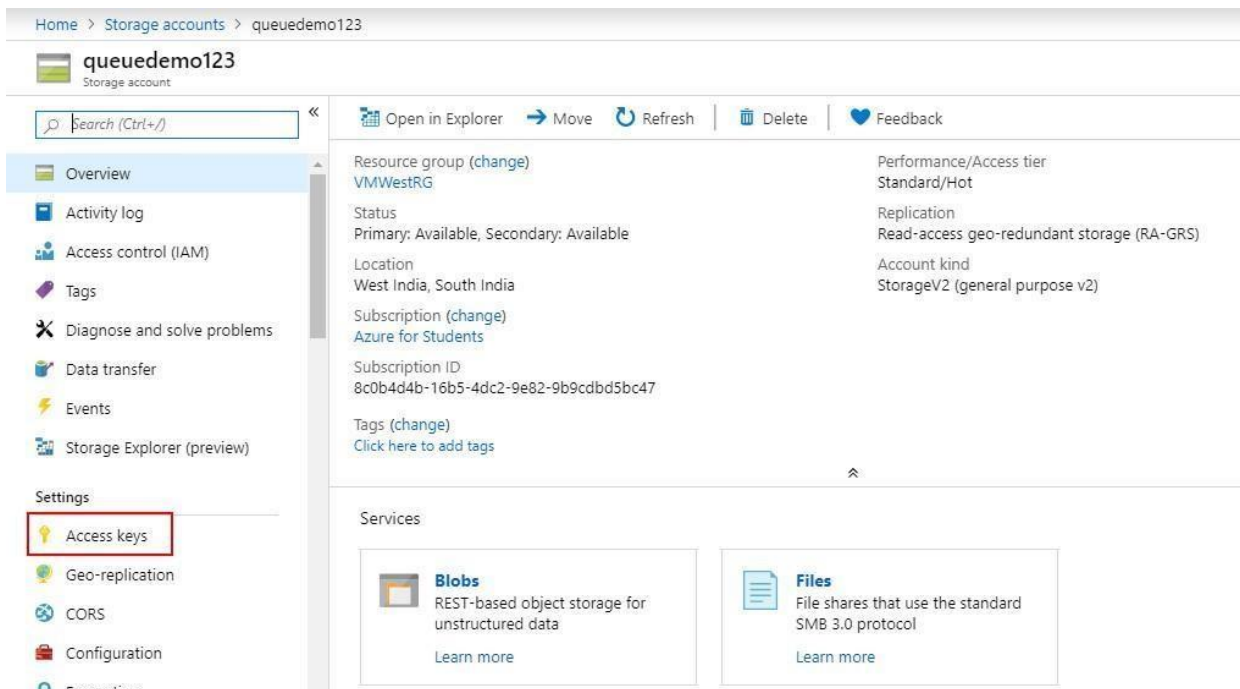
* Queue name

sample ✓

OK Cancel

Step 5: Open the Storage Account

Step 6: In the Settings section click on Access Keys



Home > Storage accounts > queuedemo123

queuedemo123
Storage account

Search (Ctrl+J)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Settings

Access keys

Geo-replication

CORS

Configuration

Encryption

Open in Explorer Move Refresh Delete Feedback

Resource group (change)
VMWestRG

Status
Primary: Available, Secondary: Available

Location
West India, South India

Subscription (change)
Azure for Students

Subscription ID
8c0b4d4b-16b5-4dc2-9e82-9b9cddb5bc47

Tags (change)
Click here to add tags

Performance/Access tier
Standard/Hot

Replication
Read-access geo-redundant storage (RA-GRS)

Account kind
StorageV2 (general purpose v2)

Services

Blobs
REST-based object storage for unstructured data
Learn more

Files
File shares that use the standard SMB 3.0 protocol
Learn more

Step 7: Copy one of the keys



Step 8: Open config.js file in the sample app.

Step 9: Replace information:

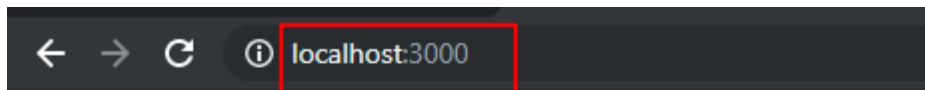
- queueName: Name of your Queue you just created.
- azureStorageAccount: Name of your Storage Account.
- azureStorageAccessKey: Access Key for your storage account.

```
module.exports = {  
  queueName: 'sample',  
  azureStorageAccount: 'queuedemo123',  
  azureStorageAccessKey: 'Nyb1zpwQbMNyb/  
eGrv4qwkjVxExPu0CNM1x3nnZC0vLCMgD7q0CiLPDF9YscEBHLGm3D+o17WOBY7VCIF4KlfQ==',  
};
```

Step 10: Run the app to check if it works (To run the app run the command: node server.js)

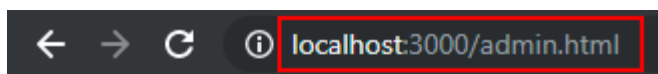
```
Intellipaat-Team@DESKTOP-FPNPKTE MINGW64 ~/Desktop/Storage Queue Demo
$ node server.js
Listening at port 3000
```

- Open <http://localhost:3000>



Push to storage queues

- Open <http://localhost:3000/admin>



Get User Tickets

Get Latest Tickets