
Capstone-Project 1

By: Ankit Ranjan

A handwritten signature in black ink that reads "Ankit Ranjan". The signature is fluid and cursive, with "Ankit" on top and "Ranjan" below it, both starting with a capital letter.

Ankit Ranjan-DevOps Engineer

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.

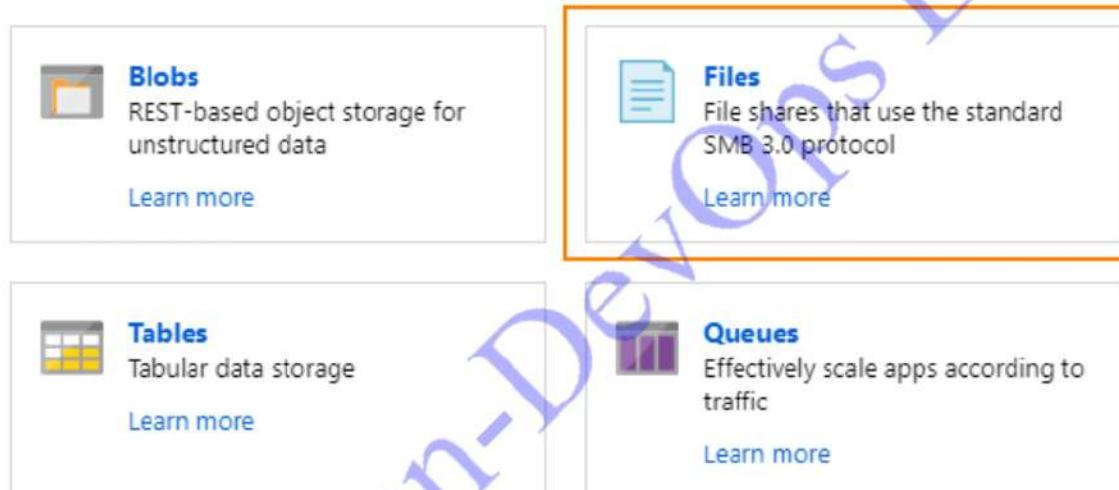
Issue 1: They wish to have a centralized store to store all their developer tools in. This store should be such that developers can mount it on their file system.

Solution:

Step 1: Create and open your storage account

Step 2: Click on Files

Services

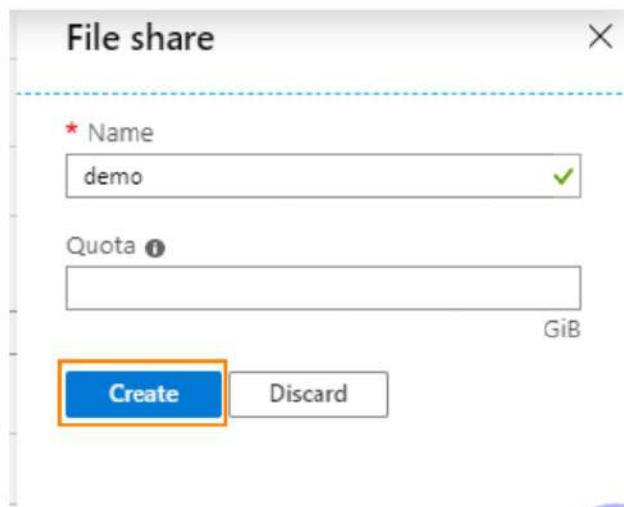


Tools and SDKs

Step 3: Click on +FileShare

The screenshot shows the 'File shares' blade for a storage account named 'vmwestrgdiag'. At the top, there are buttons for '+ File share' and 'Refresh'. Below that, a search bar says 'Search file shares by prefix'. A message at the bottom says 'You don't have any file shares yet. Click '+ File share' to get started.' There is also a note about the storage account name.

Step 4: Enter the details and click on Create



Step 5: Mount the File Share on your PC

Step 5.1: Open the File Share in Azure Portal

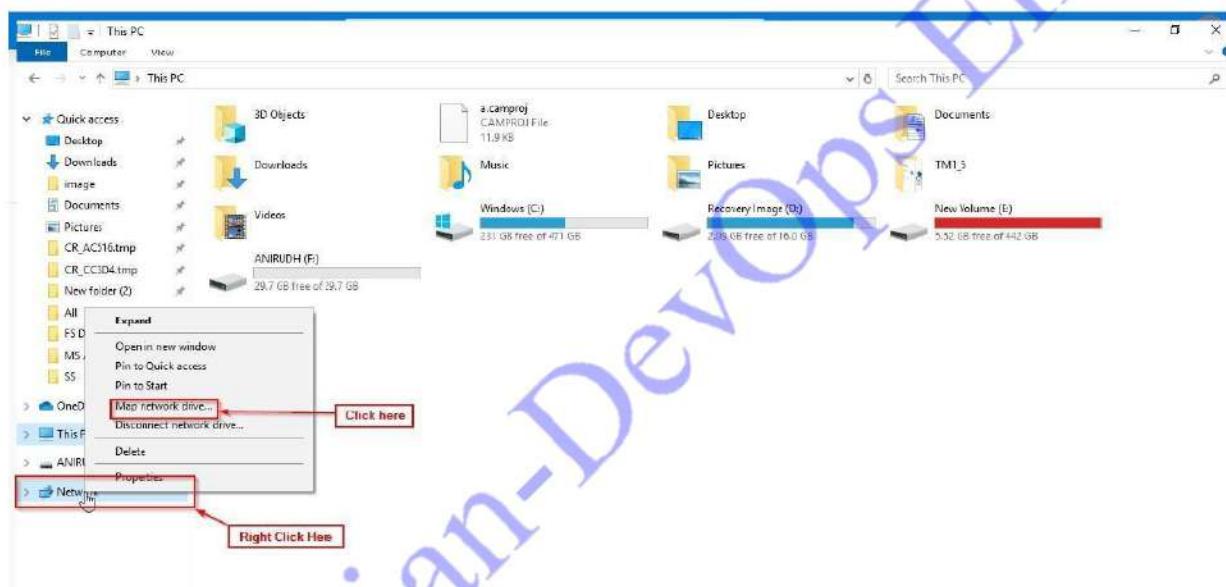
Step 5.2: Click on Connect

A screenshot of the Azure portal showing the 'demo' file share. The URL is 'Home > vmwestrigdiag - Files > demo'. The left sidebar shows 'File share' under 'demo'. The main area shows a 'Connect' button highlighted with an orange box, along with other actions like 'Upload', 'Add directory', 'Refresh', 'Delete share', 'Quota', 'View snapshots', and 'Create Snapshot'. A message says 'Backup (Preview) is not enabled for this file share. Click here to enable backup.' Below that is a 'Location: demo' section and a table with a single row 'No files found.'

