DEVOPS with MULTI-CLOUD
Practice Tasks

Institute Name	**:**	V Cube software solutions
Course	**:**	DevOps with Multi-Cloud
Batch	**:**	30
Trainer	**:**	Krishna reddy sir

Prepared by	**:**	G.Bhavish (MCD-AZ30-024)

TASK-15 : Storage Account.

Date : 06/02/26

Objective :-

To store, manage, and securely access different types of data in the cloud, including files, blobs, queues, and tables, with high availability and scalability.

Storage Account :-

→ Azure Storage Account is a cloud service used to store different types of data securely.
→ It can store files, images, videos, backups, and application data.
→ It supports different storage types like Blob storage, File storage, Queue storage, and Table storage.
→ It provides high availability and data durability using replication options.
→ Data can be accessed securely using access keys, shared access signatures (SAS), or Azure AD authentication.
→ It allows scaling automatically based on storage needs.
→ It supports encryption to protect stored data.
→ the account name should be unique across azure accounts and all characters should be lowercase,the range for the account name is 3 - 24 characters.

→ There are four types of data storages :-

➤ Blob storage :- Used to store large unstructured data like images, videos, backups, and documents.

- block blob - unstructured data,images,videos,etc.
- page blob - stores vm files like vhd.files
- append blob - stores all log files.

➤ File shares :- The file share is like a network shared folder, acts like a traditional file server. It provides fully managed file storage in the cloud.

➤ Tables :- Used to store structured NoSQL data in key-value format for fast access and scalable storage.

➤ Queues :- Used to store messages between applications, mainly for communication between different services.

Azure storage redundancy :-

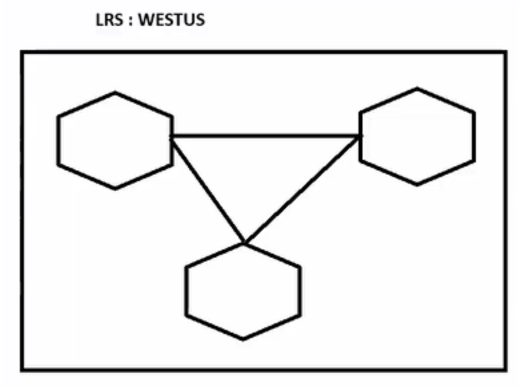
→ The azure storage always stores multiple copies of your data to protect it from planned and unplanned events.

→ it protects against hardware failure,data center failure and regional failure.

- Local Redundant Storage (LRS)
- Zone Redundant Storage (ZRS)
- Geo-Zone Redundant Storage (GZRS)
- Geo Redundant Storage (GRS)

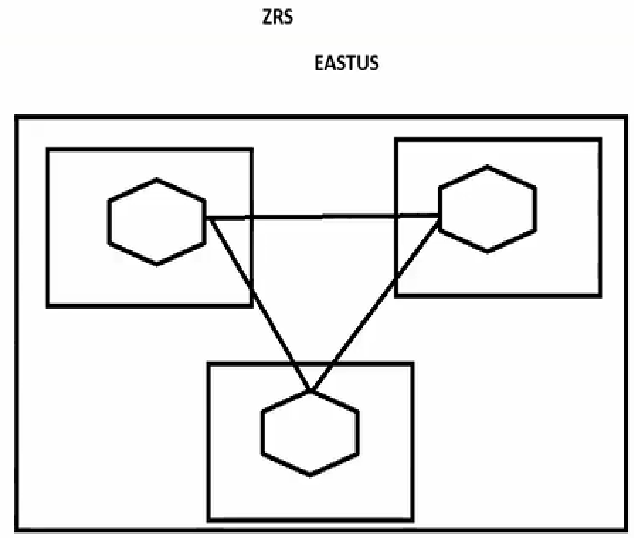
➤ Local Redundant Storage (LRS):-

- Stores three copies of data within a single datacenter.
- Protects against hardware failures within that location.
- Lowest cost replication option.



