Project 1 **Stock Exchange Data Analysis**

DESCRIPTION

Objective: To use hive features for data engineering or analysis and sharing the actionable insights

Problem Statement:

New York stock exchange data of seven years, between 2010 to 2016, is captured for 500+ listed companies. The data set comprises of intra-day prices and volume traded for each listed company. The data serves both for machine learning and exploratory analysis projects, to automate the trading process and to predict the next trading-day winners or losers.. The scope of this project is limited to exploratory data analysis.

Domain: BFSI

Analysis to be done: Exploratory analysis to understand how MoM or YoY companies from different sectors or industries and states have progressed in a period of 7 years

Content: This data set contains prices.csv and securities.csv files having the following features:

Prices.csv:

1. Date: Trading date
2. Symbol: Ticker code or listed company code on NY stock exchange
3. Open: Intra-day opening price for each listed company
4. Close: Intra-day closing price for each listed company
5. Low: Intra-day lowest price for each listed company
6. High: Intra-day highest price for each listed company
7. Volume: Number of shares traded per day per company

Securities.csv:

1. Ticker_Symbol: Country to which the customer belongs
2. Security: Legal name of the listed company
3. Sector: Business vertical of the listed company
4. Sub_Industry: Business domain of the listed company within a Sector.
5. Headquarter: Headquarters address

Steps to perform:

1) Create a data pipeline using sqoop to pull the data from the table below from MYSQL server into Hive.
    a. MYSQL DATABASE NAME: BDHS_PROJECT
        i. Stock_prices
        ii. Stock_companies

Check the TABLE description: STOCK_PRICES

Column Name	**Datatype**
Trading_date	Date
Symbol	String
Open	double
Close	double
Low	double
High	double
Volume	int

TABLE: STOCK_COMPANIES

Column Name	**Datatype**
Symbol	String

Company_name	String
Sector	String
Sub_industry	String
Headquarter	String

2) Create a new hive table with the following fields by joining the above two hive tables. Please use appropriate Hive built-in functions for columns (a,b,e and h to l).

- Trading_year: Should contain YYYY for each record
- Trading_month: Should contain MM or MMM for each record
- Symbol: Ticker code
- CompanyName: Legal name of the listed company
- State: State to be extracted from headquarters value.
- Sector: Business vertical of the listed company
- Sub_Industry: Business domain of the listed company within a sector
- Open: Average of intra-day opening price by month and year for each listed company
- Close: Average of intra-day closing price by month and year for each listed company
- Low: Average of intra-day lowest price by month and year for each listed company
- High: Average of intra-day highest price by month and year for each listed company
- Volume: Average of number of shares traded by month and year for each listed company

Project Resolution

Load mysql data tables into Hive using Sqoop.

Load the STOCK_COMPANIES table

```
sqoop import --connect jdbc:mysql://ip-10-0-1-10.ec2.internal/BDHS_PROJECT --username labuser --password
simplilearn --table STOCK_COMPANIES  --hive-import --hive-table sg_TestHive.stock_companies -m 1;
```