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

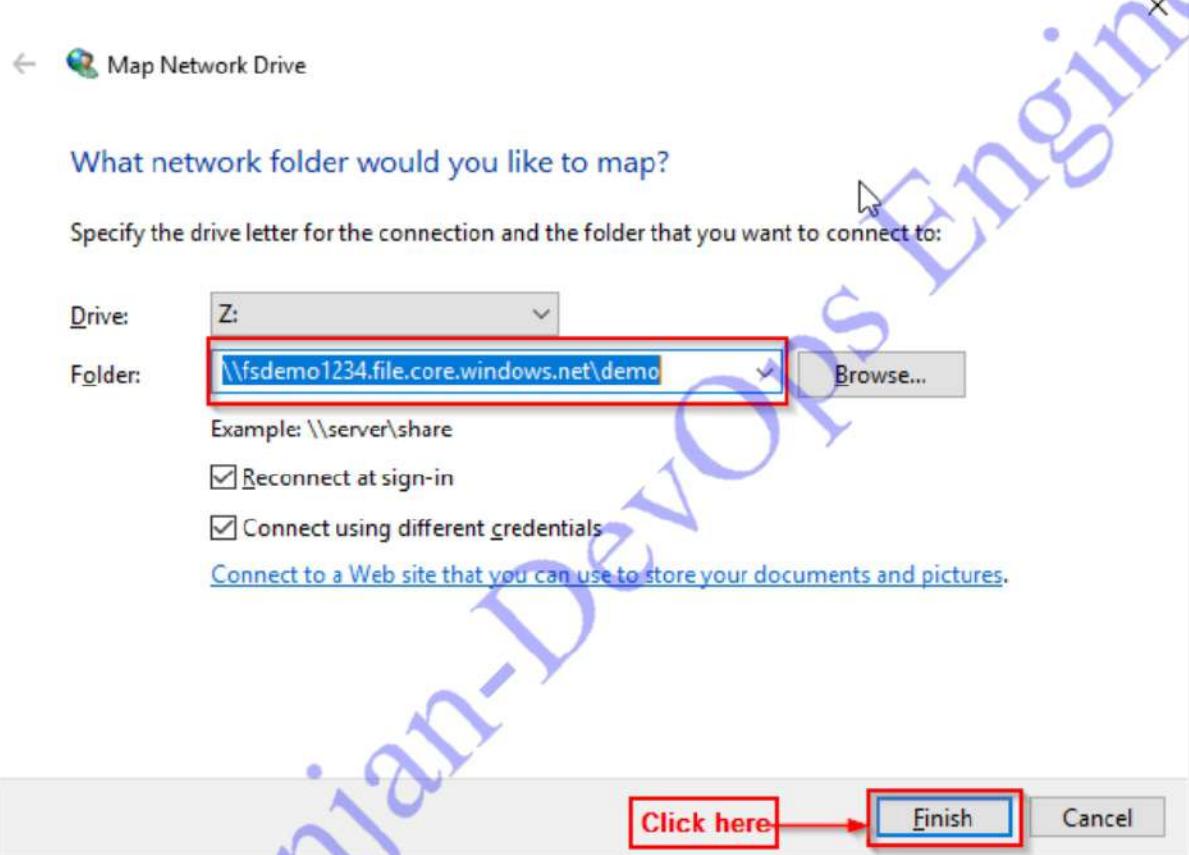
```
Test-NetConnection -ComputerName fsdemo1234.file.core.windows.net -Port 445  
# Save the password so the drive will persist on reboot  
Invoke-Expression -Command "cmdkey /add:fsdemo1234.file.core.windows.net /user:Azure\fsdemo1234  
/pass:sb+fZUElz5o6CDRC/ViFYptpKshhKnq4yJoMCPw8mHJIZ2wPCDwofeKdhCGsjQ/g7b7KWLrnLtdCC1+kfX6hpw=="  
# Mount the drive  
New-PSSDrive -Name Z -PSProvider FileSystem -Root '\\\fsdemo1234.file.core.windows.net\demo'  
Username  
Password  
Folder URL
```

Step 5.4: Open Windows Explorer

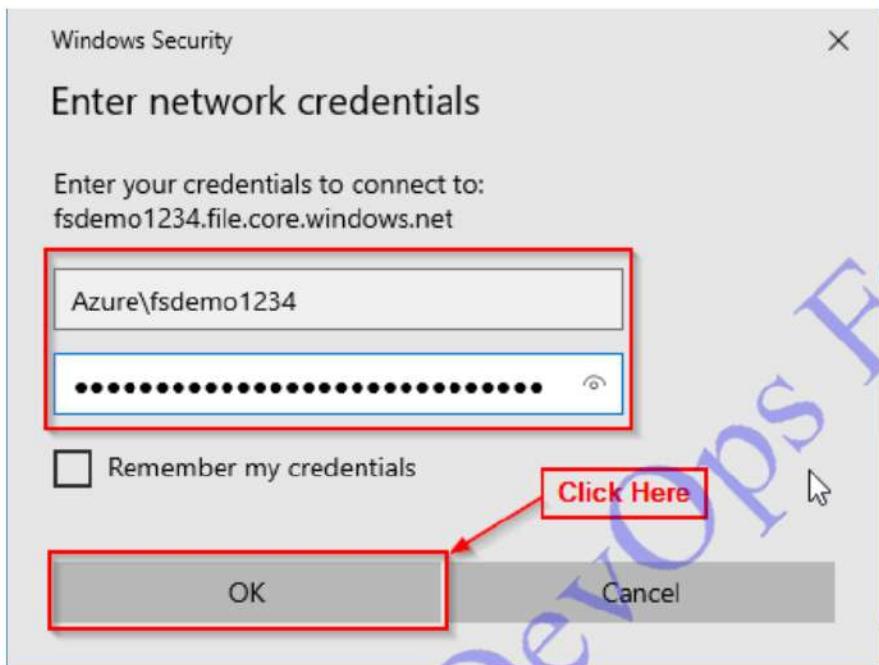


Step 5.5: Right Click on Network and Select Map network drive

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



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

The screenshot shows the Azure Storage account interface for the 'queuedemo123' account. The left sidebar is titled 'queuedemo123 - Blobs' and contains several service links: Properties, Locks, Export template, Blob service (selected), Blobs (highlighted with an orange border), Custom domain, Soft delete, Azure CDN, Add Azure Search, and Lifecycle Management (also highlighted with an orange border). The main pane displays a table of containers. The table has columns: NAME, LAST MODIFIED, PUBLIC ACCESS L., and LEASE STATE. One row is listed: sample, 9/26/2019, 4:22:26 PM, Private, Available. At the top of the main pane, there are buttons for '+ Container', 'Change access level', 'Refresh', and 'Delete'.

Step 4: Select Add rule

The screenshot shows the 'Lifecycle Management' blade for the 'queuedemo123' storage account. At the top, there are buttons for '+ Add rule', 'Enable' (with a checked checkbox), 'Disable', 'Refresh', and 'Delete'. Below this is a navigation bar with 'List view' (selected) and 'Code view'. A descriptive text block states: 'Azure blob storage lifecycle management offers a rich, rule-based policy for GPlv2 and blob storage accounts. Use the policy to transition your data to the appropriate access tiers or expire at the end of the data's lifecycle. Learn more'. The main area is a table with columns: NAME and STATUS. The table shows 'No results'.

