15: Docker Tutorial: Hive & parquet-tools – csv to parquet on Cloudera quickstart

Posted on June 2, 2019

CSV is a row based storage, and Parquet is **columnar** in nature, and it is designed from the ground up for efficient storage, compression and encoding, which gives better performance.

Run the cloudera/quickstart

This extends Docker Tutorial: BigData on Cloudera quickstart via Docker.

300+ Java Interview FAQs

300+ Java FAQs 🔥

16+ Java Key Areas Q&As

150+ Java Architect FAQs

80+ Java Code Quality Q&As

150+ Java Coding Q&As

300+ Big Data Interview FAQs

300+ Big Data FAQs 🔥

Tutorials - Big Data

TUT - ⏯ Starting Big Data

TUT - Starting Spark & Scala

Step 1: Run the container on a command line.

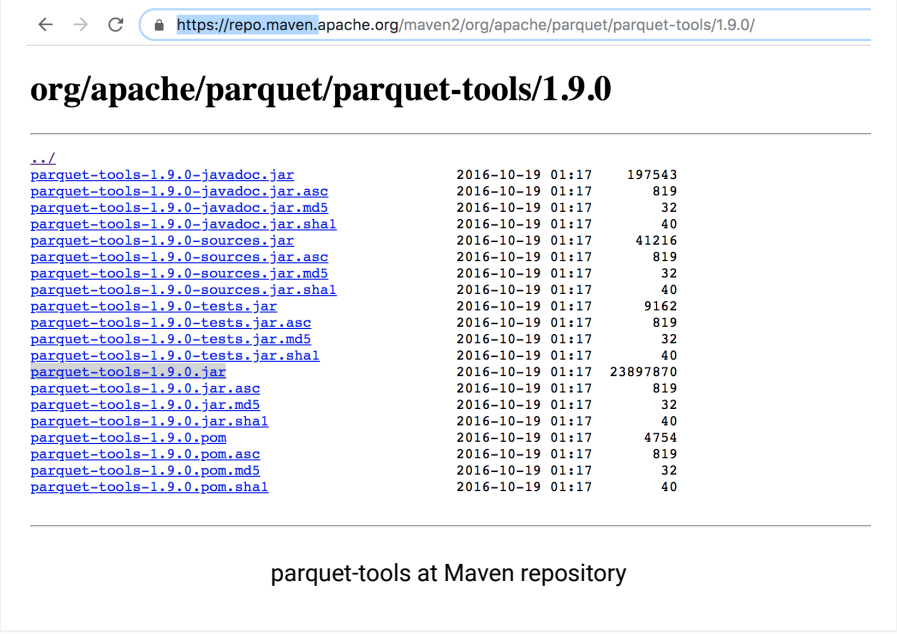
```
~/projects/docker-hadoop]$ docker run --hostname=qu
--privileged=true -t -i -v /Users/arulkumarankumar
--publish-all=true -p 8888:8888 -p 80:80 -p 7180:71
```

Get parquet-tools

Step 2: Install **wget**. The "**uname -a**" gets you the info of the kernel.

```
[root@quickstart /]# sudo yum install wget
```

Step 3: Download "parquet-tools" from maven repository using wget.



parquet-tools at Maven repository

```
[root@quickstart /]# wget https://repo.maven.apache
```

Hive table over .csv

Step 4: Create a csv file "**employee.csv**" in the local file system.

```
[root@quickstart /]# touch employee.csv
[root@quickstart /]# vi employee.csv

```

```
John,Samuel,IT
Peter,Smith,Finance
Sean,Mendis,Marketing
```

Step 5: Copy this file onto HDFS file system.

```
[root@quickstart /]# hdfs dfs -copyFromLocal emplo
```

Step 6: Connect to hive 2 via **beeline**. Beeline is a thin client that also uses the Hive JDBC driver but instead executes queries through HiveServer2, which allows multiple concurrent client connections and supports authentication.

```
[root@quickstart /]# beeline -u  jdbc:hive2://quic

```

Step 7: Create a database named "**mydb**".

```
0: jdbc:hive2://quickstart.cloudera:10000> create

```

Step 8: An external table created over a folder in HDFS will not delete the file even if the table is dropped.

```
CREATE EXTERNAL TABLE IF NOT EXISTS mydb.tbl_emplo
(first_name STRING,surname STRING, department STRI
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE
LOCATION 'hdfs://quickstart.cloudera:8020/user/roo

```

Hive table over parquet

Step 9: An external table created over a folder in HDFS will not delete the file even if the table is dropped.

```
CREATE EXTERNAL TABLE IF NOT EXISTS mydb.tbl_parqu
(first_name STRING,surname STRING, department STRI
STORED AS PARQUET
LOCATION 'hdfs://quickstart.cloudera:8020/user/roo
```

Insert data into tbl_parquet_employee

```
0: jdbc:hive2://quickstart.cloudera:10000>0: jdbc:
```

```
0: jdbc:hive2://quickstart.cloudera:10000>0: jdbc
....
+----------------------------------+-------------
| tbl_parquet_employee.first_name  | tbl_parquet_
+----------------------------------+-------------
| John                             | Samuel
| Peter                            | Smith
| Sean                             | Mendis
+----------------------------------+-------------
3 rows selected (0.58 seconds)
0: jdbc:hive2://quickstart.cloudera:10000>
```

List the parquet file

Step 10: Exit out with “ctrl+C”. List the file:

```
[root@quickstart /]# hdfs dfs -ls hdfs://quickstar
Found 1 items
-rwxrwxrwx   1 anonymous supergroup        540 201
```

cat the parquet file with parquet-tools on HDFS

```
[root@quickstart /]# hadoop jar parquet-tools-1.9
....
first_name = John
surname = Samuel
department = IT

first_name = Peter
surname = Smith
department = Finance

first_name = Sean
surname = Mendis
department = Marketing

[root@quickstart /]#
```

copy the parquet file to local file system

```
[root@quickstart /]# [root@quickstart /]# hdfs dfs
```

cat the parquet file with parquet-tools on local file system

```
[root@quickstart /]# hadoop jar parquet-tools-1.9
....
first_name = John
surname = Samuel
department = IT

first_name = Peter
surname = Smith
department = Finance

first_name = Sean
surname = Mendis
department = Marketing

[root@quickstart /]#
```

Unlike “**hdfs**” commands that work only on hdfs, “**hadoop**” command will work on both “hdfs” and “local” file systems.

‹ 14: Docker Tutorial: Hive (via beeline) on Cloudera quickstart

16: Docker Tutorial: Apache Spark (spark-shell) & parquet-tools – csv to parquet on Cloudera quickstart ›

Disclaimer