## Hive Assignment #2

Data Sets: NASDAQ Exchange Daily 1970-2010 Open, Close, High, Low and Volume Download links: NASDAQ dividends A.csv NASDAQ daily prices A sample.csv

Summary of data: There are two types of data

1. File name starting with NASDAQ\_daily\_prices\*:

These files is a CSV (comma separated values) file which contains following fields which are self explanatory.

exchange,stock\_symbol,date,stock\_price\_open,stock\_price\_high,stock\_price\_low,stock\_price,\_close,stock\_volume,stock\_p

2. File name starting with NASDAQ\_daily\_prices\*:

These files is a CSV (comma separated values) file which contains following fields which are self explanatory.

exchange,stock\_symbol,date,dividends

-- 1. Create an external table for NASDAQ daily prices data set.

```
DROP TABLE IF EXISTS daily_prices;

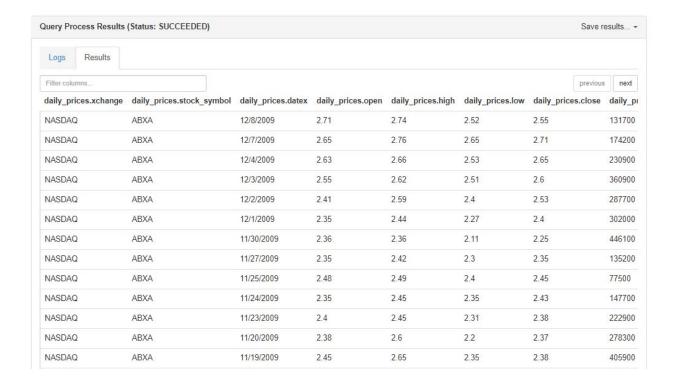
CREATE EXTERNAL TABLE daily_prices

(xchange STRING, stock_symbol STRING,
dateX STRING, open DOUBLE, high DOUBLE,
low DOUBLE, close DOUBLE, volume INT, theta DOUBLE)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
LOCATION'/user/maria_dev/hive_assignment'
tblproperties ("skip.header.line.count"="1");

LOAD DATA INPATH '/user/maria_dev/NASDAQ_daily_prices_A_sample.csv'
OVERWRITE INTO TABLE daily_prices;

--test
SELECT * FROM daily_prices;
```



## -- 2. Create an external table for NASDAQ dividends data set.

DDOD TABLE TE EVICTO dividende.

```
DROP TABLE IF EXISTS dividends;

CREATE EXTERNAL TABLE dividends

(exchange_name STRING, stock_symbol STRING, dates STRING,

dividends DOUBLE)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

LOCATION'/user/maria_dev/hive_assignment'

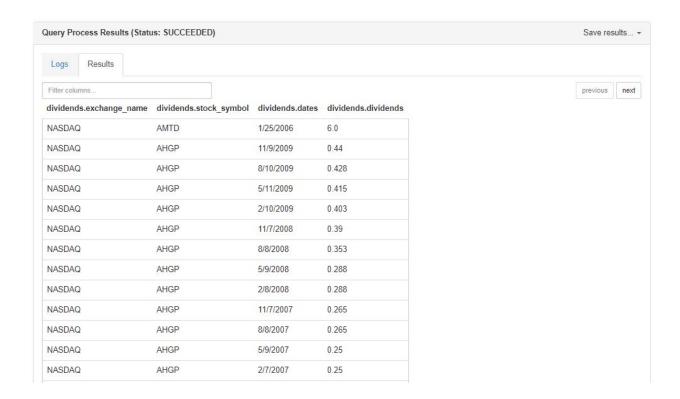
tblproperties ("skip.header.line.count"="1");

LOAD DATA INPATH '/user/maria_dev/NASDAQ_dividends_A.csv'

OVERWRITE INTO TABLE dividends;

--test

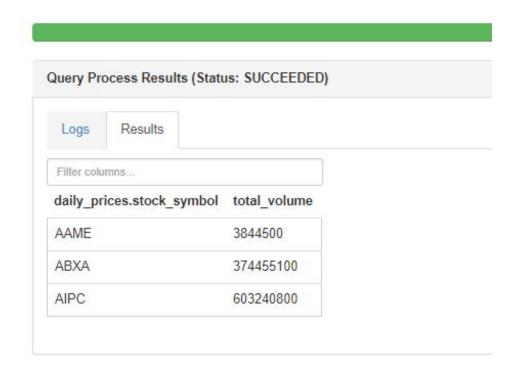
SELECT * FROM dividends LIMIT 100;
```



--3. Find out total volume sale for each stock symbol which has closing price more than \$5.

------

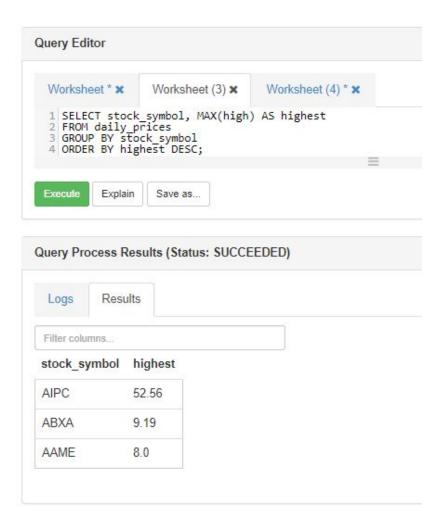
```
SELECT stock_symbol , SUM(volume) AS total_volume
FROM daily_prices WHERE close>5.00
GROUP BY stock_symbol
LIMIT 20;
```



--4. Find out highest price in the history for each stock symbol.

\_\_\_\_\_

```
SELECT stock_symbol, MAX(high) AS highest FROM daily_prices
GROUP BY stock_symbol
ORDER BY highest DESC;
```



--5. Find out highest dividends given for each stock symbol in entire history.

```
SELECT stock_symbol, MAX(dividends) AS max_divs
FROM dividends
GROUP BY stock_symbol
ORDER BY max_divs;
```



--6. Find out highest price and highest dividends for each stock symbol if highest -- price and highest dividends exist.

\_\_\_\_\_

```
(
SELECT stock_symbol, MAX(high) AS max_price
FROM daily_prices
GROUP BY stock_symbol
) AS ax
LEFT JOIN
(
SELECT stock_symbol, MAX(dividends) AS max_divs
FROM dividends
GROUP BY stock_symbol
) AS bx
ON (ax.stock_symbol=bx.stock_symbol)
WHERE ax.max_price>0;
```