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

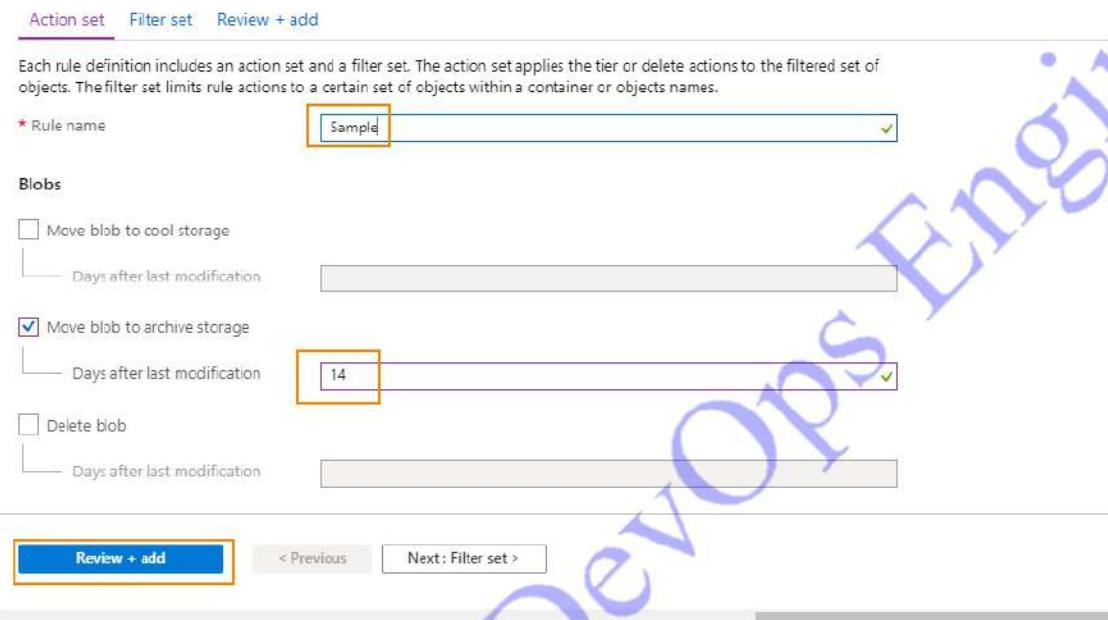
Blobs

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

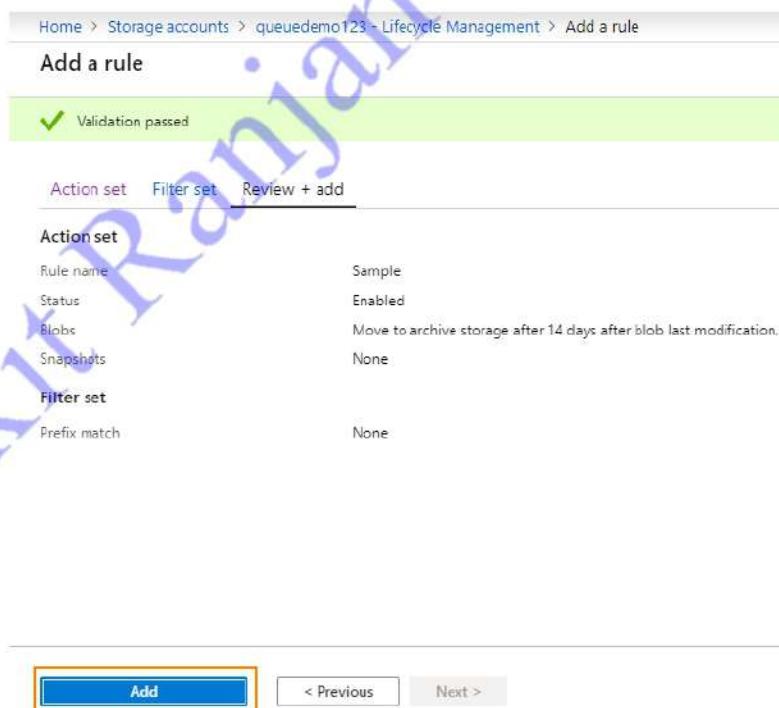
Action set

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 >

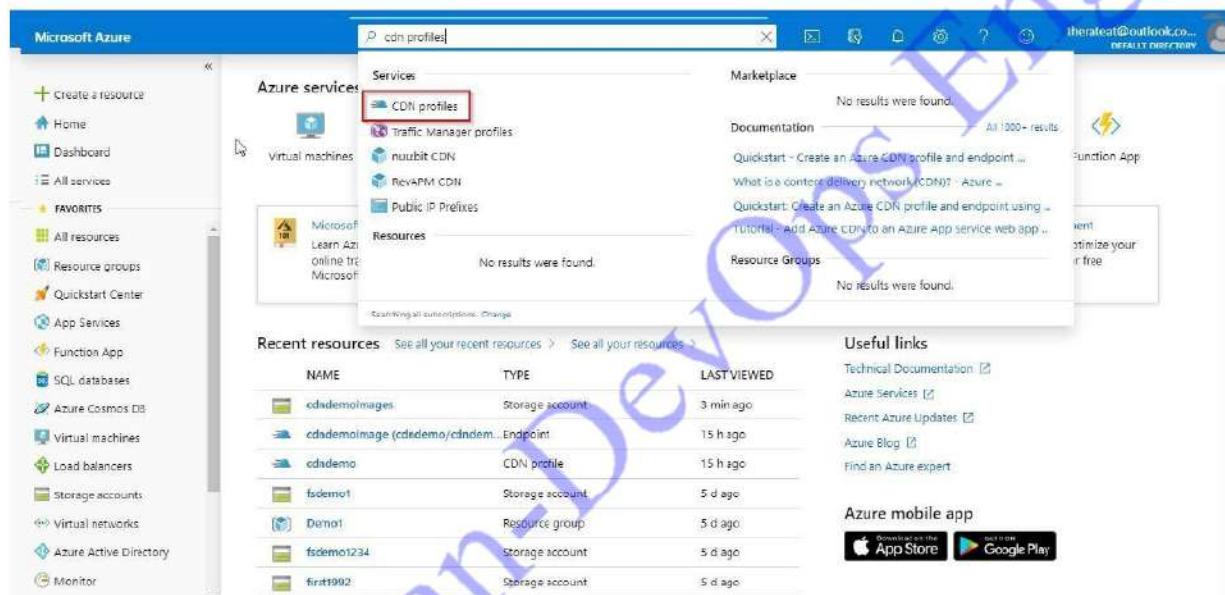


Issue 3: They wish to lower the latency of their website. They have noticed that users who are far away from their web server have complained that images take a lot of time to load.

Solution:

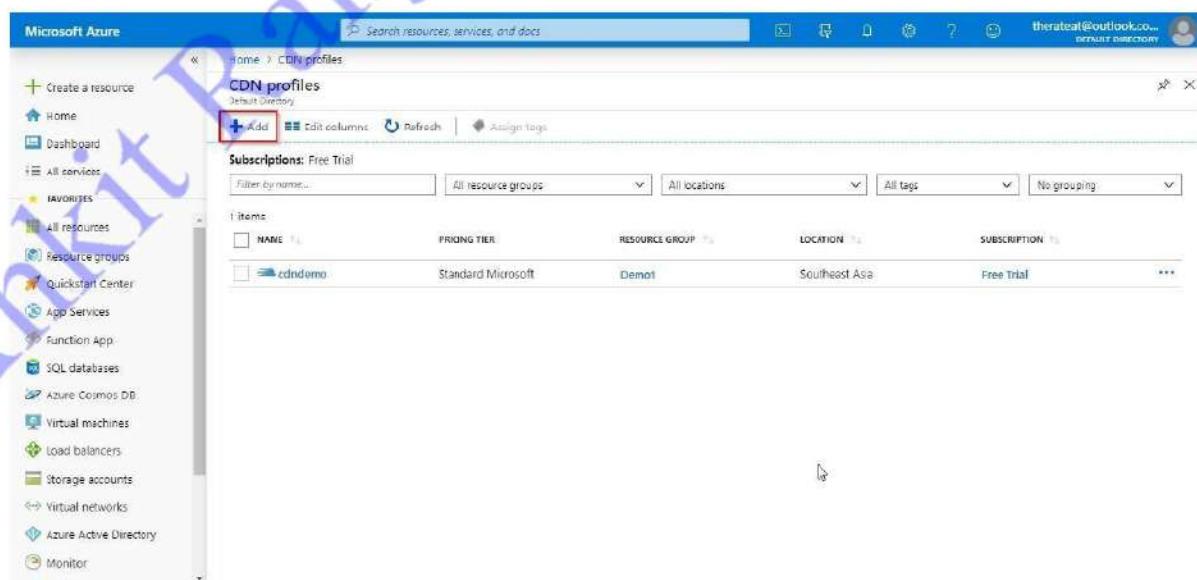
Step 1: Create a storage account for Blob Storage and upload a few images

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



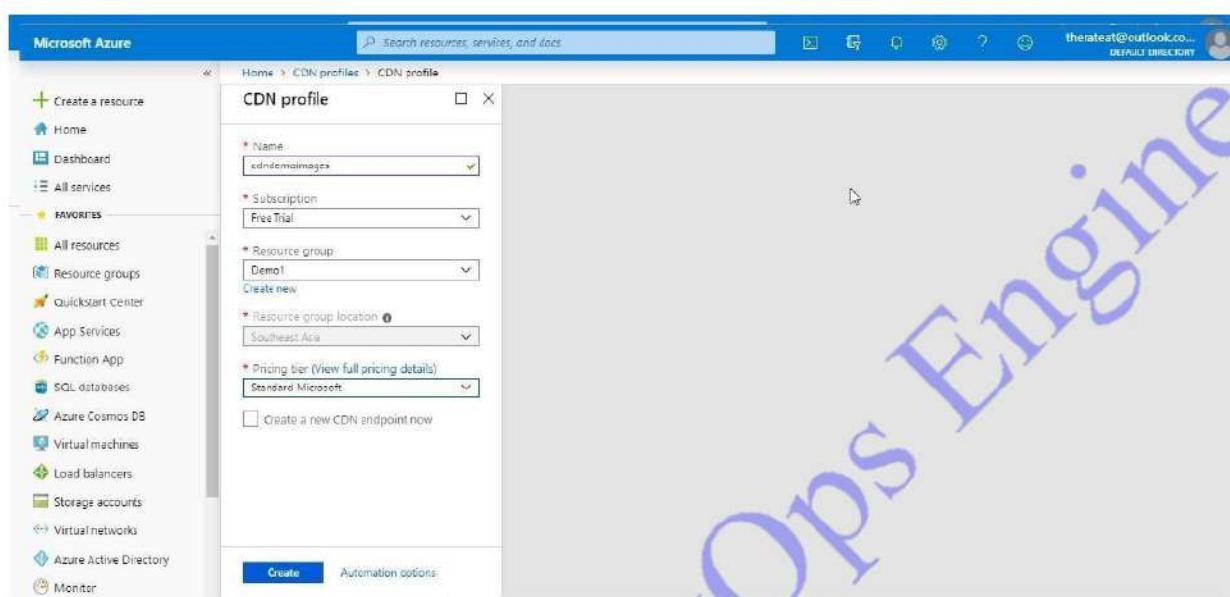
The screenshot shows the Microsoft Azure portal interface. The search bar at the top contains the text 'cdn profiles'. The left sidebar includes links for 'Create a resource', 'Home', 'Dashboard', 'All services', 'Favorites', 'All resources', 'Resource groups', 'Quickstart Center', 'App Services', 'Function App', 'SQL databases', 'Azure Cosmos DB', 'Virtual machines', 'Load balancers', 'Storage accounts', 'Virtual networks', 'Azure Active Directory', and 'Monitor'. The main content area displays search results under 'Services' and 'Resources'. Under 'Services', 'CDN profiles' is highlighted with a red box. Other options include 'Traffic Manager profiles', 'nuulbit CDN', 'REVAPM CDN', and 'Public IP Prefixes'. Under 'Resources', there are no results found. To the right, there is a 'Marketplace' section with documentation links for creating a CDN profile and endpoint, and a 'Function App' section with a 'Learn Azure Functions' link. A 'Recent resources' table lists several items, and a 'Useful links' section provides links to Technical Documentation, Azure Services, Recent Azure Updates, Azure Blog, and Find an Azure expert. At the bottom, there are download links for the Azure mobile app on the App Store and Google Play.