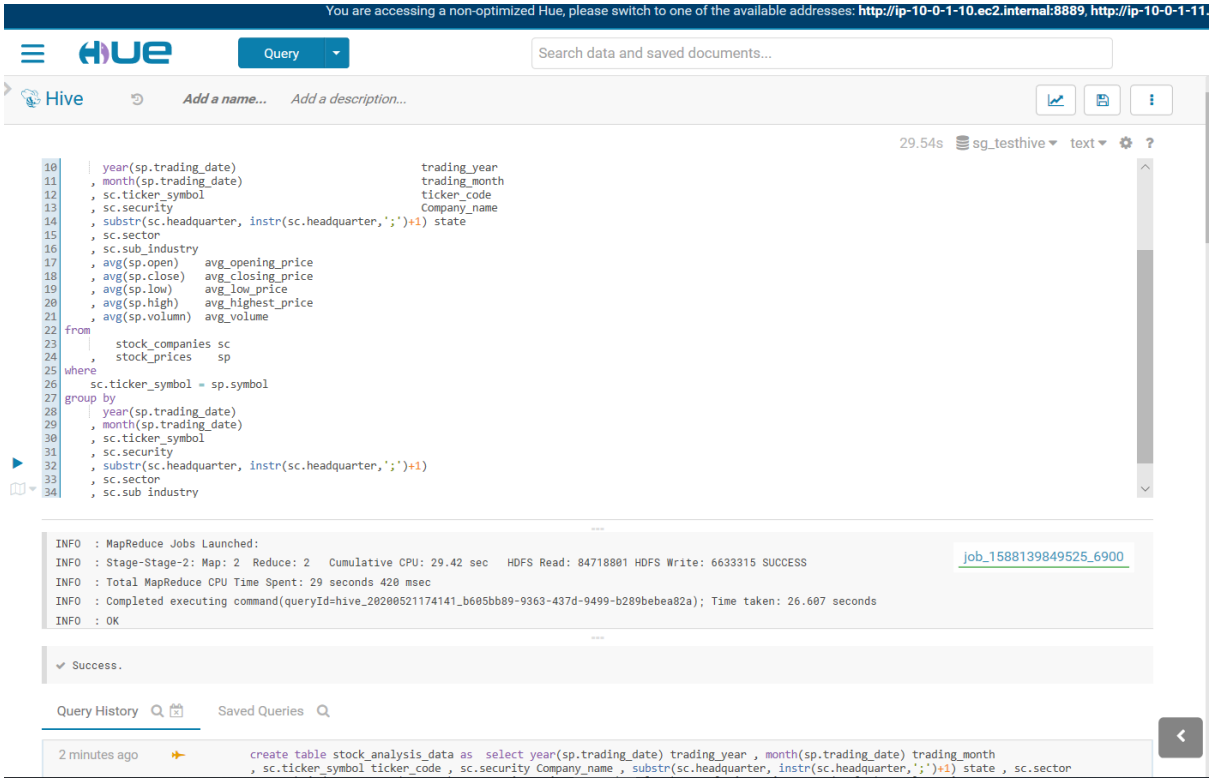
Load the STOCK_PRICES table

```
sqoop import --connect jdbc:mysql://ip-10-0-1-10.ec2.internal/BDHS_PROJECT --username labuser --password
simplilearn --table STOCK_PRICES  --hive-import --hive-table sg_TestHive.stock_prices -m 1;
```

Direct option could be used however this ran quickly.

Create new Hive table with the joined data of the above 2 tables
--Open: Average of intra-day opening price by month and year for each listed company
--Close: Average of intra-day closing price by month and year for each listed company
--Low: Average of intra-day lowest price by month and year for each listed company
--High: Average of intra-day highest price by month and year for each listed company
--Volume: Average of number of shares traded by month and year for each listed company

```
create table stock_analysis_data
as
SELECT
     year(sp.trading_date)                    trading_year
   , month(sp.trading_date)                     trading_month
   , sc.ticker_symbolticker_code
   , sc.securityCompany_name
   , substr(sc.headquarter, instr(sc.headquarter,';')+1) state
   , sc.sector
   , sc.sub_industry
   , avg(sp.open)    avg_opening_price
   , avg(sp.close)   avg_closing_price
   , avg(sp.low)   avg_low_price
   , avg(sp.high)    avg_highest_price
   , avg(sp.volumn)  avg_volume
from
stock_companiessc
,   stock_pricessp
where
sc.ticker_symbol = sp.symbol
group by
     year(sp.trading_date)
   , month(sp.trading_date)
   , sc.ticker_symbol
   , sc.security
   , substr(sc.headquarter, instr(sc.headquarter,';')+1)
   , sc.sector
   , sc.sub_industry
;
```

