Batch Data Utilization with Azure Data Lake Store & Analytics

Lab 8- Batch data utilization on Azure

Overview

In this lab, you will provision an ADLS account & store market stock data. Then execute computation jobs using Azure Data Lake Analytics.

What You’ll Need

To complete the labs, you will need the following:

* A web browser
* A Microsoft account
* A Microsoft Azure subscription
* A Microsoft Windows, Linux, or Apple Mac OS X computer on which the Azure CLI has been installed.
* The lab files for this course.

Provisioning and Configuring an Azure Data Lake Store

1. Using **File Explorer**, move to the [Github Link](https://github.com/imcuteani/PacktAzureTraining/tree/master/Day1/HOL/Lab8) ,right-click **Setup.cmd**, and then click **Run as administrator**.
2. In the **User Account Control** dialog box, click **Yes**.
3. Using Internet Explorer, log in to the Azure portal using the account associated with your Azure pass.
4. In the Azure portal, click **+ Create a resource**, click **Storage**, and then click **Data Lake Storage Gen 1**.
5. On the **New Data Lake Storage Gen 1** blade, in the **Name** box, type **stockstore<your name><date>**.
6. Under **Resource group**, click **Create new**, and then type **StockMarketRG**.
7. In the **Location** list, select your nearest location from the currently available Data Lake Store regions.
8. Leave all other settings at their defaults, and then click **Create**.
9. Wait until the storage has deployed before continuing with the demo.
10. In the Azure portal, click **All resources**, click **stockstore<your name><date>**, and then click **Data Explorer**.
11. On the **stockstore<your name><date>** blade, click **New Folder**.
12. In the **Create new folder** dialog box, type **StockMarket**, and then click **OK**.
13. Click the **StockMarket** folder, and then click **Upload**.
14. On the **Upload files** blade, click the **Select a file** button.
15. Go to the **OneDrive Lab8** folder in [this link](https://1drv.ms/u/s!Ai6oG1oOWxh_h-1PNOASdwdZ5iuyNQ) & download ‘StockPrices.csv’ file in your local drive.
16. In the **Open** dialog box of ADLS, click **StockPrices.csv**, and then click **Open**.
17. On the **Upload files** blade, click **Add selected files**.
18. After the file has uploaded, close the **Upload files** blade.
19. On the **stockstore<your name><date>** blade, click **StockPrices.csv**. This file contains some information about stock price movements (tickers, prices, and hour of day). The file contains over 19000 records.
20. Close the **File Preview** blade.

**Create and run a job using the Azure portal**

1. In the Azure portal, click **+ Create a resource**, click **Analytics**, and then click **Data Lake Analytics**.
2. On the **New Data Lake Analytics Account** blade, in the Name box, type **stocksdla<your name><date>**.
3. Under **Resource group**, click **Use existing**, and then click **StockMarketRG**.
4. In the **Location** list, select the same location as you used for the Data Lake Store.
5. Under **Data Lake Storage Gen 1**, click **Configure required settings**.
6. On the **Select Data Lake Storage Gen 1** blade, click **stockstore<your name><date>**
7. Leave all other settings at their defaults, and then click **Create**.
8. Wait until the account has deployed before continuing with the demo.
9. Click **All resources**, click **stocksdla<your name><date>**, and then click **+ New Job**.
10. On the **New Job** blade, in the **Job name** box, type **Ticker price**.
11. In the **AUs** box, type 2.
12. In the code pane, type the following U-SQL code:
13. @priceData =
14. EXTRACT Ticker string,
15. Price int,
16. HourOfDay int
17. FROM "/StockMarket/StockPrices.csv"
18. USING Extractors.Csv(skipFirstNRows: 1);
19. @maxPrices =
20. SELECT Ticker, MAX(Price) AS MaxPrice
21. FROM @priceData
22. GROUP BY Ticker;
23. OUTPUT @maxPrices
24. TO "/output/MaxPrices.csv"

USING Outputters.Csv(outputHeader: true);

1. On the **New Job** blade, click **Submit**.
2. On the **Ticker price** blade, notice the **Preparing**, and **Queued** steps.
3. When the job enters the Running phase, notice that the job details show how the job has been broken down into vertices (there will only be one vertex for this simple job).
4. On the **Ticker price** blade, click **Data**, click **Output**, and then click **MaxPrices.csv**. This file contains the results of the job—specifically, the maximum price for each stockticker.
5. Close the **File Preview** blade.
6. Click **All resources**, click **stockstore<your name><date>**, and then click **Data Explorer**.
7. On the **stockstore<your name><date>** blade, click **output**, and notice that this is where the results file is located (as specified in the OUTPUT statement in the U-SQL code).
8. Next to **MaxPrices.csv**, click the ellipses (...), and then click **Delete File**.
9. In the **Delete** dialog box, click **OK**.

