**Analyze data in a storage account**

**Overview**

So far, we've covered scenarios where data resides in databases in the workspace. Now we'll show you how to work with files in storage accounts. In this scenario, we'll use the primary storage account of the workspace and container that we specified when creating the workspace.

* The name of the storage account: **ckctestdbmount1**
* The name of the container in the storage account: **demo**

**Create CSV and Parquet files in your storage account**

Run the following code in a notebook in a new code cell. It creates a CSV file and a parquet file in the storage account.

%%pyspark

df = spark.sql("SELECT \* FROM nyctaxi.passengercountstats")

df = df.repartition(1) # This ensures we'll get a single file during write()

df.write.mode("overwrite").csv("/NYCTaxi/PassengerCountStats\_csvformat")

df.write.mode("overwrite").parquet("/NYCTaxi/PassengerCountStats\_parquetformat")

**Analyze data in a storage account**

You can analyze the data in your workspace default ADLS Gen2 account or you can link an ADLS Gen2 or Blob storage account to your workspace through "**Manage**" > "**Linked Services**" > "**New**" (The steps below will refer to the primary ADLS Gen2 account).

1. In Synapse Studio, go to the **Data** hub, and then select **Linked**.
2. Go to **Azure Data Lake Storage Gen2** > **myworkspace (Primary - contosolake)**.
3. Select **users (Primary)**. You should see the **NYCTaxi** folder. Inside you should see two folders called **PassengerCountStats\_csvformat** and **PassengerCountStats\_parquetformat**.
4. Open the **PassengerCountStats\_parquetformat** folder. Inside, you'll see a parquet file with a name like part-00000-2638e00c-0790-496b-a523-578da9a15019-c000.snappy.parquet.
5. Right-click **.parquet**, then select **New notebook**, then select **Load to DataFrame**. A new notebook is created with a cell like this:

%%pyspark

abspath = 'abfss://users@contosolake.dfs.core.windows.net/NYCTaxi/PassengerCountStats\_parquetformat/part-00000-1f251a58-d8ac-4972-9215-8d528d490690-c000.snappy.parquet'

df = spark.read.load(abspath, format='parquet')

display(df.limit(10))

1. Attach to the Spark pool named **Spark1**. Run the cell.
2. Select back to the **users** folder. Right-click the **.parquet** file again, and then select **New SQL script** > **SELECT TOP 100 rows**. It creates a SQL script like this:

SELECT

TOP 100 \*

FROM OPENROWSET(

BULK 'https://contosolake.dfs.core.windows.net/users/NYCTaxi/PassengerCountStats\_parquetformat/part-00000-1f251a58-d8ac-4972-9215-8d528d490690-c000.snappy.parquet',

FORMAT='PARQUET'

) AS [result]

In the script window, make sure the **Connect to** field is set to the **Built-in** serverless SQL pool.

1. Run the script.