Processing Data with U-SQL – Basics (Code)

Updated on: 1/29/2017

## Exercise 1

@projects =

EXTRACT id int?,

url string,

owner\_id int?,

name string,

descriptor string,

language string,

created\_a DateTime?,

forked\_from int?,

deleted int?,

updated\_a DateTime?

FROM @"adl://adltrainingsampledata.azuredatalakestore.net/GHData/Projects.csv"

USING Extractors.Csv();

@projects =

SELECT \*

FROM @projects

ORDER BY 1 ASC

FETCH FIRST 50000 ROWS;

OUTPUT @projects

TO @"/GHDataSmall/projectssmall.csv"

ORDER BY updated\_a ASC

USING Outputters.Csv();

Takeaways

* Reads from one ADLS account and writes the another ADLS account
* Uses an absolute path to identify one ADLS account, and uses a relative path to specify the default ADLS account of the ADLA account being used
* Demonstrates how to get a subset of the rows (there are other techniques in U-SQL to do this, but the simplest is FETCH FIRST n ROWS)
* Note that the ORDER BY 1 is NOT saying “order by the first column” it really is saying order by the literal value “1” (i.e. don’t bother ordering the rows at all)

## Exercise 2

DECLARE @file\_projects\_large string = @"adl://adltrainingsampledata.azuredatalakestore.net/GHData/Projects.csv";

DECLARE @file\_projects\_small string = @"/GHDataSmall/projectssmall.csv";

@projects =

EXTRACT id int?,

url string,

owner\_id int?,

name string,

descriptor string,

language string,

created\_a DateTime?,

forked\_from int?,

deleted int?,

updated\_a DateTime?

FROM @file\_projects\_large

USING Extractors.Csv();

@projects =

SELECT \*

FROM @projects

ORDER BY 1 ASC

FETCH FIRST 50000 ROWS;

OUTPUT @projects

TO @file\_projects\_small

ORDER BY updated\_a ASC

USING Outputters.Csv();

## Exercise 3

This script creates a database if it does not exists and creates a TVF

CREATE FUNCTION HandsOnLab.dbo.Projects2()

RETURNS @projects

AS BEGIN

DECLARE @file\_projects\_large string = "adl://adltrainingsampledata.azuredatalakestore.net/GHData/Projects.csv";

@projects =

EXTRACT id int?,

url string,

owner\_id int?,

name string,

descriptor string,

language string,

created\_a DateTime?,

forked\_from int?,

deleted int?,

updated\_a DateTime?

FROM @file\_projects\_large

USING Extractors.Csv();

RETURN;

END;

## Exercise 4

@projects = HandsOnLab.dbo.Projects();

@stats =

SELECT language, COUNT(\*) AS Count

FROM @projects

GROUP BY language;

OUTPUT @stats

TO "/project\_language\_count.csv"

ORDER BY Count ASC

USING Outputters.Csv();

## Exercise 5

CREATE DATABASE IF NOT EXISTS HandsOnLab;

CREATE

ASSEMBLY ADLHandsOnLabLib

FROM "adl://adltrainingsampledata.azuredatalakestore.net/Assemblies/ADLHandsOnLabLib.dll";

REFERENCE ASSEMBLY HandsOnLab.ADLHandsOnLabLib;

@projects = HandsOnLab.dbo.Projects();

@projects =

SELECT \*

FROM @projects

ORDER BY 1 ASC

FETCH FIRST 1 ROWS;

OUTPUT @projects

TO "/output.csv"

ORDER BY updated\_a ASC

USING Outputters.Csv();

REFERENCE ASSEMBLY HandsOnLab.ADLHandsOnLabLib;

@projects = HandsOnLab.dbo.Projects();

@projects =

SELECT \*

FROM @projects

ORDER BY 1 ASC

FETCH FIRST 100 ROWS;

@rows =

SELECT

name,

ADLHandsOnLabLib.Helpers.DoubleIt(name) AS name2

FROM @projects;

OUTPUT @rows

TO "/output.csv"

USING Outputters.Csv();