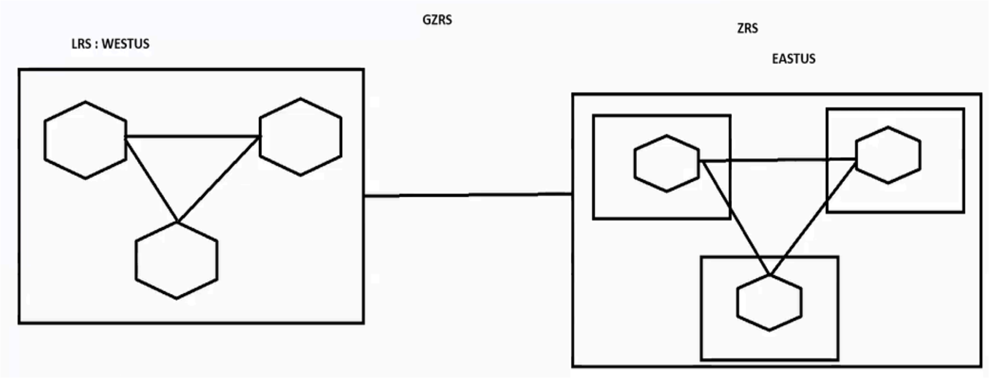
➤ Zone Redundant Storage (ZRS):-

- Stores data across multiple availability zones in the same region.
- Protects against datacenter-level failures.
- Provides higher availability than LRS.



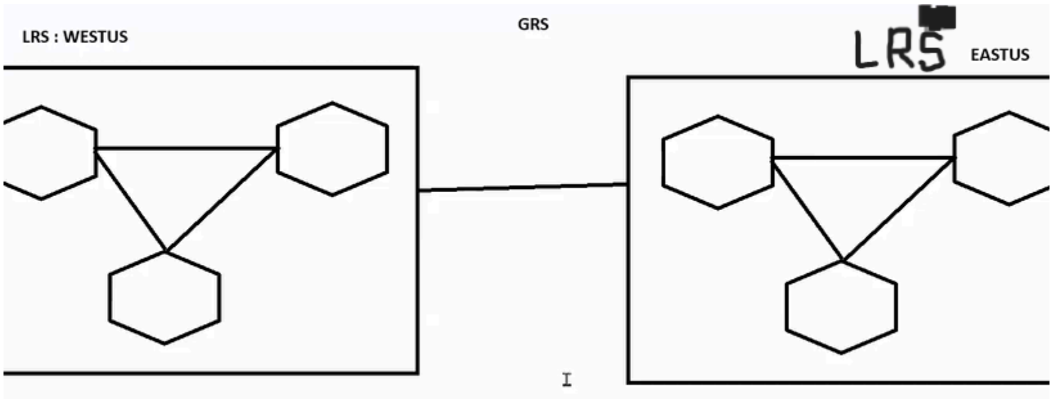
➢ Geo-Zone Redundant Storage (GZRS):-

- Stores data in the primary region and replicates to a secondary region
- Protects against regional outages.
- The secondary region is used only during failover.

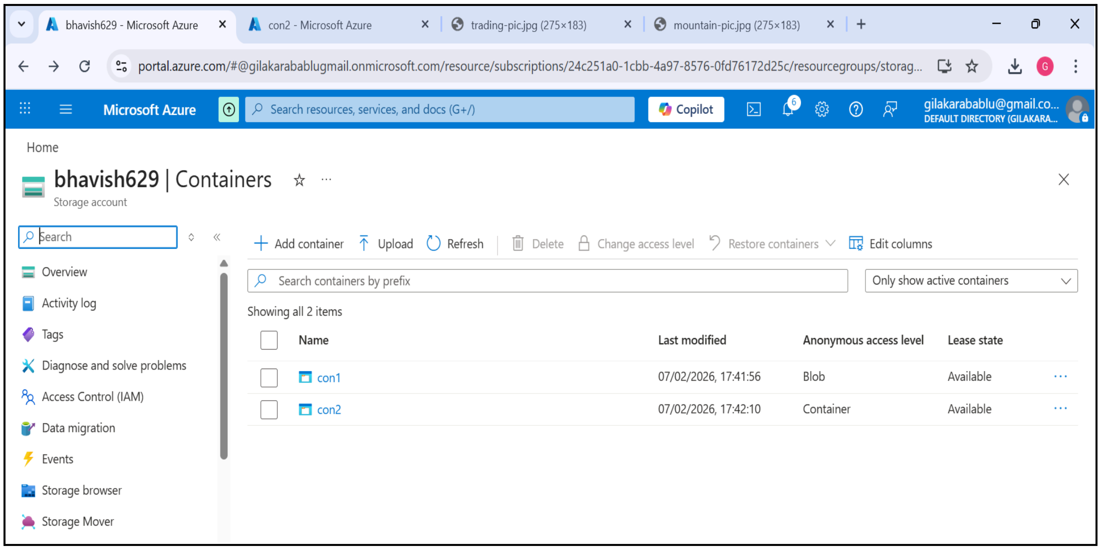


➢ Geo Redundant Storage (GRS):-

- Combines zone redundancy and geo-replication.
- Data is stored across multiple zones and also replicated to another region.
- Provides the highest level of durability and availability.



