
🧪 Azure Storage Accounts and Services – Hands-On Lab

📋 Lab Overview

This lab will guide you through:

1. Creating a Storage Account
2. Exploring Storage Services:
    - Blob Storage
    - File Share
    - Table Storage
    - Queue Storage
3. Uploading and managing data
4. Configuring access & security
5. Cleaning up resources

👩‍💻 Pre-requisites

- Azure Subscription (Pay-as-you-go or trial)
- Access to Azure Portal: https://portal.azure.com
- Azure Storage Explorer (Optional): Download

✅ Lab 1: Create a Storage Account

🔹 Step 1: Sign in to Azure Portal

1. Go to https://portal.azure.com
2. Sign in with your Azure credentials.

🔹 Step 2: Create Storage Account

1. Click on **"Create a resource"**
2. Search for **"Storage account"** and click **"Create"**
3. Fill in the **Basics** tab:
    - **Subscription:** Select your subscription
    - **Resource group:** Create new (e.g., `RG-StorageLab`) or use an existing one
    - **Storage account name:** Must be unique globally (e.g., `storagelab<yourinitials>`)
    - **Region:** Select closest region
    - **Performance:** Standard
    - **Redundancy:** Locally-redundant storage (LRS)
4. Click **"Review + Create"** → Click **"Create"**

⏳ Wait for deployment to complete.

✅ Lab 2: Explore Blob Storage

🔹 Step 1: Navigate to Blob Service

1. Go to your newly created storage account
2. Under **Data storage**, click on **"Containers"**
3. Click **"+ Container"**
    - **Name:** `blobcontainer`
    - **Public Access Level:** Private (no anonymous access)
    - Click **"Create"**

🔹 Step 2: Upload a Blob

1. Click on the newly created container (`blobcontainer`)
2. Click **"Upload"**
3. Choose a file (e.g., image or document) from your local machine
4. Click **"Upload"**

You've now uploaded a blob into Azure Blob Storage!

✅ Lab 3: Create a File Share

🔹 Step 1: Access File Shares

1. In your storage account, go to **"File shares"**
2. Click **"+ File share"**
    - **Name:** `filesharelab`

- **Quota:** Leave default (e.g., 5 GB)
- Click **"Create"**

🔹 Step 2: Upload a File

1. Click on the `filesharelab`
2. Click **"Upload"** → Select a file → Click **"Upload"**

You can also connect this to a Windows machine using the **SMB path** shown in the portal.

✅ Lab 4: Create a Table Storage

🔹 Step 1: Access Tables

Note: Table service is available only in GPv2 and GPv1 storage accounts.

1. In the storage account, go to **"Tables"**
2. Click **"+ Table"**
   - **Table Name:** `personinfo`
   - Click **"OK"**

🔹 Step 2: Add Data to the Table

You can use **Storage Explorer (GUI)** or write a small script with **Azure SDK** (e.g., PowerShell or Python) to add data.

Example using Storage Explorer:

1. Open Storage Explorer

2. Connect to your Azure account
3. Navigate to the `personinfo` table
4. Right-click → **Add Entity**
    - PartitionKey: `group1`
    - RowKey: `1`
    - Name: `Alice`
    - Email: `alice@example.com`
    - Click **Insert**

✅ Lab 5: Create and Test a Queue

🔹 Step 1: Access Queues

1. In your storage account, go to **"Queues"**
2. Click **"+ Queue"**
    - **Name:** `taskqueue`
    - Click **"OK"**

🔹 Step 2: Add a Message

1. Click on `taskqueue`
2. Click **"+ Add message"**
    - Message text: `Process Order #1234`

- Click **OK**

You've successfully added a message to your Azure Queue Storage!

✅ Lab 6: Configure Access & Security

🔹 Step 1: Generate SAS Token

1. In your storage account, go to **"Shared access signature"**
2. Select:
   - Allowed services: **Blob**
   - Allowed resource types: **Object**
   - Permissions: **Read**
   - Start/expiry time: Set a short duration
3. Click **"Generate SAS and connection string"**
4. Copy the **Blob SAS URL**

Test this URL in an incognito browser tab to verify restricted access.

🧹 Lab 7: Clean Up Resources

🔹 Delete Resource Group

To avoid charges:

1. In Azure Portal, go to **Resource groups**
2. Find your resource group (e.g., `RG-StorageLab`)

3. Click **"Delete resource group"**
4. Type the name to confirm → Click **"Delete"**

✅ Lab Complete!

You have now:

- Created an Azure Storage Account
- Worked with Blob, File, Table, and Queue services
- Configured basic access controls
- Cleaned up your environment