Step 3: Click on + Add



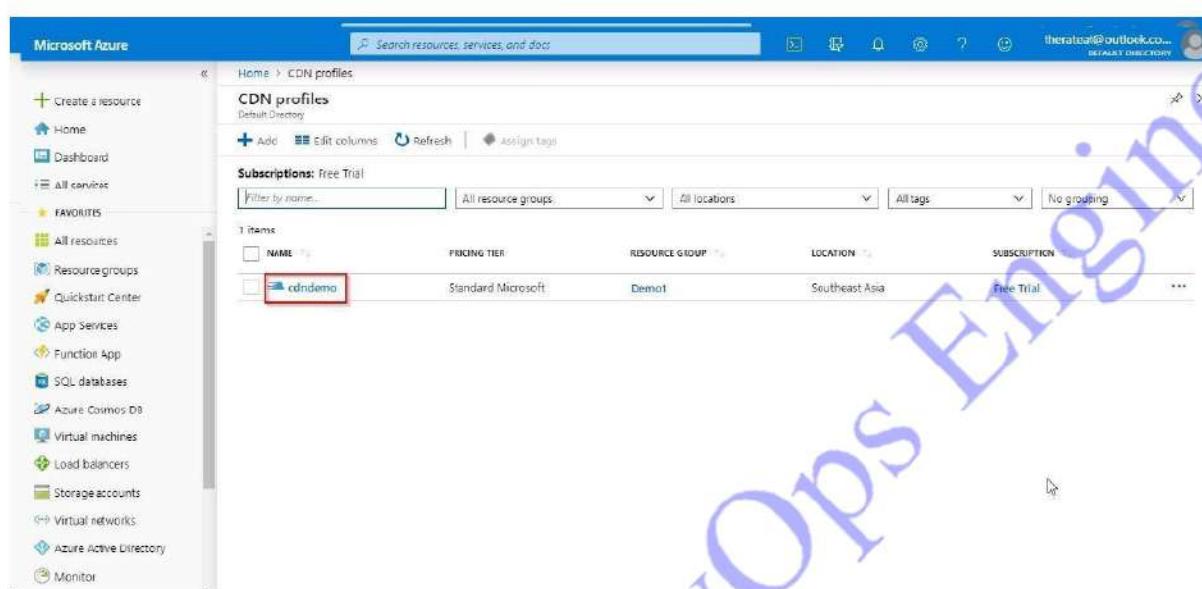
The screenshot shows the 'CDN profiles' list page in the Microsoft Azure portal. The left sidebar is identical to the previous screenshot. The main area shows a table of existing CDN profiles. The first row, 'cdndemo', is selected, indicated by a blue border. The table has columns for NAME, PRICING TIER, RESOURCE GROUP, LOCATION, and SUBSCRIPTION. The 'cdndemo' entry shows 'Standard Microsoft' as the tier, 'Demot' as the resource group, 'Southeast Asia' as the location, and 'Free Trial' as the subscription. At the top of the table, there is a '+ Add' button, which is highlighted with a red box. Other buttons in the header include 'edit columns', 'refresh', and 'Assign tags'.

Step 4: Fill the details and click on Create



Ankit Ranjan-DevOps Engineer

Step 5: Click on the CDN Profile just created



The screenshot shows the Microsoft Azure portal interface. On the left, there's a navigation sidebar with various service icons like Home, Dashboard, All services, Favorites, and others. The main content area is titled 'CDN profiles' under 'Default Directory'. It includes a search bar, filter options for Subscriptions, Resource groups, Locations, Tags, and Grouping, and buttons for Add, Edit columns, Refresh, and Assign tags. A table lists one item: 'cdnprofile'. The table has columns for NAME, PRICING TIER, RESOURCE GROUP, LOCATION, and SUBSCRIPTION. The 'cdnprofile' row is highlighted with a red box around its first column. The 'cdnprofile' entry has a standard Microsoft logo icon, belongs to the 'Demot1' resource group, is located in Southeast Asia, and is associated with the 'Free Trial' subscription.

NAME	PRICING TIER	RESOURCE GROUP	LOCATION	SUBSCRIPTION
cdnprofile	Standard Microsoft	Demot1	Southeast Asia	Free Trial

Step 6: Click on +Endpoint

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various service icons like Home, Dashboard, All services, and Favorites. The main area is titled 'cdndemo' under 'CDN profiles'. It has a search bar at the top. Below it, there are several tabs: Overview (which is selected), Activity log, Access control (IAM), Tags, Diagnose and solve problems, Properties, quickstart, Locks, Export template, Support + troubleshooting, Resource health, and New support request. In the center, there's a summary card for 'Resource group (change) Demo 1' with details: Status (Active), Location (Southeast Asia), Subscription (change) (free trial), Subscription ID (5b3488ad-7cf1-459f-9996-689235d9b206). To the right, there's a table titled 'Endpoints' with one row: cdndemoimage.azureedge.net, Running, HTTP HTTPS, Storage. At the bottom right, it says 'Pricing Tier Standard Microsoft'.

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 shows the Microsoft Azure portal interface. On the left, the navigation menu includes 'Create a resource', 'Home', 'Dashboard', 'All services', 'FAVORITES' (with items like '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'), and a 'Search resources, services, and docs' bar at the top.

In the center, the 'cdndemo' CDN profile is selected. The 'Overview' tab is active, showing details such as 'Resource group (change) Demo1', 'Status Active', 'Location Southeast Asia', 'Subscription (change) Free Trial', and 'Subscription ID'. Below this is the 'Endpoints' section, which lists one endpoint: 'cdndemoimage.azureedge.net' (STATUS: Running, PROTOCOL: HTTP HTTPS).

A modal dialog titled 'Add an endpoint' is open on the right. It contains fields for 'Name' (set to 'cdndemomg'), 'Origin type' (set to 'Storage'), 'Origin hostname' (set to 'cdndemomimages.blob.core.windows.net'), 'Origin path' (set to '/Path'), 'Origin host header' (set to 'cdndemomimages.blob.core.windows.net'), 'Protocol' (checkboxes for 'HTTP' checked and 'HTTPS' checked), 'Origin port' (set to 80 for HTTP and 443 for HTTPS), and 'Optimized for' (set to 'General web delivery'). A red box highlights the 'Add' button at the bottom-left of the dialog.