→ Create a storage account also with 2 containers.
Storage-rg,bhavish629,eastus


bhavish629 - Microsoft Azure
con2 - Microsoft Azure
trading-pic.jpg (275×183)
mountain-pic.jpg (275×183)
portal.azure.com/#@gilakarabablugmail.onmicrosoft.com/resource/subscriptions/24c251a0-1cbb-4a97-8576-0fd76172d25c/resourcegroups/storag...
Microsoft Azure
Search resources, services, and docs (G+/)
Copilot
gilakarabablu@gmail.co...
DEFAULT DIRECTORY (GILAKARA...
Home
Search
Overview
Activity log
Tags
Diagnose and solve problems
Access Control (IAM)
Data migration
Events
Storage browser
Storage Mover
Add container
Upload
Refresh
Delete
Change access level
Restore containers
Edit columns
Search containers by prefix
Only show active containers

bhavish629 | Containers

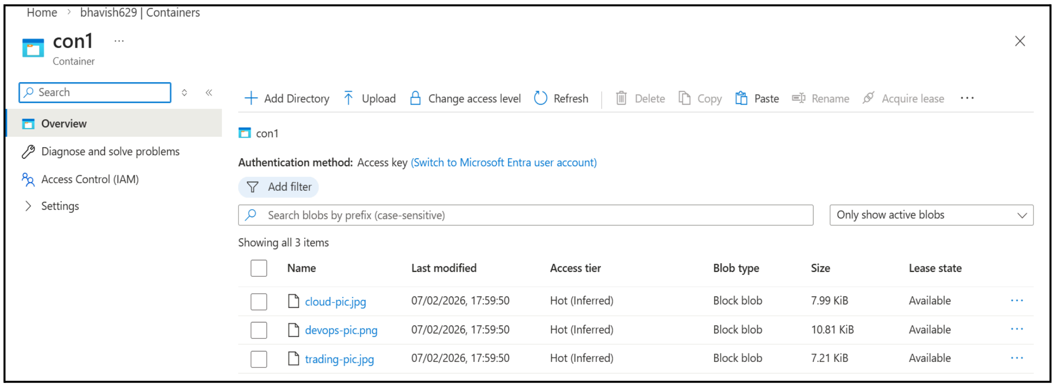
Storage account

Showing all 2 items

Name	Last modified	Anonymous access level	Lease state
con1	07/02/2026, 17:41:56	Blob	Available
con2	07/02/2026, 17:42:10	Container	Available

fig(1) created a storage account.

→ Now upload some files in the containers 1&2.



con1

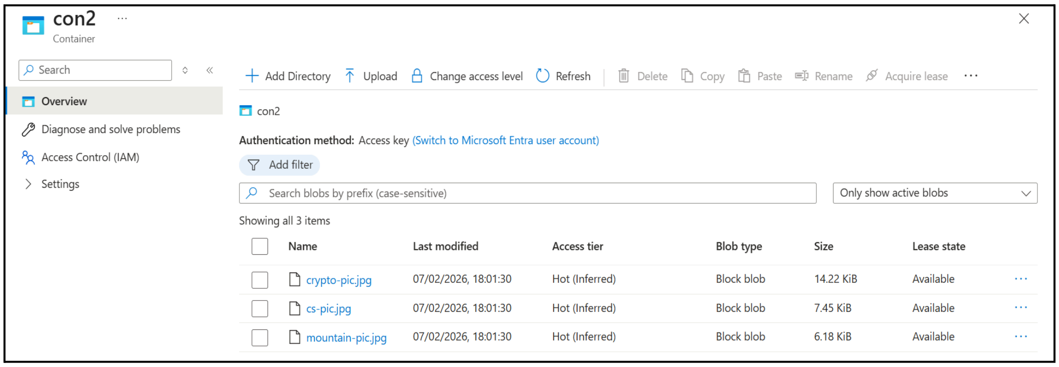
Container

con1

Authentication method: Access key (Switch to Microsoft Entra user account)