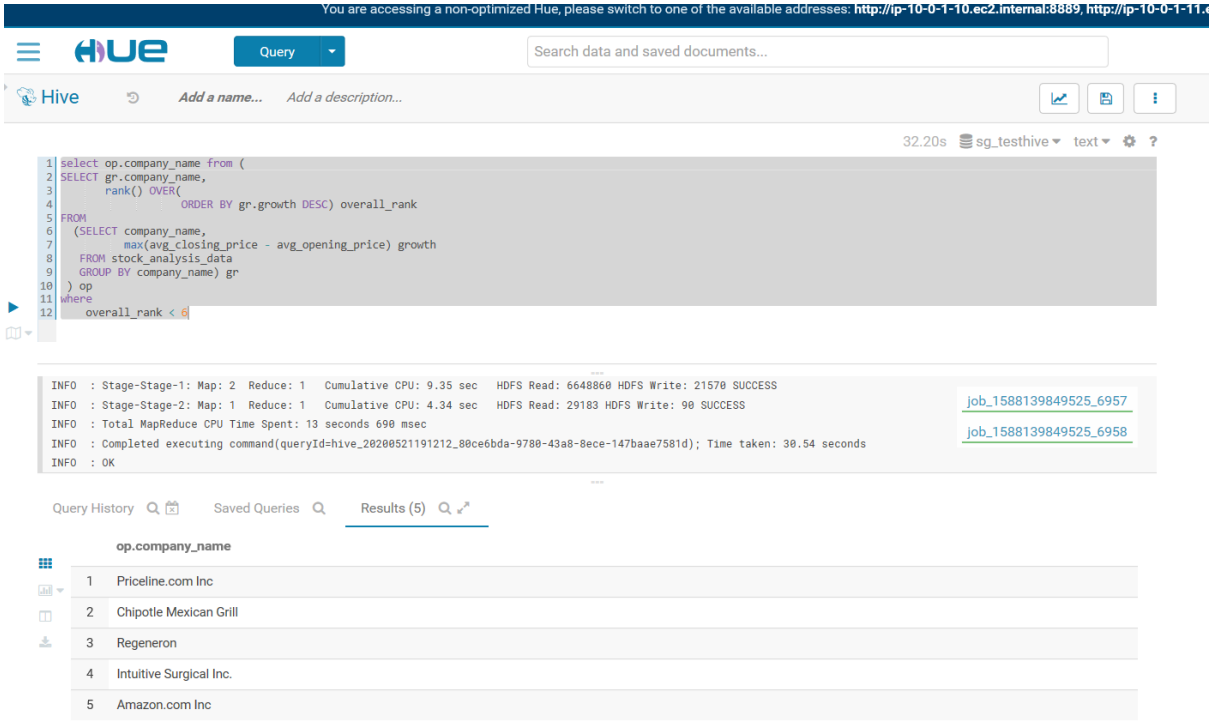


Analysis on the loaded data

The top five companies that are good for investment

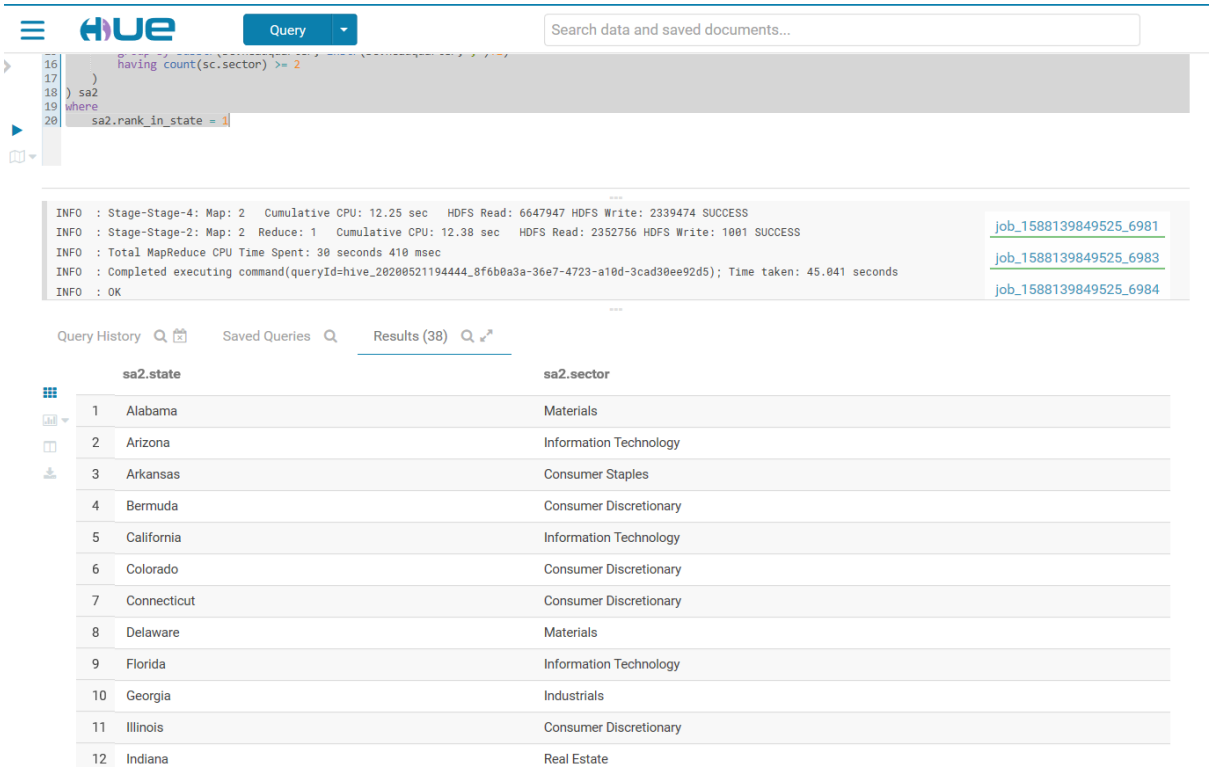
The rational is that the companies with highest average growth i.e. avg. closing price minus avg. opening price, are the ones which are good for investment

```
SELECT op.company_name from (
SELECT gr.company_name,
rank() OVER(
            ORDER BY gr.growth DESC) overall_rank
FROM
 (SELECT company_name,
max(avg_closing_price - avg_opening_price) growth
  FROM stock_analysis_data
  GROUP BY company_name) gr
 ) op
where
overall_rank< 6
```



The best-growing industry by each state, having at least two or more industries mapped.

```
select sa2.state, sa2.sector
from
(
   select sa1.state, sa1.sector,  rank() over (partition by sa1.state order by (sa1.avg_closing_price -