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

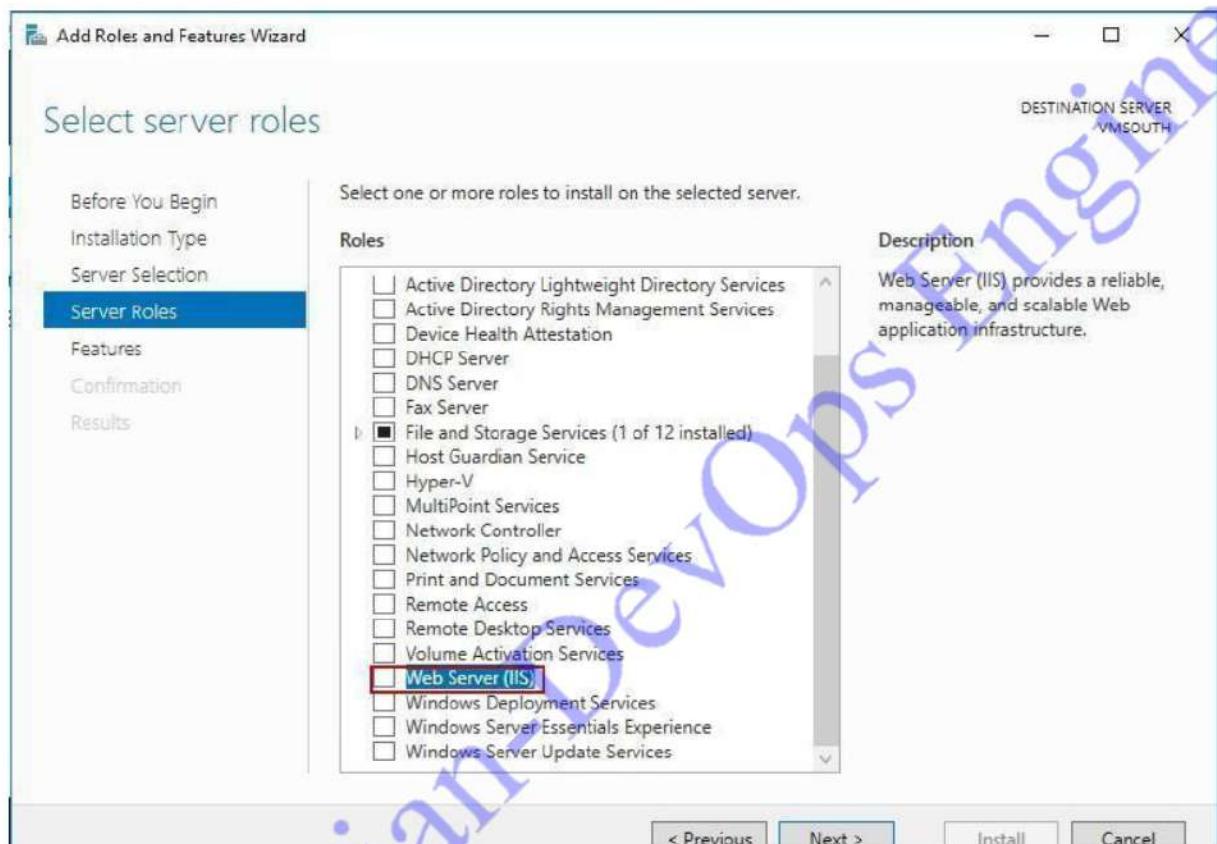
Step 3: Open Server Manager

Step 4: Click on 'Add Roles and Features'

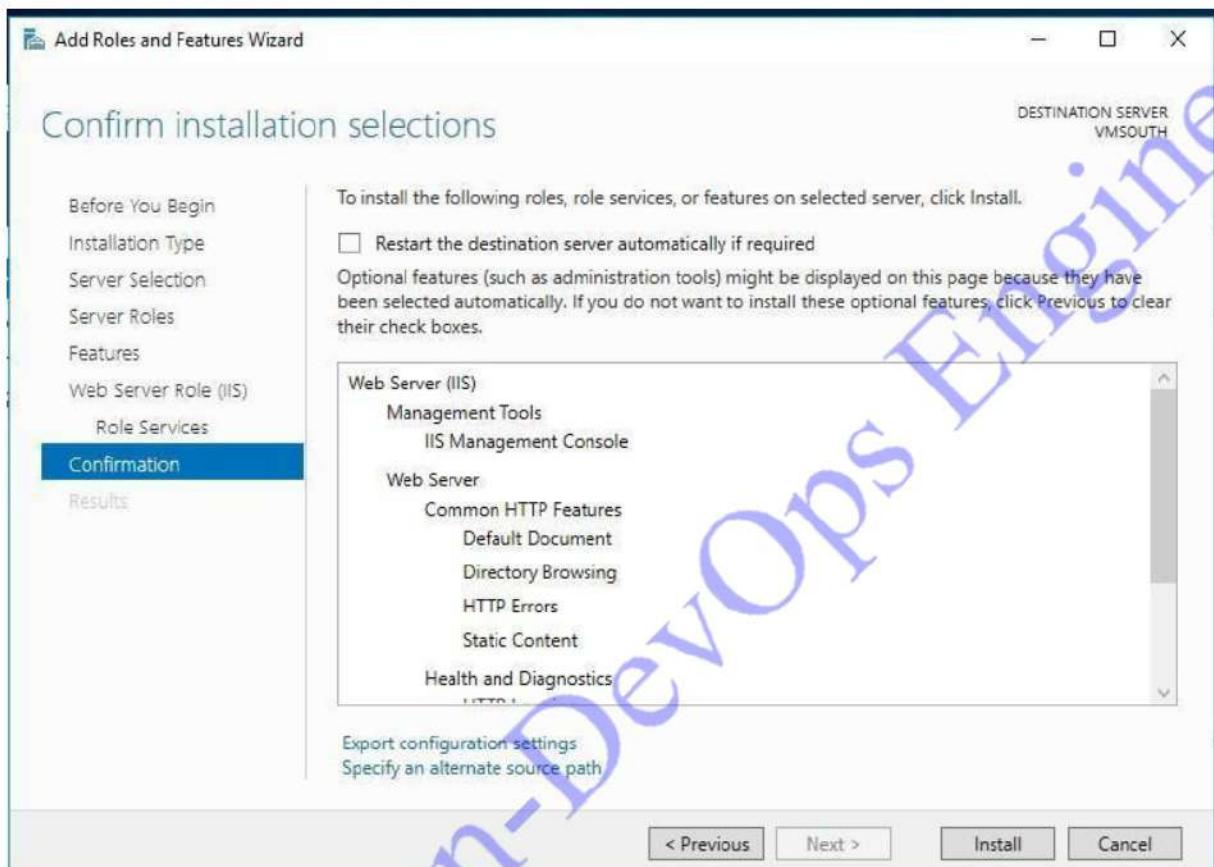


- 1 Configure this local server
 - 2 Add roles and features
 - 3 Add other servers to manage
 - 4 Create a server group
 - 5 Connect this server to cloud services

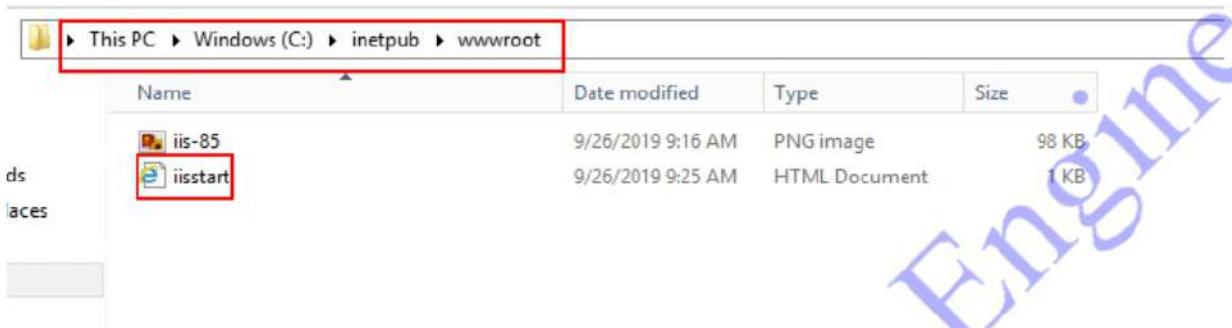
Step 5: In the wizard click on 'Next' 3 times and then in 'roles' select Web Server (IIS)



Step 6: Click on Next and Install



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

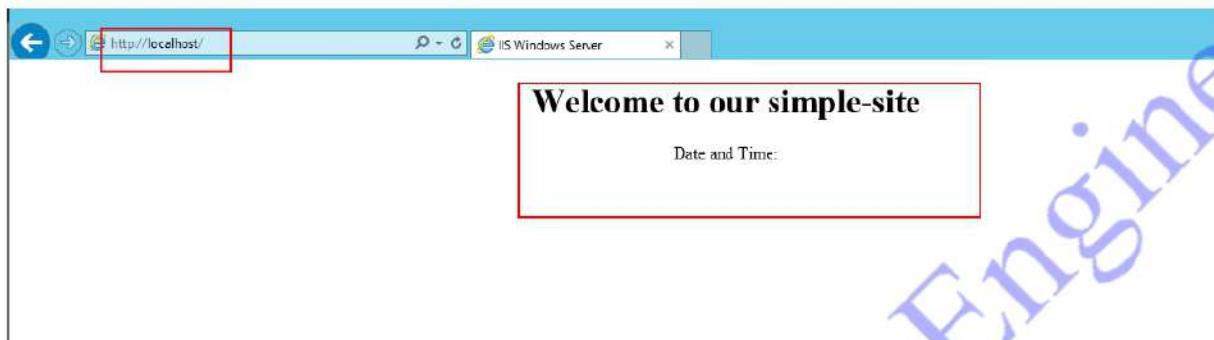


Step 8: Open iisstart.html in Notepad

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

VMWest

Resource group (change): VMWestRG

Status: Running

Location: West India

Subscription (change): Azure for Students

Subscription ID: 8c0b4d4b-1eb5-4dc2-9e82-9b9cd1d5bc47

Tags (change): Click here to add tags

Computer name: VMWest

Operating system: Windows (Windows Server 2012 R2 Datacenter)

Size: Standard DS1 v2 (1 vcpus, 3.5 GiB memory)

Ephemeral OS disk: N/A

Public IP address: 104.211.166.68

Private IP address: 10.0.1.4

Virtual network/subnet: VMWestRG-vnet/default

DNS name: Configure

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

AnkitRanjan-DevOps Engineer

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

The screenshot shows the 'Virtual networks' page in the Azure portal. It lists three VNets: 'demo1-vnet' (Resource Group: demo1, Location: West US, Subscription: Free Trial), 'VN1' (Resource Group: VN1DEMO, Location: South India, Subscription: Free Trial), and 'VN2' (Resource Group: VN1DEMO, Location: South India, Subscription: Free Trial). A red box highlights the 'Click Here' button next to the VN1 entry.

