Creating and Querying U-SQL Tables

Updated on: 1/5/2017

# Introduction

In this exercise, you will create a table with a predefined schema. You will then query data from the table you created.

Creating a table with U-SQL is similar to creating a table in a relational database such as SQL Server. You can either create a table with a predefined schema, or create a table and infer the schema from the query that populates the table (also known as CREATE TABLE AS SELECT or CTAS).

One of the benefits of creating a table, rather than a view or a table-valued function, is that the data is stored in an optimized format for the query processor to operate on. The data is indexed, partitioned, and stored in its native data type representation. In this first exercise we will create a table for our Projects data and then load that data into the table.

You will now persist the searchlog data in a schematized format in a table called Searchlog in your own database. The script will:

* Create a new database.
* Set the query context to the created database.
* Create tables. To illustrate both ways of creating a table, we create several tables over the next few exercises.
* Copy the following U-SQL script into the editor of your choice:

@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();

DROP DATABASE IF EXISTS ghanalytics;

CREATE DATABASE ghanalytics;

USE DATABASE ghanalytics;

DROP TABLE IF EXISTS Projects;

CREATE TABLE Projects

(

id int,

url string,

owner\_id int?,

name string,

descriptor string,

language string,

created DateTime,

forked\_from int?,

deleted int?,

updated DateTime?,

INDEX projects\_idx CLUSTERED (id ASC)

PARTITIONED BY HASH (created)

);

INSERT INTO Projects

SELECT \*

FROM @projects;

Instead of creating the table first, you can also create the table using a CREATE TABLE AS SELECT statement. This will create the table directly from a select or extract statement. The following script creates a Users table from the existing csv file on ADLS.

* Submit this U-SQL Script

USE DATABASE [ghanalytics];

@users =

EXTRACT id int,

login string,

name string,

company string,

city\_country string,

email string,

created DateTime?,

type string,

fake int?,

deleted int?,

longitude decimal?,

latitude decimal?,

country\_code string,

state string,

city string

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

USING Extractors.Csv(quoting: true);

CREATE TABLE Users

(

INDEX users\_idx CLUSTERED (id ASC)

PARTITIONED BY HASH (created)

) AS SELECT \* FROM @users;