Power Shell with ADLS & ADLA :

Learn how to use Azure PowerShell to create an Azure Data Lake Store account and perform basic operations such as create folders, upload and download data files, delete your account, etc.

Create an Azure Data Lake Store account

1. From your desktop, open a new Windows PowerShell window, and enter the following snippet to log in to your Azure account, set the subscription, and register the Data Lake Store provider. When prompted to log in, make sure you log in as one of the subscription admininistrators/owner:

   Copy

```
# Log in to your Azure account
Login-AzureRmAccount

# List all the subscriptions associated to your account
Get-AzureRmSubscription

# Select a subscription
Set-AzureRmContext -SubscriptionId <subscription ID>

# Register for Azure Data Lake Store
Register-AzureRmResourceProvider -ProviderNamespace "Microsoft.DataLakeStore"
```

2. An Azure Data Lake Store account is associated with an Azure Resource Group. Start by creating an Azure Resource Group.

   Copy

```
$resourceGroupName = "<your new resource group name>"
New-AzureRmResourceGroup -Name $resourceGroupName -Location "East US 2"
```

```
ResourceGroupName : mynewresourcegroup
Location          : eastus2
ProvisioningState : Succeeded
Tags              :
Permissions       :
                    Actions  NotActions
                    =======  ==========
                    *

ResourceId        : /subscriptions/                                /resourceGroups/mynewresourcegrou
```

3. Create an Azure Data Lake Store account. The name you specify must only contain lowercase letters and numbers.

Copy

```
 $dataLakeStoreName = "<your new Data Lake Store name>"
 New-AzureRmDataLakeStoreAccount -ResourceGroupName $resourceGroupName -Name
$dataLakeStoreName -Location "East US 2"
```

```
Id         : /subscriptions/                                /resourceGroups/hdiadlgroup/providers/Microsoft.DataL
             eStore/accounts/hdiadlstore
Location   : East US 2
Name       : hdiadlstore
Properties : Microsoft.Azure.Management.DataLake.Store.Models.DataLakeStoreAccountProperties
Tags       : {}
Type       : Microsoft.DataLakeStore/accounts
```

4. Verify that the account is successfully created.

Copy

```
 Test-AzureRmDataLakeStoreAccount -Name $dataLakeStoreName
```

The output for this should be True.

Create directory structures in your Azure Data Lake Store

You can create directories under your Azure Data Lake Store account to manage and store data.

1. Specify a root directory.

Copy

```
$myrootdir = "/"
```

2. Create a new directory called mynewdirectory under the specified root.

   Copy

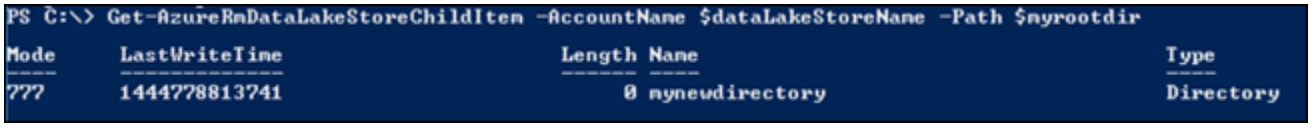
```
New-AzureRmDataLakeStoreItem -Folder -AccountName $dataLakeStoreName -Path
$myrootdir/mynewdirectory
```

3. Verify that the new directory is successfully created.

   Copy

```
Get-AzureRmDataLakeStoreChildItem -AccountName $dataLakeStoreName -Path
$myrootdir
```

   It should show an output like the following:



Upload data to your Azure Data Lake Store

You can upload your data to Data Lake Store directly at the root level or to a directory that you created within the account. The snippets below demonstrate how to upload some sample data to the directory (mynewdirectory) you created in the previous section.

If you are looking for some sample data to upload, you can get the Ambulance Data folder from the Azure Data Lake Git Repository. Download the file and store it in a local directory on your computer, such as C:\sampledata.