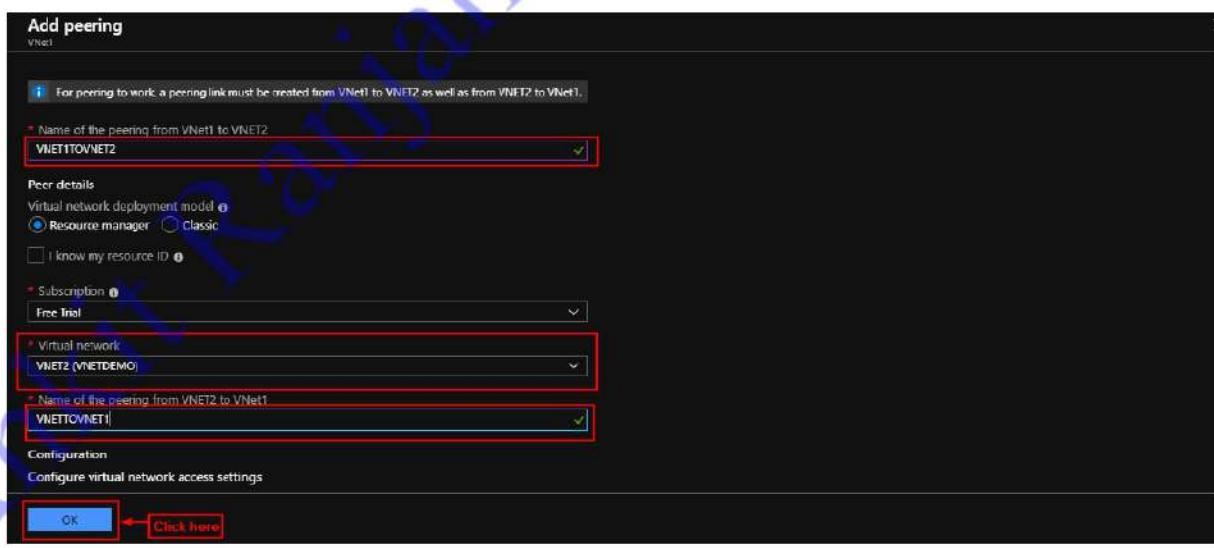
Step 5: In the sidebar click on Peerings

The screenshot shows the 'VNet1' settings page in the Azure portal. The left sidebar has a 'Peerings' option with a red box and 'Click Here' button. The main pane displays the VNet1 configuration, including its resource group (VN1DEMO), location (South India), subscription (Free Trial), and address space (172.168.4.0/24). It also shows connected devices: 'vm199' and 'vm2112' (both Network interface, IP addresses 172.168.4.4 and 172.168.4.5 respectively).

Step 6: Click on + Add



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



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

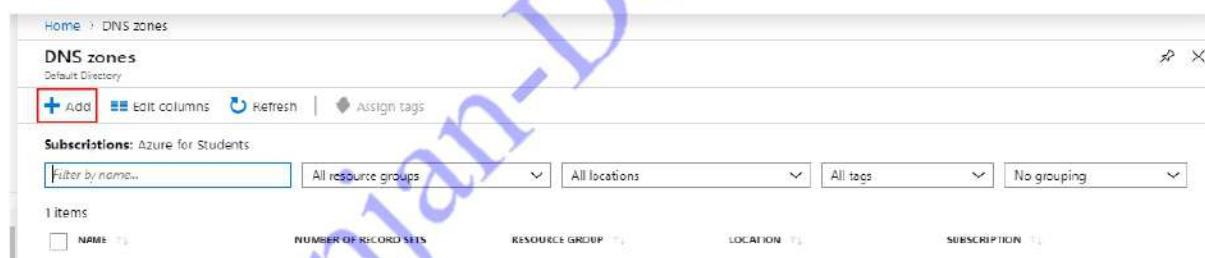
Step 2: Buy a free domain (for our purposes it can be anything)



-- 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: your 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'

The screenshot shows the Azure portal interface for managing a DNS zone. The URL in the address bar is "Home > DNS zones > simple-site.ml". On the left, there's a sidebar with options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Properties, and Locks. The main content area has a search bar at the top. A red box highlights the "+ Record set" button. Below it, there are sections for Resource group (change) to "vmwestrg", Subscription (change) to "Azure for Students", and Subscription ID "8c0b4d4b-16b5-4dc2-9e82-9b9cd8d5bc47". To the right, there's a list of Name servers: ns1-04.azure-dns.com, ns2-04.azure-dns.net, ns3-04.azure-dns.org, and ns4-04.azure-dns.info. There's also a section for Tags (change) with a note to click here to add tags. A tooltip at the bottom says: "You can search for record sets that have been loaded on this page. If you don't see what you're looking for, you can record sets to load."

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

The screenshot shows a web browser window with the URL "simple-site.ml" in the address bar. The page content includes a "Welcome to our simple-site" header, a "Date and Time:" label, and the timestamp "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

The screenshot shows the Azure portal interface for managing load balancers. At the top, there's a breadcrumb navigation: Home > Load balancers. Below the navigation, there's a header with 'Load balancers' and a 'Default Directory' dropdown. A red box highlights the 'Click Here' link next to the 'Default Directory' dropdown. Below the header, there are buttons for '+ Add', 'Edit columns', 'Refresh', and 'Assign tags'. A 'Subscriptions' section shows 'Free Trial'. There are also filters for 'Filter by name...', 'All resource groups', 'All locations', 'All tags', and 'No grouping'. A message '0 items' is displayed above a table header with columns: NAME, RESOURCE GROUP, LOCATION, and SUBSCRIPTION. The table body below the header is empty and displays the message 'No load balancers to display'. At the bottom of the page, there's a note about Azure Load Balancer supports TCP/UDP-based protocols and a 'Create load balancer' button.

Step 4: Add details and click Review + create

Home > Load balancers > Create load balancer

Create load balancer

* Subscription: Free Trial

* Resource group: LoadBalancerDEMO

Instance details

* Name: load-balancer

* Region: (Asia Pacific) South India

* Type: Public

* SKU: Basic

Public IP address

* Public IP address: Create new

* Public IP address name: load-balancer-ip

Public IP address SKU: BASIC

* Assignment: Dynamic

Review + create

< Previous | Next > | Click Here | download a template for automation

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 > | Click Here | download a template for automation

Step 6: Open the load balancer

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

The screenshot shows the Azure portal interface for managing a load balancer. The left sidebar has a 'Load-balancer - Backend pools' section selected. The main content area displays a table with columns for VIRTUAL MACHINE, VIRTUAL MACHINE ST..., NETWORK INTERFACE, and PRIVATE IP ADDRESS. A search bar at the top says 'Search backend address pools'. At the top right, there are 'Add' and 'Refresh' buttons, with the 'Add' button being the one highlighted by a red box and an arrow pointing to it from the text 'Click Here'.

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



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

The screenshot shows the AWS Management Console interface for a Load Balancer. The top navigation bar includes 'Home', 'load-balancer - Health probes', and a search bar. On the left, a sidebar menu lists various options: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings (with sub-options like Frontend IP configuration, Backend pools, and Health probes), Load balancing rules, Inbound NAT rules, Properties, Locks, Export template, and Monitoring. The 'Health probes' option is highlighted with a red box and a red arrow points to the '+ Add' button at the top of the main content area. The main content area displays a table header for 'NAME', 'PROTOCOL', 'PORT', and 'USED BY'. Below the header, a message says 'No results.'