Showing all 3 items

Name	Last modified	Access tier	Blob type	Size	Lease state
cloud-pic.jpg	07/02/2026, 17:59:50	Hot (Inferred)	Block blob	7.99 KiB	Available
devops-pic.png	07/02/2026, 17:59:50	Hot (Inferred)	Block blob	10.81 KiB	Available
trading-pic.jpg	07/02/2026, 17:59:50	Hot (Inferred)	Block blob	7.21 KiB	Available



con2

Container

con2

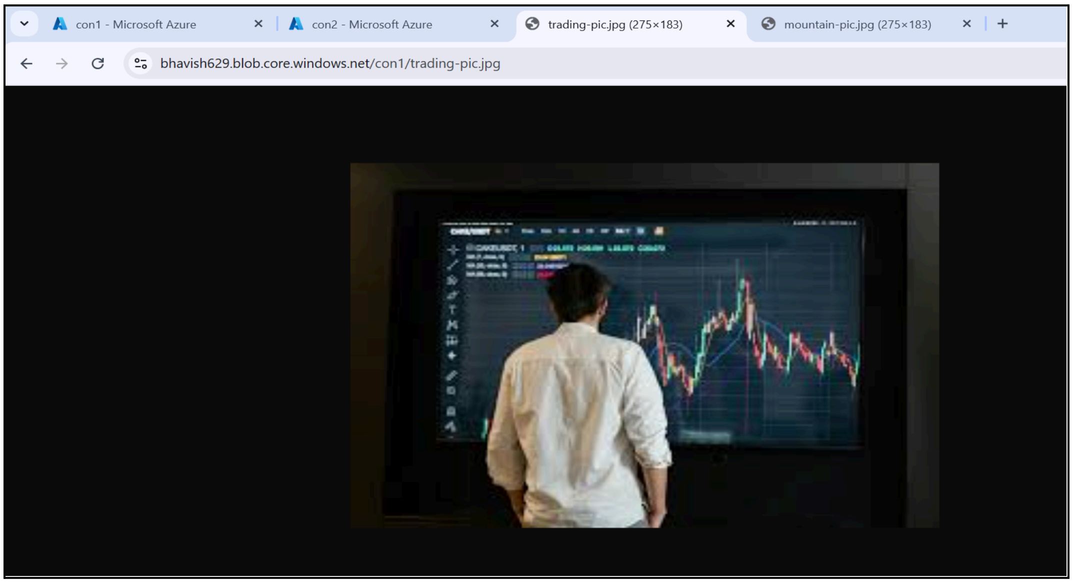
Authentication method: Access key (Switch to Microsoft Entra user account)

Showing all 3 items

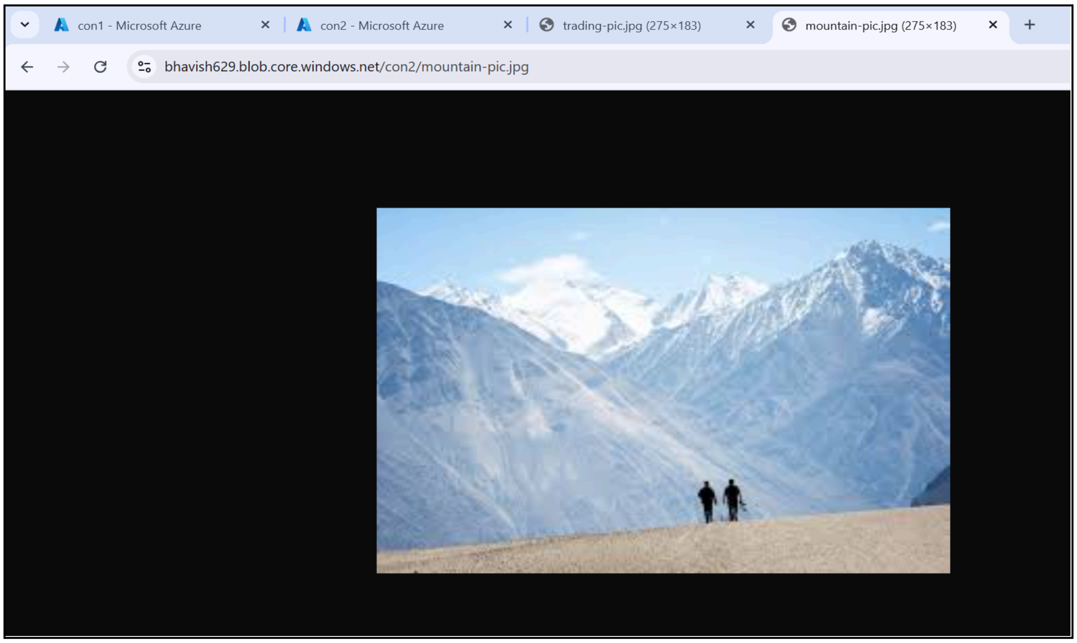
Name	Last modified	Access tier	Blob type	Size	Lease state
crypto-pic.jpg	07/02/2026, 18:01:30	Hot (Inferred)	Block blob	14.22 KiB	Available
cs-pic.jpg	07/02/2026, 18:01:30	Hot (Inferred)	Block blob	7.45 KiB	Available
mountain-pic.jpg	07/02/2026, 18:01:30	Hot (Inferred)	Block blob	6.18 KiB	Available

