**Step 1: Provision ADLS Gen2 Storage Account**

1. Sign in to the **Azure Portal**.
2. Navigate to **Storage accounts → Create**.
3. Fill in details:
   * **Resource group**: RG-ADLS-Lab
   * **Storage account name**: adlsgen2lab<uniqueid>
   * **Region**: Select nearest region
   * **Performance**: Standard
   * **Redundancy**: Locally Redundant Storage (LRS)
   * **Advanced → Data Lake Storage Gen2**: Enable **Hierarchical namespace**
4. Click **Review + Create → Create**.
5. Verify that the storage account is created.

**CLI Alternative:**

az storage account create \

--name adlsgen2lab$RANDOM \

--resource-group RG-ADLS-Lab \

--location eastus \

--sku Standard\_LRS \

--kind StorageV2 \

--hierarchical-namespace true

**Step 2: Create Containers & Folders**

We’ll create three data zones: **raw, staged, curated**.

**Using Azure Portal:**

1. Go to your storage account → **Containers** → **+ Container**.
   * Create containers: raw, staged, curated.
2. Open the raw container → **+ Add directory** → Create folders like 2025/09/.

**Using CLI:**

az storage container create --account-name <storageaccount> --name raw

az storage container create --account-name <storageaccount> --name staged

az storage container create --account-name <storageaccount> --name curated

az storage fs directory create --account-name <storageaccount> --file-system raw --name "2025/09/"

**Step 3: Upload & Download Data**

**Using Azure Portal:**

1. Open the raw container.
2. Upload a sample file (sample.csv).
3. Download it back to verify.

**Using Storage Explorer:**

1. Connect to the storage account using **Azure AD authentication**.
2. Navigate to raw → drag & drop files.
3. Right-click on a file → Download.

**Using CLI:**

# Upload

az storage blob upload \

--account-name <storageaccount> \

--container-name raw \

--file ./sample.csv \

--name sample.csv

# Download

az storage blob download \

--account-name <storageaccount> \

--container-name raw \

--name sample.csv \

--file ./downloaded\_sample.csv

**Step 4: Access Control**

**RBAC Assignment:**

1. In Azure Portal → Storage Account → **Access Control (IAM)**.
2. Click **+ Add → Add role assignment**.
   * Role: Storage Blob Data Contributor
   * Assign to: A test Azure AD user
3. Log in as that user and verify they can access the container.

**Generate SAS Token:**

1. Go to **Storage account → Shared access signature**.
2. Allow **Blob service** → Permissions: Read, Write, List.
3. Set expiry date (e.g., +1 day).
4. Click **Generate SAS and connection string**.
5. Copy the **Blob SAS URL** and test with Storage Explorer or CLI.

**Test SAS via CLI:**

az storage blob list \

--container-name raw \

--account-name <storageaccount> \

--sas-token "<SAS\_TOKEN>"

**Step 5: Lifecycle Policy**

**Create Lifecycle Rule:**

1. Go to **Storage account → Data management → Lifecycle management**.
2. Click **+ Add rule**:
   * Rule name: archive-old-files
   * Scope: Apply to raw container
   * Condition: **Base blob, last modified > 7 days**
   * Action: Move to **Cool tier** or **Archive tier**
3. Save the rule.

**Test:**

* Upload a file and change its **last modified date** (simulate older file using AzCopy or wait 7+ days).
* Verify the blob tier changes automatically.

**Verification & Cleanup**

* Verify containers, folders, uploaded files, access, and lifecycle transitions.
* Cleanup resources to avoid cost:

az group delete --name RG-ADLS-Lab --yes --no-wait