Step 10: Enter details click on OK

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

Add health probe

load-balancer

* Name
health-probe

IP version
IPv4

Protocol
TCP

* Port
80

* Interval
5 seconds

* Unhealthy threshold
2 consecutive failures

OK **Click Here**

AnkitRanjan-DevOps Engineer

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

The screenshot shows the AWS CloudFront Load Balancer configuration interface. On the left, there is a navigation sidebar with the following items:

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Settings
 - Frontend IP configuration
 - Backend pools
 - Health probes
 - Load balancing rules** (this item is highlighted with a red box)
 - Inbound NAT rules
 - Properties
 - Locks
 - Export template
- Monitoring

The main content area is titled "load-balancer - Load balancing rules". It features a search bar at the top labeled "Search (Ctrl+J)" and a button labeled "+ Add". A red arrow points from the text "Click Here" to the "+ Add" button. Below this, there is a table header with columns: NAME, LOAD BALANCING RULE, BACKEND POOL, and HEALTH PROBE. The table body contains the message "No results."

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)	Backend pool
LoadBalancerDEMO	backend-pool (2 virtual machines)
Location	Health probe
South India	health-probe (Tcp:80)
Subscription (change)	Load balancing rule
Free Trial	load-balancing-rule (Tcp/80)
Subscription ID	NAT rules
b9be3eac-a886-420f-88c4-c8a6156af111	0 inbound
SKU	Public IP address
Basic	104.211.204.139 LoadBalancerFrontEnd
Tags (change)	Open this IP
Click here to add tags	

✖

AnkitRanjan-DevOps Engineer

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

The screenshot shows the Azure Portal search interface. The search bar at the top contains the text "Active". Below the search bar, there are sections for "Services", "Marketplace", and "Documentation".

- Services:** Includes "Activity log", "Azure Active Directory" (which is highlighted with a red box), "HDInsight clusters", and "Monitor".
- Marketplace:** Includes "Active Cypher - Scouting Reports", "Active Directory Health Check", "Azure Active Directory", and "Kinetica Active Analytics Platform".
- Documentation:** Includes links to "Active geo-replication - Azure SQL Database | Microsoft Docs", "Overview of Azure Active Directory Domain Services ...", "What is Azure Active Directory? - Azure Active Directory ...", and "Tutorial - Create an Azure Active Directory Domain ...".

At the bottom left, it says "Searching all subscriptions. Change".

Step 2: Click on it and open it

Step 3: In the sidebar, click on Users

The screenshot shows the Azure Active Directory - Default Directory - Overview page. The left sidebar has a 'Manage' section with several options: 'Users' (selected and highlighted with a red box), 'Groups', 'Organizational relationships', 'Roles and administrators', 'Enterprise applications', 'Devices', 'App registrations', 'Identity Governance', 'Application proxy', 'Licenses', 'Azure AD Connect', and 'Custom domain names'. The main content area displays 'Default Directory' information, including the email address 'pandeyshlokyahoo.com.onmicrosoft.com' and the status 'Azure AD Free'. It also shows a 'Sign-ins' section with a message: 'To see sign-in data, your organization needs Azure AD Premium P1 or P2. Start a free trial.' Below this, there's a 'What's new in Azure AD' section with a link to 'View archive'.

Step 4: Click on Bulk Create

The screenshot shows the 'Users - All users' page in Azure Active Directory. The top navigation bar includes buttons for 'New user', 'New guest user', 'Bulk create' (highlighted with a red box), 'Bulk invite', 'Bulk delete', 'Download users', and 'More'. The left sidebar shows 'All users' and 'Deleted users'. The main area contains search and filter fields for 'Search' (Name or email), 'Search attributes' (Name, email (begins with)), and 'Show' (All users).

Step 5: Click on Download CSV Template

Bulk create user (Preview) X

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 pas First name Last name Job title [] Department Block sign Usage loc Street address State or province Country or Office [ph] City [city] ZIP or post Office phc Mobile phone [mobile])	John Doe	John@parabcd@123	John Doe	Sample User	Sample No	India											
Jane Doe	Jane@parabcd@123	Jane Doe	Sample User	Sample No	India												

Step 7: Upload the CSV File

Bulk create user (Preview) X

Default Directory - Azure Active Directory

1. Download csv template (optional)

[Download](#)

2. Edit your csv file

3. Upload your csv file

"UserCreateTemplate.csv" Select

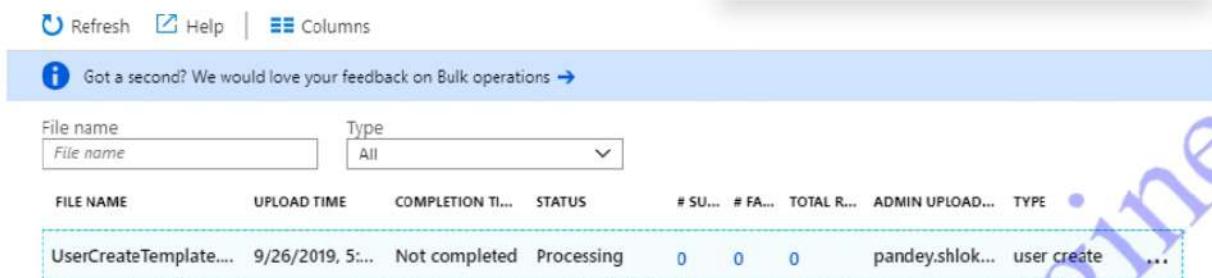
File uploaded successfully

[Learn more about bulk import users](#)

Submit

Step 8: Click on Submit

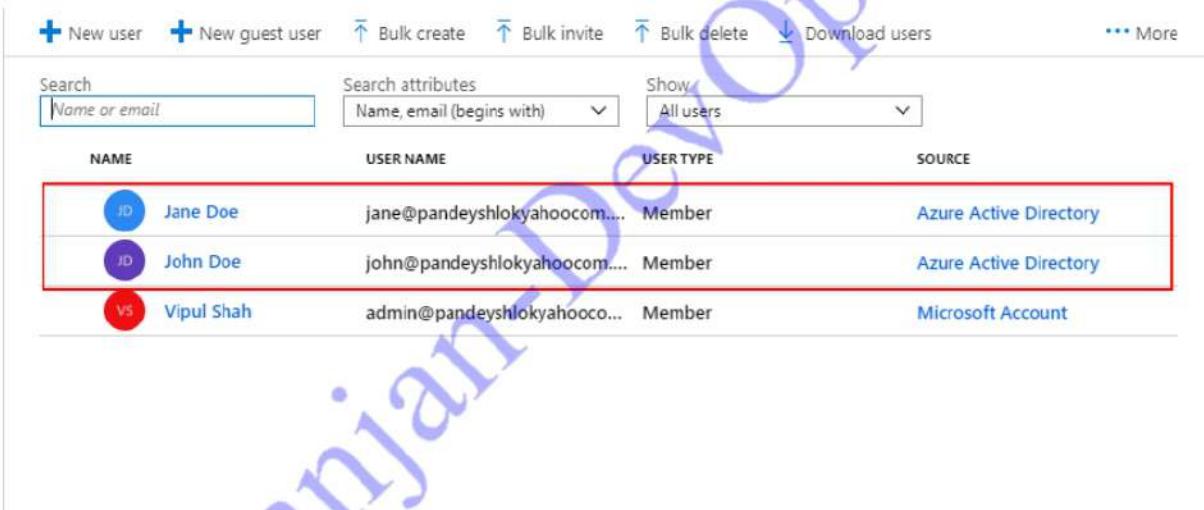
Step 9: In the sidebar click on Bulk Operation Details to view the status



The screenshot shows a web-based application interface for managing bulk operations. At the top, there are navigation links for Refresh, Help, and Columns. A blue header bar contains a feedback message: "Got a second? We would love your feedback on Bulk operations →". Below this, there are two dropdown menus: "File name" set to "File name" and "Type" set to "All". The main content area displays a table with the following data:

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



The screenshot shows the User management page. At the top, there are several buttons: New user, New guest user, Bulk create, Bulk invite, Bulk delete, Download users, and More. Below these are search and filter fields: Search (Name or email), Search attributes (Name, email (begins with)), Show (All users). The main table lists users with the following data:

