# **Apache Drill Workshop**

**1.Query Parquet File :**

SELECT \* FROM dfs.`<path-to-installation>/apache-drill-<version>/sample-data/region.parquet`;

Replace

<path-to-installation>/ - C:\installedsoftwares\ApacheDrill\

apache-drill-<version>/ - apache-drill-1.14.0/

Just drag and drop file and Drill is able to query

**2. Workspaces in Drill :**

1. Workspaces are **shortcuts** to the file system. You’ll want to use them when you have lengthy file paths.

* They work in any “file based” storage plugin (IE: S3, Hadoop, Local File System)

Example Of Workspace Plugin :

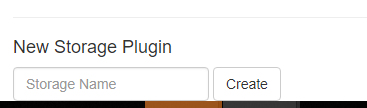
SHOW FILES IN <workspace>

SHOW FILES IN dfs ;

**3. Create a new workspace for the workshop :**

**Step 1 :** Open Drill Web UI

**Step 2:** Got to Storage tab , Create a new storage plugin

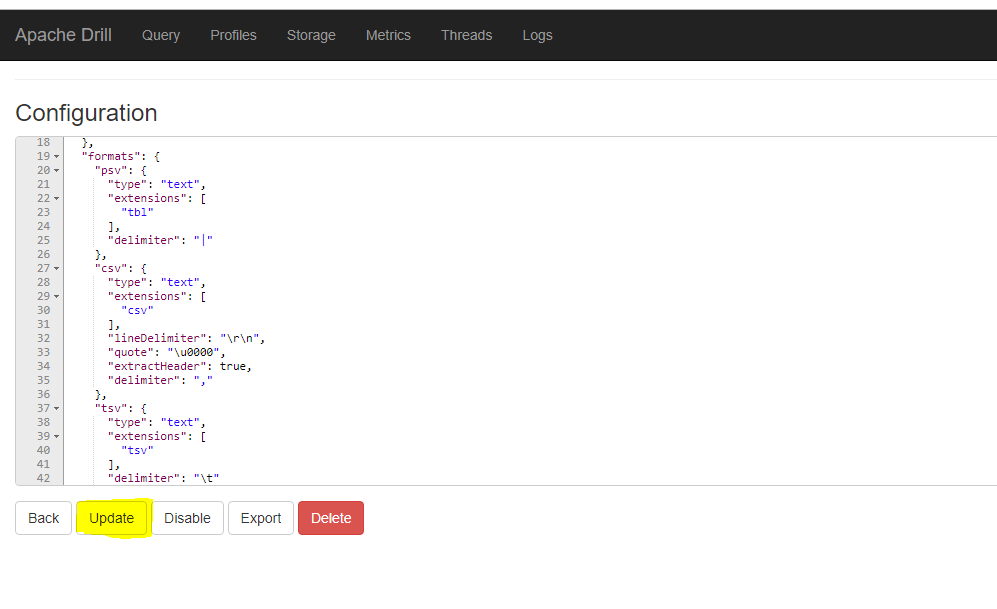


Storage Name – workshop

Give a name to storage plugin and click on create

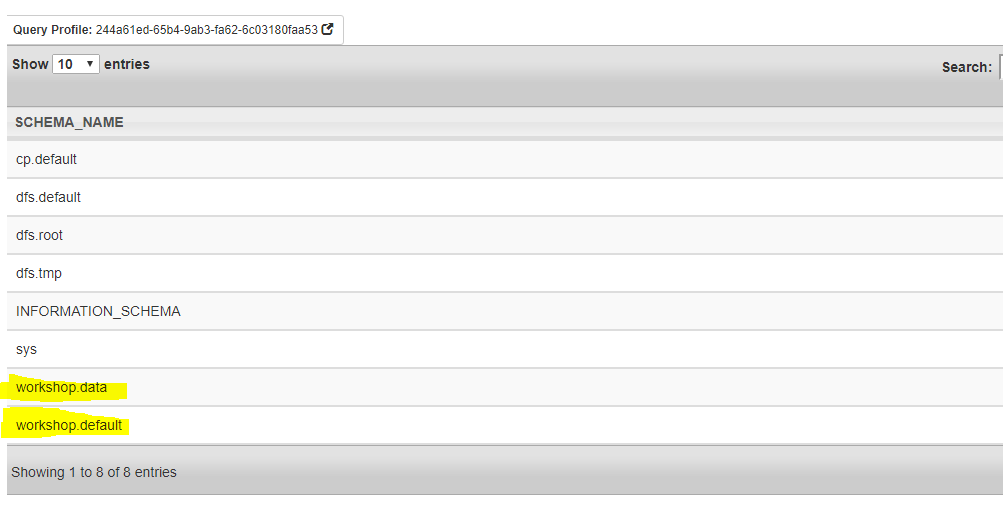


Add the above lines from the file and click on Update



**Step 3 :** Verify Storage Plugin creation :

Show Schemas;