fig(2&3) uploaded files in the containers 1 & 2.

→ Now let's try to access the files through the internet.
→ there are two levels of access.
⦿ blob level :- accessing with the full url path of that image



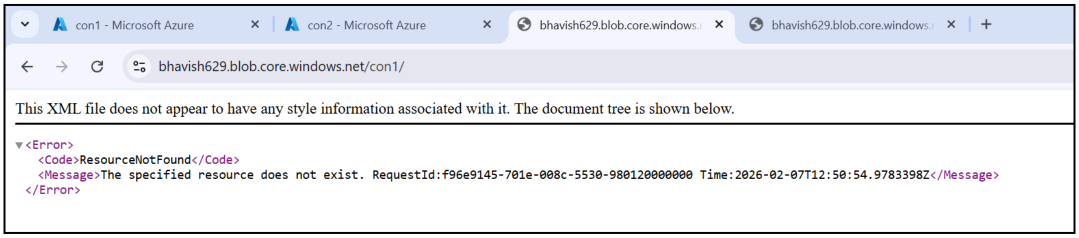
bhavish629.blob.core.windows.net/con1/trading-pic.jpg



bhavish629.blob.core.windows.net/con2/mountain-pic.jpg

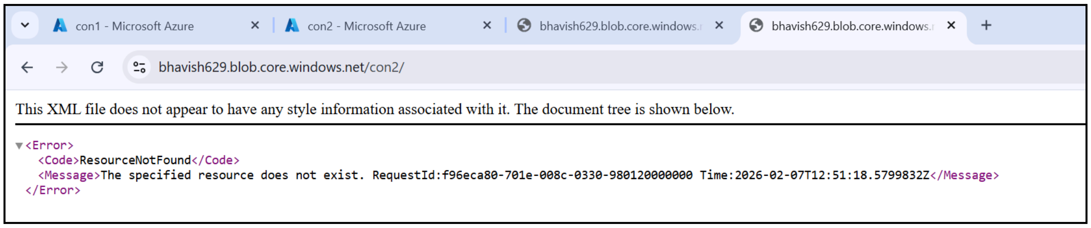
fig(4&5) blob level access.

⦿ Container level :- accessing only till the container path, excluding the image path.


con1 - Microsoft Azure
con2 - Microsoft Azure
bhavish629.blob.core.windows.
bhavish629.blob.core.windows.
bhavish629.blob.core.windows.net/con1/

