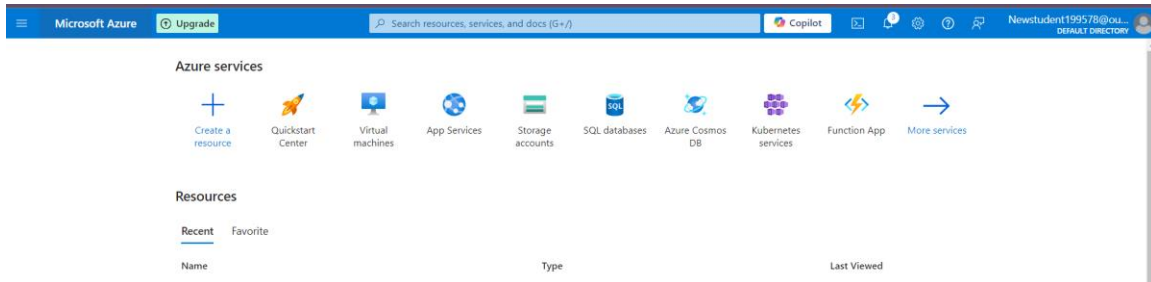
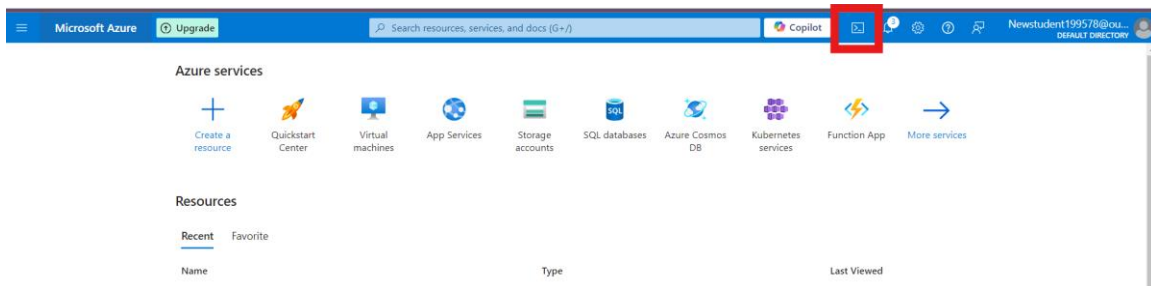


Step 1: Connect to Azure Cloud Shell

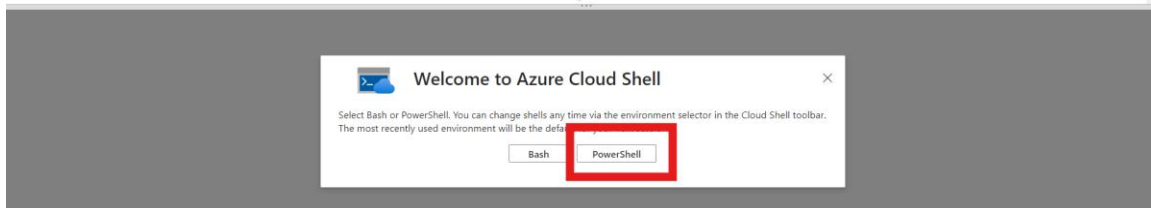
1. Go to the [Azure Portal](#).



2. Click on the **Cloud Shell** icon (a terminal icon) at the top-right corner of the portal.



3. Select **PowerShell**.



4. The Cloud Shell will automatically connect to your Azure environment.

Step 2: Create a Resource Group

Once inside Cloud Shell:

PowerShell:

1. Run the following command to create the resource group:

New-AzResourceGroup -Name "rg-1" -Location "South Central US"

1. Create 3 Storage Accounts with “Team” Tags

Create Storage Account for Team 1

az storage account create --name shahidteam1 --resource-group rg-1 --location "South Central US" --sku Standard_LRS --tags team="team1"

```
new [ ~ ]$ az storage account create --name shahidteam1 --resource-group rg-1 --location "South Central US" --sku Standard_LRS --tags team="team1"
{
  "accessTier": "Hot",
  "accountMigrationInProgress": null,
  "allowBlobPublicAccess": false,
  "allowCrossTenantReplication": false,
  "allowSharedKeyAccess": null,
  "allowedCopyScope": null,
  "azureFilesIdentityBasedAuthentication": null,
  "blobRestoreStatus": null,
  "creationTime": "2024-09-24T05:39:12.7647974000"
```

Step 2: Create a Blob Container

az storage container create --name mycontainer --account-name shahidteam1

```
Your Cloud Shell session will be ephemeral so no files or system changes will persist beyond your current session.
new [ ~ ]$ az storage container create --name mycontainer --account-name shahidteam1

There are no credentials provided in your command and environment, we will query for account key for your storage account.
It is recommended to provide --connection-string, --account-key or --sas-token in your command as credentials.

You also can add `--auth-mode login` in your command to use Azure Active Directory (Azure AD) for authorization if your login account is
For more information about RBAC roles in storage, visit https://docs.microsoft.com/azure/storage/common/storage-auth-aad-rbac-cli.

In addition, setting the corresponding environment variables can avoid inputting credentials in your command. Please use --help to get
{
  "created": true
}
new [ ~ ]$
```

Step 3: Upload Files


[+ Add Directory](#) [↑ Upload](#) [🔒 Change access level](#) [🔄 Refresh](#) | [🗑 Delete](#) [...](#)

[📁 Blob containers](#) > [📁 mycontainer](#)

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

[🔍 Add filter](#)

Showing all 1 items

<input type="checkbox"/>	Name	Last modified	Access tier	Blob type
<input type="checkbox"/>	 0DJVH91vU...	9/24/2024, 12:09:36 PM	Hot (Inferred)	Block blob

2. Create a CDN Profile

You can create a CDN profile using Azure CLI as well. Here's how:

```
az cdn profile create --name myCDNProfile --resource-group rg-1 --location southcentralus -  
-sku Standard_Microsoft
```

3. Create a CDN Endpoint and Connect to Azure Blob

Step 1: Create a CDN Endpoint

You can create a CDN endpoint that points to your Blob storage:

```
az cdn endpoint create --name myCDNEndpoint --profile-name myCDNProfile --resource-  
group rg-1 --origin mystorageaccount.blob.core.windows.net --origin-host-header  
mystorageaccount.blob.core.windows.net
```

Step 2: Access Uploaded Files

Once the CDN endpoint is created, you can access the uploaded files using the following URL structure:

```
https://myCDNEndpoint.azureedge.net/mycontainer/file.txt
```