sa1.avg_opening_price)) rank_in_state
   from
stock_analysis_data sa1
   where sa1.state in
   (
      select substr(sc.headquarter, instr(sc.headquarter,';')+1) state
      from
stock_companiessc
      group by substr(sc.headquarter, instr(sc.headquarter,';')+1)
      having count(sc.sector) >= 2
   )
) sa2
where
   sa2.rank_in_state = 1
```



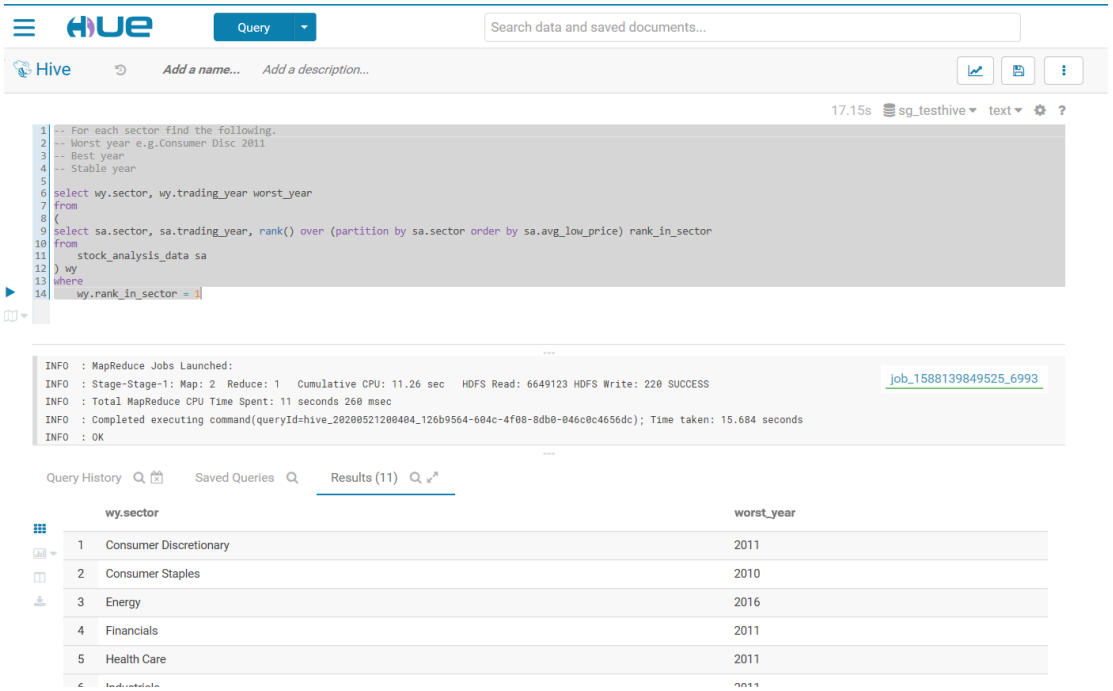
	sa2.state	sa2.sector
1	Alabama	Materials
2	Arizona	Information Technology
3	Arkansas	Consumer Staples
4	Bermuda	Consumer Discretionary
5	California	Information Technology
6	Colorado	Consumer Discretionary
7	Connecticut	Consumer Discretionary
8	Delaware	Materials
9	Florida	Information Technology
10	Georgia	Industrials
11	Illinois	Consumer Discretionary
12	Indiana	Real Estate