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
   <Code>ResourceNotFound</Code>
   <Message>The specified resource does not exist. RequestId:f96e9145-701e-008c-5530-980120000000 Time:2026-02-07T12:50:54.9783398Z</Message>
 </Error>
```

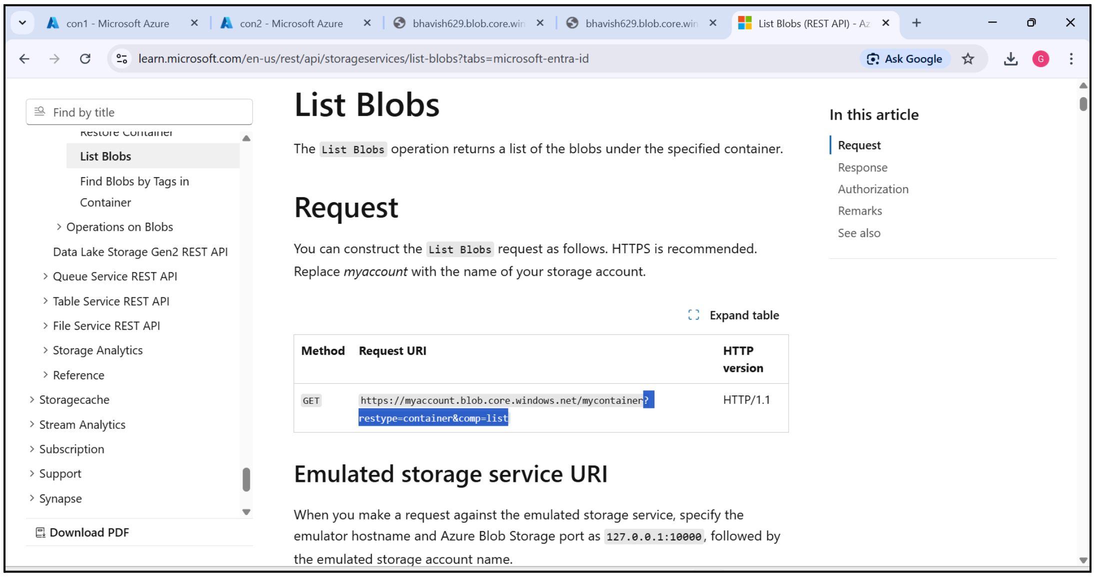

con1 - Microsoft Azure
con2 - Microsoft Azure
bhavish629.blob.core.windows.
bhavish629.blob.core.windows.
bhavish629.blob.core.windows.net/con2/

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
   <Code>ResourceNotFound</Code>
   <Message>The specified resource does not exist. RequestId:f96eca80-701e-008c-0330-980120000000 Time:2026-02-07T12:51:18.5799832Z</Message>
 </Error>
```

fig(6&7) cannot access the list of files at container level.

→ To get the list of files, -search blob list files,
   Copy method URI from the " ? " and paste it in the container path.



List Blobs

The `List Blobs` operation returns a list of the blobs under the specified container.

Request

You can construct the `List Blobs` request as follows. HTTPS is recommended. Replace *myaccount* with the name of your storage account.

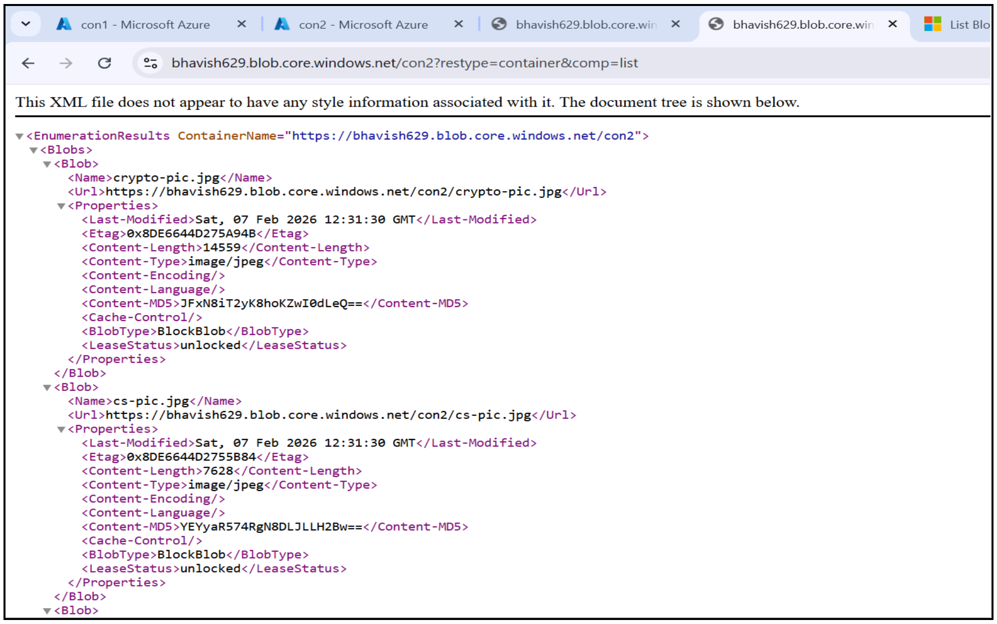
Expand table

Method	Request URI	HTTP version
`GET`	`https://myaccount.blob.core.windows.net/mycontainer?restype=container&comp=list`	HTTP/1.1

Emulated storage service URI

When you make a request against the emulated storage service, specify the emulator hostname and Azure Blob Storage port as `127.0.0.1:10000`, followed by the emulated storage account name.

fig(8) coping the method uri from the ?.