Copy

```
Import-AzureRmDataLakeStoreItem -AccountName $dataLakeStoreName -Path
"C:\sampledata\vehicle1_09142014.csv" -Destination
$myrootdir\mynewdirectory\vehicle1_09142014.csv
```

Rename, download, and delete data from your Data Lake Store

To rename a file, use the following command:

Copy

```
Move-AzureRmDataLakeStoreItem -AccountName $dataLakeStoreName -Path
$myrootdir\mynewdirectory\vehicle1_09142014.csv -Destination
$myrootdir\mynewdirectory\vehicle1_09142014_Copy.csv
```

To download a file, use the following command:

Copy

```
Export-AzureRmDataLakeStoreItem -AccountName $dataLakeStoreName -Path
$myrootdir\mynewdirectory\vehicle1_09142014_Copy.csv -Destination
"C:\sampledata\vehicle1_09142014_Copy.csv"
```

To delete a file, use the following command:

Copy

```
Remove-AzureRmDataLakeStoreItem -AccountName $dataLakeStoreName -Paths
$myrootdir\mynewdirectory\vehicle1_09142014_Copy.csv
```

When prompted, enter Y to delete the item. If you have more than one file to delete, you can provide all the paths separated by comma.

Copy

```
Remove-AzureRmDataLakeStoreItem -AccountName $dataLakeStoreName -Paths
$myrootdir\mynewdirectory\vehicle1_09142014.csv,
$myrootdir\mynewdirectoryvehicle1_09142014_Copy.csv
```

Delete your Azure Data Lake Store account

Use the following command to delete your Data Lake Store account.

Copy

```
Remove-AzureRmDataLakeStoreAccount -Name $dataLakeStoreName
```

When prompted, enter Y to delete the account.

Learn how to use Azure PowerShell to create Azure Data Lake Analytics accounts and then submit and run U-SQL jobs.

Log in to Azure

This tutorial assumes you are already familiar with using Azure PowerShell. In particular, you need to know how to log in to Azure. See the Get started with Azure PowerShell if you need help.

To log in with a subscription name:

Copy

```
Login-AzureRmAccount -SubscriptionName "ContosoSubscription"
```

Instead of the subscription name, you can also use a subscription id to log in:

Copy

```
Login-AzureRmAccount -SubscriptionId "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
```

If successful, the output of this command looks like the following text:

Copy

```
Environment           : AzureCloud
Account               : joe@contoso.com
TenantId              : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
SubscriptionId        : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
SubscriptionName      : ContosoSubscription
```

```
CurrentStorageAccount :
```

Preparing for the tutorial

The PowerShell snippets in this tutorial use these variables to store this information:
Copy

```
$rg = "<ResourceGroupName>"
$adls = "<DataLakeStoreAccountName>"
$adla = "<DataLakeAnalyticsAccountName>"
$location = "East US 2"
```

Get information about a Data Lake Analytics account

Copy

```
Get-AdlAnalyticsAccount -ResourceGroupName $rg -Name $adla
```

Submit a U-SQL job

Create a PowerShell variable to hold the U-SQL script.
Copy

```
$script = @"
@a  =
    SELECT * FROM
        (VALUES
            ("Contoso", 1500.0),
            ("Woodgrove", 2700.0)
        ) AS
              D( customer, amount );
OUTPUT @a
    TO "/data.csv"
    USING Outputters.Csv();

"@
```

Submit the script.

Copy

```
$job = Submit-AdlJob -AccountName $adla –Script $script
```

Alternatively, you could save the script as a file and submit with the following command:

Copy

```
$filename = "d:\test.usql"
$script | out-File $filename
$job = Submit-AdlJob -AccountName $adla –ScriptPath $filename
```

Get the status of a specific job. Keep using this cmdlet until you see the job is done.

Copy

```
$job = Get-AdlJob -AccountName $adla -JobId $job.JobId
```

Instead of calling Get-AdlAnalyticsJob over and over until a job finishes, you can use the Wait-AdlJob cmdlet.

Copy

```
Wait-AdlJob -Account $adla -JobId $job.JobId
```

Download the output file.

Copy

```
Export-AdlStoreItem -AccountName $adls -Path "/data.csv" -Destination "C:\data.csv"
```