

# 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 are CSV (comma separated values) files which contain the 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 are CSV (comma separated values) files which contain the 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
    (exchange 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;
```

Query Process Results (Status: SUCCEEDED)								Save results... ▾
<div> <div>Logs</div> <div>Results</div> </div>								
<div> <div>Filter columns...</div> <div>previous</div> <div>next</div> </div>								
daily_prices.exchange	daily_prices.stock_symbol	daily_prices.date	daily_prices.open	daily_prices.high	daily_prices.low	daily_prices.close	daily_prices.volume	
NASDAQ	ABXA	12/8/2009	2.71	2.74	2.52	2.55	131700	
NASDAQ	ABXA	12/7/2009	2.65	2.76	2.65	2.71	174200	
NASDAQ	ABXA	12/4/2009	2.63	2.66	2.53	2.65	230900	
NASDAQ	ABXA	12/3/2009	2.55	2.62	2.51	2.6	360900	
NASDAQ	ABXA	12/2/2009	2.41	2.59	2.4	2.53	287700	
NASDAQ	ABXA	12/1/2009	2.35	2.44	2.27	2.4	302000	
NASDAQ	ABXA	11/30/2009	2.36	2.36	2.11	2.25	446100	
NASDAQ	ABXA	11/27/2009	2.35	2.42	2.3	2.35	135200	
NASDAQ	ABXA	11/25/2009	2.48	2.49	2.4	2.45	77500	
NASDAQ	ABXA	11/24/2009	2.35	2.45	2.35	2.43	147700	
NASDAQ	ABXA	11/23/2009	2.4	2.45	2.31	2.38	222900	
NASDAQ	ABXA	11/20/2009	2.38	2.6	2.2	2.37	278300	
NASDAQ	ABXA	11/19/2009	2.45	2.65	2.35	2.38	405900	

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

```

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;

```

Query Process Results (Status: SUCCEEDED)				Save results... ▾
<div> <div>Logs</div> <div>Results</div> </div>				
<div> <div>Filter columns...</div> <div>previous</div> <div>next</div> </div>				
dividends.exchange_name	dividends.stock_symbol	dividends.dates	dividends.dividends	
NASDAQ	AMTD	1/25/2006	6.0	
NASDAQ	AHGP	11/9/2009	0.44	
NASDAQ	AHGP	8/10/2009	0.428	
NASDAQ	AHGP	5/11/2009	0.415	
NASDAQ	AHGP	2/10/2009	0.403	
NASDAQ	AHGP	11/7/2008	0.39	
NASDAQ	AHGP	8/8/2008	0.353	
NASDAQ	AHGP	5/9/2008	0.288	
NASDAQ	AHGP	2/8/2008	0.288	
NASDAQ	AHGP	11/7/2007	0.265	
NASDAQ	AHGP	8/8/2007	0.265	
NASDAQ	AHGP	5/9/2007	0.25	
NASDAQ	AHGP	2/7/2007	0.25	

--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;
```

Query Process Results (Status: SUCCEEDED)

Logs

Results

Filter columns...

daily_prices.stock_symbol	total_volume
AAME	3844500
ABXA	374455100
AIPC	603240800

--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;
```

Query Editor

Worksheet \*✕

Worksheet (3)✕

Worksheet (4) \*✕

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

Execute

Explain

Save as...

Query Process Results (Status: SUCCEEDED)

Logs

Results

Filter columns...

stock\_symbolhighest

AIPC	52.56
ABXA	9.19
AAME	8.0

--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;
```

**Query Editor**

Worksheet \* ✕    Worksheet (1) ✕    Worksheet (2) ✕

```

1 SELECT stock_symbol, MAX(dividends) AS max_divs
2 FROM dividends
3 GROUP BY stock_symbol
4 ORDER BY max_divs DESC;

```

Execute   Explain   Save as...

**Query Process Results (Status: SUCCEEDED)**

Logs   Results

Filter columns...

stock_symbol	max_divs
AIPC	51.65
ARDNA	20.0
ALOY	16.6
AMIE	12.0
AFCE	12.0
ATEA	10.25
ABXA	9.09
AAME	7.63
AMTD	6.0
ATRO	5.5
ALDA	5.0
ACCL	4.95
ADP	4.925
ARKR	3.35
ASML	2.84
AACC	2.45
ADPT	2.343
AMOV	1.862

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

---

```

SELECT bx.stock_symbol, ax.max_price, bx.max_divs
FROM

```

```

(
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;

```

#### Query Process Results (Status: SUCCEEDED)

Logs

Results

Filter columns...

**bx.stock\_symbol** **ax.max\_price** **bx.max\_divs**

AAME	8.0	7.63
------	-----	------

ABXA	9.19	9.09
------	------	------

AIPC	52.56	51.65
------	-------	-------

## Query Editor

Worksheet \* x

Worksheet (3) x

Worksheet (4) \* x

```
1 SELECT bx.stock_symbol,ax.max_price,bx.max_divs
2 FROM
3 (
4 SELECT stock_symbol, MAX(high) AS max_price
5 FROM daily_prices
6 GROUP BY stock_symbol
7 ) AS ax
8 LEFT JOIN
9 (
10 SELECT stock_symbol, MAX(dividends) AS max_divs
11 FROM dividends
12 GROUP BY stock_symbol
13 ) AS bx
14 ON (ax.stock_symbol=bx.stock_symbol)
15 WHERE ax.max_price>0;
```

Execute

Explain

Save as...

## Query Process Results (Status: SUCCEEDED)

Logs

Results

Filter columns...

bx.stock_symbol	ax.max_price	bx.max_divs
AAME	8.0	7.63
ABXA	9.19	9.09
AIPC	52.56	51.65