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

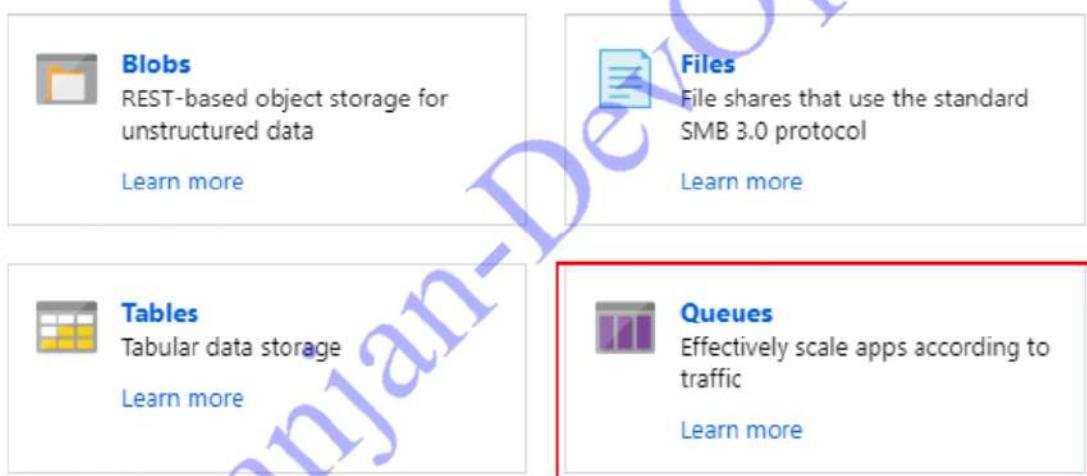
Issue 9: Finally they have two applications that need to pass messages between one another on an on-demand basis i.e. an application will send the message and another application will receive and process it when it can. You need to set up a service in such a way that these applications can do so (you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

Solution:

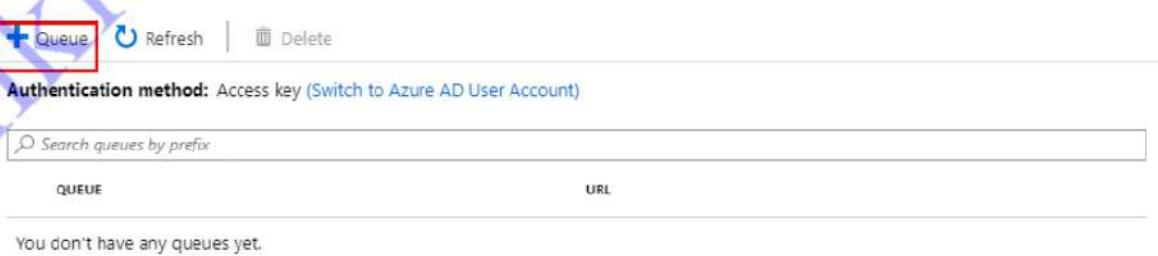
- Make sure you have node js installed and run npm install in the project directory –

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+I)

Overview

Activity log

Access control (IAM)

Tags

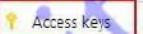
Diagnose and solve problems

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

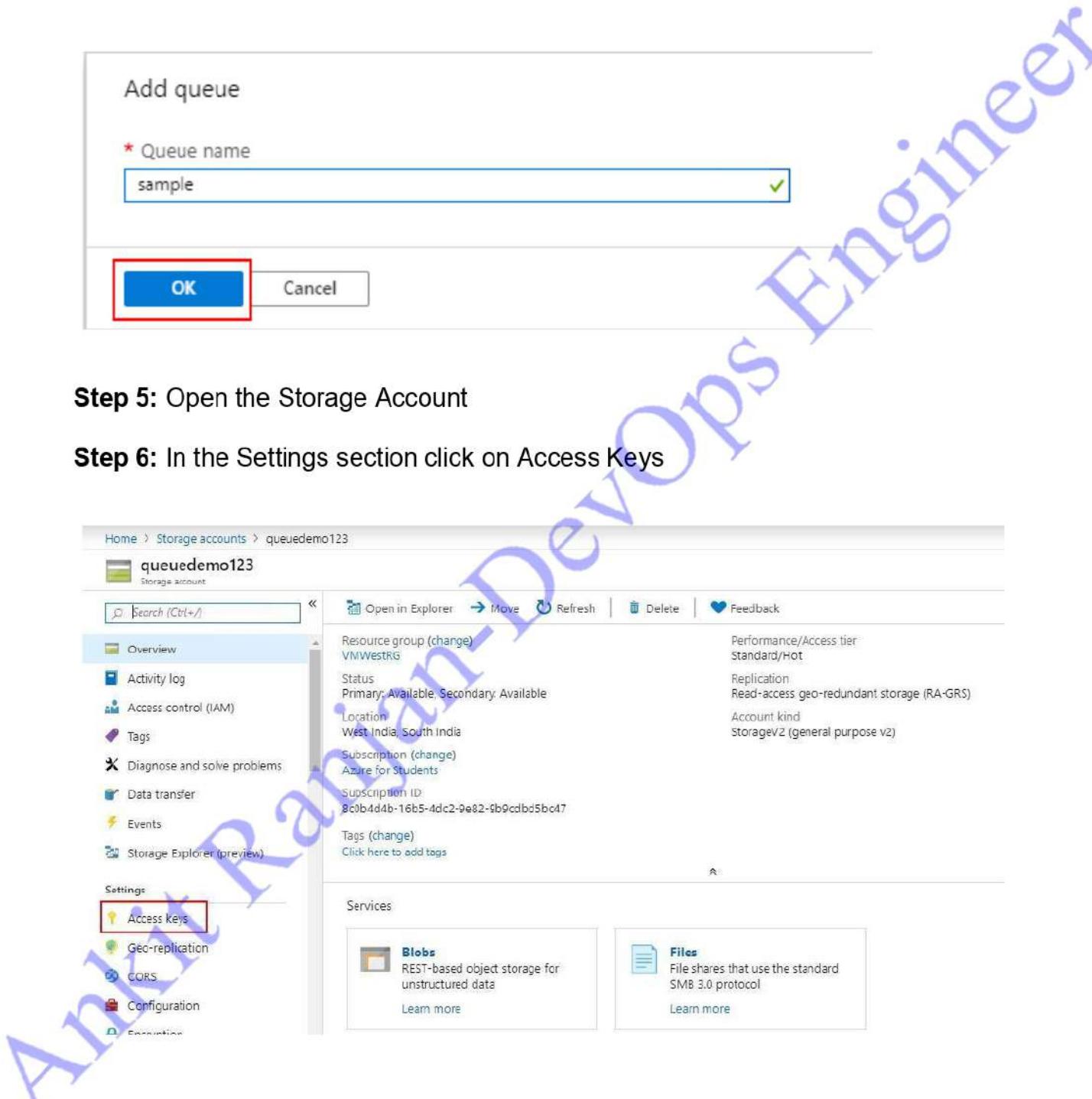
Subscription ID 8c0b4d4b-16b5-4dc2-9e82-9b9cd8d5bc47

Tags (change) Click here to add tags

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

The screenshot shows the Azure Storage portal interface. It displays two sets of key information for a storage account:

- key1**:
 - Key: Nyb1zpwQbMNyb/eGrv4qwkjVxExPu0CNM1x3nnZC0vLCMgD7q0CiLPDF9YscEBHLGm3D+o17WOBY7VCIF4KlfQ==
 - Connection string: DefaultEndpointsProtocol=https;AccountName=queuedemo123;AccountKey=Nyb1zpwQbMNyb/eGrv4qwkjVxExPu0CNM1x3nnZC0vLCMgD7...
- key2**:
 - Key: fqg1Gn8YIdF8TYLiT5KyF74YD4AxNovDhvIqhSwjP07bCf6fj/ObSSbQ5PpB+q3HVRXjnYpX2aOc4bpVIENWdg==
 - Connection string: DefaultEndpointsProtocol=https;AccountName=queuedemo123;AccountKey=fqg1Gn8YIdF8TYLiT5KyF74YD4AxNovDhvIqhSwjP07bCf6fj/ObSS...

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

Step 9: Replace information:

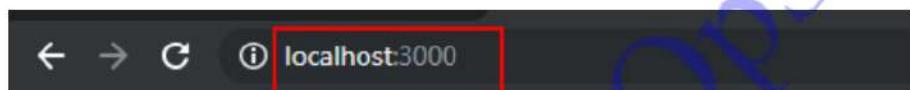
- queueName: Name of your Queue you just created.
- azureStorageAccount: Name of your Storage Account.
- azureStorageAccessKey: Access Key for your storage account.

```
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)

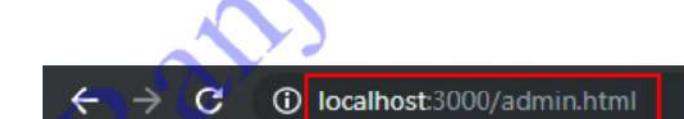
```
Intellipaat-Team@DESKTOP-FPNPKTE MINGW64 ~/Desktop/Storage Queue Demo
$ node server.js
Listening at port 3000
```

- Open <http://localhost:3000>



Push to storage queues

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



Get User Tickets

Get Latest Tickets

Thank you
Ankit Ranjan