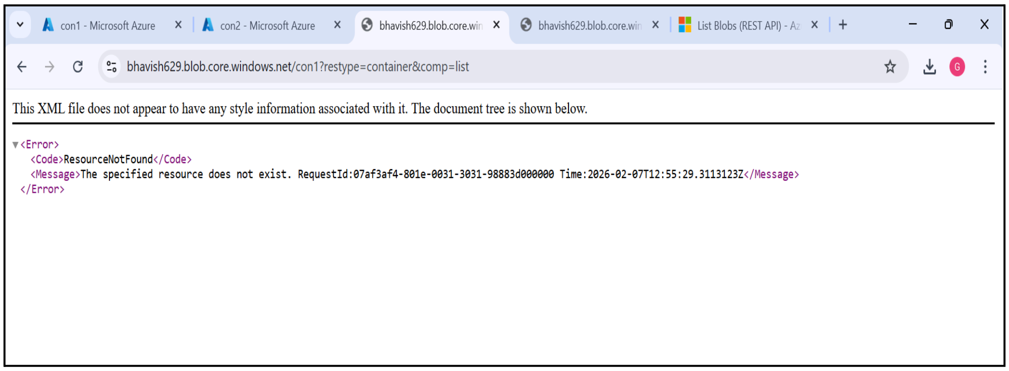


bhavish629.blob.core.windows.net/con2?restype=container&comp=list

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<EnumerationResults ContainerName="https://bhavish629.blob.core.windows.net/con2">
  ▼<Blobs>
    ▼<Blob>
       <Name>crypto-pic.jpg</Name>
       <Url>https://bhavish629.blob.core.windows.net/con2/crypto-pic.jpg</Url>
      ▼<Properties>
         <Last-Modified>Sat, 07 Feb 2026 12:31:30 GMT</Last-Modified>
         <Etag>0x8DE6644D275A94B</Etag>
         <Content-Length>14559</Content-Length>
         <Content-Type>image/jpeg</Content-Type>
         <Content-Encoding/>
         <Content-Language/>
         <Content-MD5>JFxN8iT2yK8hoKZwI0dLeQ==</Content-MD5>
         <Cache-Control/>
         <BlobType>BlockBlob</BlobType>
         <LeaseStatus>unlocked</LeaseStatus>
       </Properties>
     </Blob>
    ▼<Blob>
       <Name>cs-pic.jpg</Name>
       <Url>https://bhavish629.blob.core.windows.net/con2/cs-pic.jpg</Url>
      ▼<Properties>
         <Last-Modified>Sat, 07 Feb 2026 12:31:30 GMT</Last-Modified>
         <Etag>0x8DE6644D2755B84</Etag>
         <Content-Length>7628</Content-Length>
         <Content-Type>image/jpeg</Content-Type>
         <Content-Encoding/>
         <Content-Language/>
         <Content-MD5>YEYyaR574RgN8DLJLLH2Bw==</Content-MD5>
         <Cache-Control/>
         <BlobType>BlockBlob</BlobType>
         <LeaseStatus>unlocked</LeaseStatus>
       </Properties>
     </Blob>
    ▼<Blob>
```

fig(9) list of files in container 2.



bhavish629.blob.core.windows.net/con1?restype=container&comp=list

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
   <Code>ResourceNotFound</Code>
   <Message>The specified resource does not exist. RequestId:07af3af4-801e-0031-3031-98883d000000 Time:2026-02-07T12:55:29.3113123Z</Message>
 </Error>
```

fig(10) list of files in container 1.

→ Here, we are unable to access the files in con 1.

Since we have given

Container level access to con 2

Blob level access to con 1.

→ We can select this option for containers while creating them. And this is the main difference between them.

"We can access the list of files at the Container level but not at the Blob level."