Copy <path-to-installation>\apache-drill-<version>\sample-data/nation.parquet to

Plugin creation directory i.e. <StoragePluginDirectory>\data

**Run the following commands in Drill Web UI/Drill Explorer/sqlline :**

Download the workshop datasets

USE `workshop`.`data`.;

SELECT \* FROM `workshop`.`data`.`./nation.parquet`

Observe the shorten of the path

**5.Query CSV Files :**

SELECT \* FROM `workshop`.`data`.`./csvfiles`

Observe here you don’t have to give the filename , Drill has the ability to read the data at directory level

**6.Browse data set without creating the schema :**

SELECT \* FROM `workshop`.`data`.`./salaries`

Observe that data has headers and we don’t have to create any schema or define any data type while reading the data set

**7.Arrays in Drill :**

SELECT \* FROM `workshop`.`data`.`./customer/customer\_data.csv`

Drill can read the data set without headers too

**8.Access data individual columns**

SELECT columns[0] AS first\_name, columns[1] AS last\_name,columns[2] AS birthday FROM `workshop`.`data`.`./customer/customer\_data.csv`

**9.Parse JSON Files :**

SELECT \* FROM `workshop`.`data`.`./json/records.json`

Looks Easy ?

Open splits.json in editor and observer the data

SELECT FLATTEN(data) AS row\_data FROM `workshop`.`data`.`./json/split.json`

**10.Custom UDF usage : Mask Data**

SELECT MASK(first\_name, '\*' , 3) FIRST , MASK(last\_name, '#', 7) LAST FROM cp.`employee.json` LIMIT 5;

**11.Analysing Yelp Data Set :**

SELECT attributes from `workshop`.`data`.`./json/yelp\_academic\_dataset\_business.json` limit 10;

SELECT name, state, city, `review\_count` FROM `workshop`.`data`.`./json/yelp\_academic\_dataset\_business.json`WHERE review\_count > 1000 ORDER BY `review\_count` desc limit 10;

SELECT stars,TRUNC(AVG(review\_count)) reviewsavg FROM `workshop`.`data`.`./json/yelp\_academic\_dataset\_business.json` GROUP BY stars ORDER BY stars desc;

SELECT name, city, review\_count, row\_number()

OVER (PARTITION BY city ORDER BY review\_count DESC) AS rownum

FROM `workshop`.`data`.`./json/yelp\_academic\_dataset\_business.json` limit 10 ;

**12. Read MAP Data Using KVGEN**

SELECT \* FROM `workshop`.`data`.`./json/columns.json`

SELECT FLATTEN( KVGEN( first\_name ) )['value'] AS firstname

FROM `workshop`.`data`.`./json/columns.json`

**12.Read httpd files :**

SELECT \* FROM `workshop`.`data`.`./httpd`

**13. Join Two Different format datasets :**

SELECT \* FROM

`workshop`.`data`.`./flights/flights.csvh` AS flights

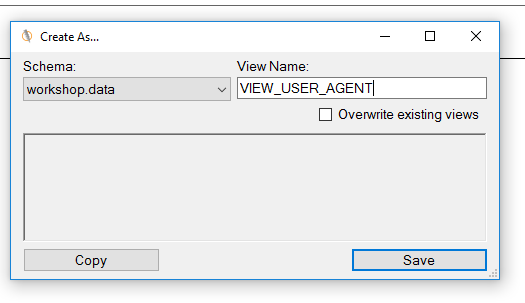
LEFT OUTER JOIN

`workshop`.`data`.`./airports/airports.json` AS airports

ON

TRIM(flights.ORIGIN\_AIRPORT) = TRIM(airports.IATA\_CODE) limit 10 ;

**14. CREATE VIEWS :**

****

**Syntax to create view:**

CREATE [OR REPLACE] VIEW [workspace.]view\_name [ (column\_name [, ...]) ] AS query;

15. **Query Images:**

select FileName, \* from `workshop`.`data`.`./images/pexels-photo-248797.jpeg`;

SELECT FileName, PixelWidth, PixelHeight FROM `workshop`.`data`.`./images` where PixelWidth >= 640 and PixelHeight >= 480

**Connect to Other JDBC Data Sources**

Update the Drill Storage plugin with below:

{

“type”:”jdbc”,

“driver”:”com.mysql.jdbc.Driver”,

“url”:”jdbc:mysql://localhost:3306",

“username”:”merlinuser”,

“password”:”merlinuser”,

“enabled”:true

}