

# Assignment 6

1. Create a Storage account and explore all the option while creating

## Resources :

<https://www.youtube.com/watch?v=2nPYLmciN4&pp=ygUSYXp1cmUgbWFuYWdlZCBkaXNrhttps://learn.microsoft.com/en-us/azure/storage/common/storage-account-overview>

2. Upload and access the blob
3. Go through different Auth. Tech and test the same
4. Try Azure storage explorer
5. Provision Access keys and use them for connection with storage account
6. Create a Shared access signature and check access scope
7. Create a stored access policy over shared access signature and check access scope
8. Learn about different access tiers and their use cases
9. Apply lifecycle policy over objects and test the same
0. test object replication in blob
1. Create a file share and test functionality
2. Create a Azure File sync

## Step 1: Create a Storage Account and Upload Blob

1. **Create a Storage Account:**
  - o Log in to the [Azure portal](#).
  - o Click on "Create a resource" > "Storage" > "Storage account".
  - o Fill out the necessary details like subscription, resource group, storage account name, region, performance, and redundancy options.
  - o Click "Review + create" and then "Create".
2. **Upload and Access a Blob:**
  - o After the storage account is created, navigate to it in the Azure portal.
  - o Go to "Containers" under "Blob service" and create a new container.
  - o Upload a blob (file) into the container using the Azure portal or Azure Storage Explorer.

## **Step 2: Explore Authentication Technologies**

1. **Access Keys:**
  - In the storage account settings, go to "Settings" > "Access keys".
  - Copy either of the two access keys provided.
  - Use the access key with Azure Storage SDKs, PowerShell, Azure CLI, or other tools to connect and manage blobs programmatically.
2. **Shared Access Signature (SAS):**
  - Create a SAS token with specific permissions (read, write, list, delete, etc.) and duration.
  - Test the SAS by accessing blobs using the generated URL with the SAS token appended.
3. **Stored Access Policy:**
  - Create a stored access policy with specific permissions (similar to SAS) and associate it with a container.
  - Generate a SAS token using the stored access policy and test its scope and permissions.

## **Step 3: Learn about Access Tiers**

1. **Access Tiers:**
  - Understand the different access tiers available for Azure Blob Storage: Hot, Cool, and Archive.
  - Learn their use cases based on data access patterns and cost considerations.
2. **Lifecycle Management:**
  - Apply a lifecycle policy to move blobs between access tiers automatically based on defined rules (e.g., move blobs to the Cool tier after 30 days, archive blobs after 90 days).

## **Step 4: Object Replication and File Share**

1. **Object Replication:**
  - Enable object replication in Azure Blob Storage to replicate data between storage accounts or regions asynchronously.
  - Test replication by uploading blobs and verifying their replication status in the destination storage account.
2. **Create and Test File Share:**
  - Navigate to your storage account and go to "File shares".
  - Create a new file share and upload files to it.
  - Access the file share using SMB or REST APIs.

## **Step 5: Azure File Sync**

1. **Azure File Sync:**
  - Install the Azure File Sync agent on your Windows Server.
  - Register the server with your Azure Storage account.
  - Set up sync groups and sync files between on-premises servers and Azure File shares.

## **Step 6: Azure Storage Explorer**

1. **Azure Storage Explorer:**
  - Download and install Azure Storage Explorer from [Microsoft Azure Storage Explorer](#).
  - Connect to your Azure Storage account using your storage account name and access key.

- Use Storage Explorer to manage containers, upload/download blobs, create SAS tokens, manage access policies, and view object replication status.

### **Summary**

By following these steps, you will have a comprehensive understanding of Azure Storage account features, including blob storage, authentication methods (access keys, SAS), access tiers, lifecycle management, object replication, file sharing, and Azure File Sync. Azure Storage Explorer will also help you manage and visualize your storage resources efficiently. Adjust configurations based on specific requirements and security best practices for your applications and data storage needs.