For each sector find the following.

Worst year

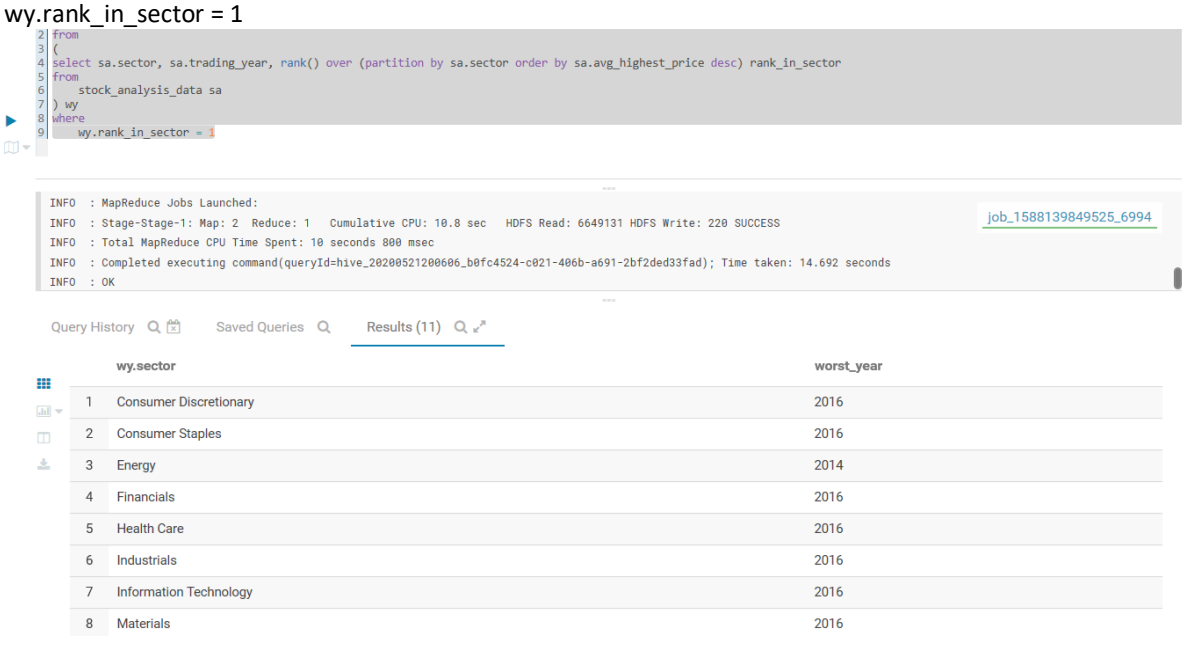
Got the year with the minimum average low price for a given sector

```
select wy.sector, wy.trading_yearworst_year
from
(
select sa.sector, sa.trading_year, rank() over (partition by sa.sector order by sa.avg_low_price) rank_in_sector
from
stock_analysis_datasa
) wy
where
wy.rank_in_sector = 1
```