Run a job using PowerShell

1. On the Windows **Start** menu, type **PowerShell ISE**, right-click **Windows PowerShell ISE**, and then click **Run as administrator**.
2. In the **User Account Control** dialog box, click **Yes**.
3. On the **File** menu, click **Open**.
4. In the **Open** dialog box, go to the folder **in** [**OneDrive Lab8**](https://1drv.ms/t/s!Ai6oG1oOWxh_h-1QnXUBrMo7PgbV1Q)**(**[this link](https://1drv.ms/t/s!Ai6oG1oOWxh_h-1QnXUBrMo7PgbV1Q)**)**, click **SubmitJob.ps1**, and then click **Open**.
5. Examine the script. This script logs in to your Azure account and retrieves the details of you ADLA account. The script then submits a USQL job using the code in the StockPriceJob.usql file and waits for the job to complete before downloading the results from your Data Lake Storage account to the MaxPrices.csv file in the Demo1 folder. The file StockPriceJob.usql contains the same code used in the previous demo. You can examine this file by using Notepad if you wish:
6. # Log in to Azure
7. Login-AzureRmAccount
8. $adla = "&lt;name of your ADLA account&gt;"
9. $adls = "<name of your Data Lake Store>"
10. $rg = "StockMarketRG"
11. # Verify that the ADLA account exists
12. Get-AdlAnalyticsAccount -ResourceGroupName $rg -Name $adla
13. # Submit an ADLA job
14. $scriptpath = "E:\Demofiles\Mod05\Demo1\StockPriceJob.usql"
15. $job = Submit-AdlJob -AccountName $adla –ScriptPath $scriptpath -Name "Demo"
16. # Repeatedly view the status of the job
17. $job = Get-AdlJob -AccountName $adla -JobId $job.JobId
18. # Wait for the job to complete
19. Wait-AdlJob -Account $adla -JobId $job.JobId
20. # Download the results

Export-AdlStoreItem -AccountName $adls -Path "/output/MaxPrices.csv" -Destination "E:\Demofiles\Mod05\Demo1\MaxPrices.csv"

1. Edit the script and replace **<name of your ADLA account>** with **stocksdla<your name><date>**.
2. Replace **<name of your Data Lake Store>** with **stockstore<your name><date>**.
3. On the **File** menu, click **Save**.
4. Highlight lines 1-9 using the mouse, and then on the toolbar, click **Run Selection**.
5. In the **Sign in to your account** dialog box, sign in using the details of the Microsoft account that is associated with your Azure Learning Pass subscription.
6. Examine the details of the Data Lake Analytics account that are returned by the **Get-AdlAnalyticsAccount** cmdlet.
7. In the script, highlight lines 11-22, and then on the toolbar, click **Run Selection**.
8. When the script has finished, examine the job details that have been returned.
9. Open the file **E:\Demofiles\Mod05\Demo1\MaxPrices.csv** using Excel. This file contains the same results as the previous demo.
10. Close Excel, and then close the PowerShell ISE.

**Run a job using Data Lake Tools for Visual Studio**

1. Start Visual Studio 2017.
2. On the **File** menu, click **New**, and then click **Project**.
3. In the **New Project** dialog box, in the **Templates** list, expand **Azure Data Lake**, click **U-SQL (ADLA)**, and then click **OK**.
4. In the Script.usql pane, type the following code
5. @priceData =
6. EXTRACT Ticker string,
7. Price int,
8. HourOfDay int
9. FROM "/StockMarket/StockPrices.csv"
10. USING Extractors.Csv(skipFirstNRows: 1);
11. @maxPrices =
12. SELECT Ticker, MAX(Price) AS MaxPrice
13. FROM @priceData
14. GROUP BY Ticker;
15. OUTPUT @maxPrices
16. TO "/output/MaxPrices.csv"

USING Outputters.Csv(outputHeader: true);

1. On the **View** menu, click **Cloud Explorer**.
2. In **Cloud Explorer**, if the account associated with your Azure Learning Pass subscription folder is not listed, perform the following steps:
   1. Click the **Azure Account settings** icon, and then click **Add an account**.
   2. In the **Sign in to your account** dialog box, sign in using the Microsoft account that is associated with your Azure Learning Pass subscription.
   3. In **Cloud Explorer**, select your Azure Learning Pass subscription, and then click **Apply**.
3. In the **Script.usql** pane, in the **(Local)** list, click **stocksdla<your name><date>**, and then click **Submit**. Note that Visual Studio Tools for Azure Data Lake Analytics provides a similar interface to that used in the Azure portal.
4. When the job has completed, on the **Job Graph** tab, double-click MaxPrices.csv. Note that, to see the file name in the Job Graph pane, you might need to close other panes, such as Solution Explorer.
5. Wait while the file is downloaded, and then review the results displayed in **File Preview** window. THe results should be the same as those generated by the previous tasks.
6. Close Visual Studio.