Best year

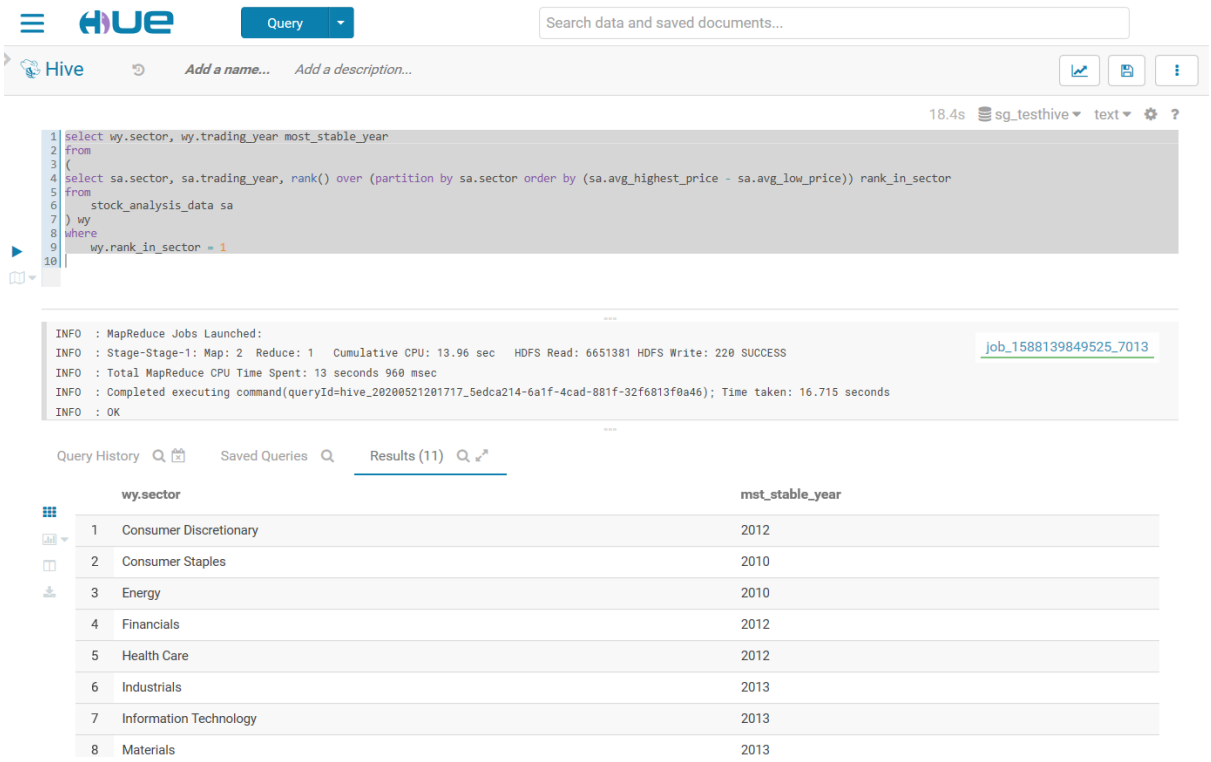
Got the year with the maximum average high price for a given sector
select wy.sector, wy.trading_yearbest_year
from
(
select sa.sector, sa.trading_year, rank() over (partition by sa.sector order by sa.avg_highest_pricedesc) rank_in_sector
from
stock_analysis_datasa
) wy
where
wy.rank_in_sector = 1



Stable year

An year with minimum difference between the average high price and average low price would be called as a stable year

```
select wy.sector, wy.trading_yearmost_stable_year
from
(
select sa.sector, sa.trading_year, rank() over (partition by sa.sector order by (sa.avg_highest_price - sa.avg_low_price)) rank_in_sector
from
stock_analysis_datasa
) wy
where
wy.rank_in_sector = 1
```



	wy.sector	mst_stable_year
1	Consumer Discretionary	2012
2	Consumer Staples	2010
3	Energy	2010
4	Financials	2012
5	Health Care	2012
6	Industrials	2013
7	Information Technology	2013
8	Materials	2013

<End of Project>