Candidate Name: Rahul Ramachandran

Please find the step by step procedure followed to create the table, load the data and exploratory analysis.

- **Connect to Mysql and see whether the tables exist in the mentioned database**
  - Database Exist

```
mysql> show databases like '%BDHS_PROJECT';
+--------------------------+
| Database (%BDHS_PROJECT) |
+--------------------------+
| BDHS_PROJECT             |
| HIVE_BDHS_PROJECT        |
| K214_BDHS_PROJECT        |
| SBK_BDHS_PROJECT         |
| VinayTS_BDHS_PROJECT     |
| bhavani_BDHS_PROJECT     |
| harshit_BDHS_PROJECT     |
| koushik_BDHS_PROJECT     |
| laxman_BDHS_PROJECT      |
+--------------------------+
9 rows in set (0.01 sec)
```

  - Table exists

```
mysql>
mysql> show tables;
+-------------------------+
| Tables_in_BDHS_PROJECT |
+-------------------------+
| STOCK_COMPANIES         |
| STOCK_COMPANIES_ISH     |
| STOCK_COMPANIES_MAH     |
| STOCK_PRICES            |
| STOCK_PRICES_MAH        |
| STOK_PRICES             |
| Stock_companies         |
| Stock_price             |
| Stock_prices            |
| comapny                 |
| comapny1                |
| comapny2                |
```

- **Import the data from Mysql to Hadoop using sqoop**
  - Import STOCK_PRICES

sqoop import --connect jdbc:mysql://localhost:3306/BDHS_PROJECT --username labuser --password simplilearn --table STOCK_PRICES -m 1 --fields-terminated-by ','

```
                GC time elapsed (ms)=106
                CPU time spent (ms)=9030
                Physical memory (bytes) snapshot=465690624
                Virtual memory (bytes) snapshot=2926596096
                Total committed heap usage (bytes)=646971392
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=52156430
20/08/06 09:00:01 INFO mapreduce.ImportJobBase: Transferred 49.7402 MB in 16.0969 seconds (3.0901 MB/sec)
20/08/06 09:00:01 INFO mapreduce.ImportJobBase: Retrieved 851264 records.
[rahulsbnt_gmail@ip-10-0-1-10 ~]$
```

- File generated in Destination folder

```
[rahulsbnt_gmail@ip-10-0-1-10 ~]$ hadoop fs -ls /user/rahulsbnt_gmail/STOCK_PRICES
Found 2 items
-rw-r--r--   2 rahulsbnt_gmail rahulsbnt_gmail          0 2020-08-06 08:59 /user/rahulsbnt_gmail/STOCK_PRICES/_SUCCESS
-rw-r--r--   2 rahulsbnt_gmail rahulsbnt_gmail   52156430 2020-08-06 08:59 /user/rahulsbnt_gmail/STOCK_PRICES/part-m-00000
[rahulsbnt_gmail@ip-10-0-1-10 ~]$
```

- Import STOCK_COMPANIES

sqoop import --connect jdbc:mysql://localhost:3306/BDHS_PROJECT --username labuser --password simplilearn --table STOCK_COMPANIES -m 1 --fields-terminated-by ','

```
                Physical memory (bytes) snapshot=365285376
                Virtual memory (bytes) snapshot=2912108544
                Total committed heap usage (bytes)=628097024
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=80004
20/08/06 09:34:00 INFO mapreduce.ImportJobBase: Transferred 78.1289 KB in 13.081 seconds (5.9727 KB/sec)
20/08/06 09:34:00 INFO mapreduce.ImportJobBase: Retrieved 1010 records.
[rahulsbnt_gmail@ip-10-0-1-10 ~]$
```

- File generated in Destination folder

```
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-07-10 07:48 /user/rahulsbnt_gmail/Jar_results
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-07-15 06:38 /user/rahulsbnt_gmail/MapReduceTest
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-08-06 08:54 /user/rahulsbnt_gmail/Project1
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-08-06 09:33 /user/rahulsbnt_gmail/STOCK_COMPANIES
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-08-06 08:59 /user/rahulsbnt_gmail/STOCK_PRICES
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-07-10 06:04 /user/rahulsbnt_gmail/TestCmd
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-07-15 09:51 /user/rahulsbnt_gmail/_sqoop
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-08-06 08:44 /user/rahulsbnt_gmail/employee
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-07-15 07:12 /user/rahulsbnt_gmail/sqoop
[rahulsbnt_gmail@ip-10-0-1-10 ~]$ hadoop fs -ls /user/rahulsbnt_gmail/STOCK_COMPANIES
Found 2 items
-rw-r--r--   2 rahulsbnt_gmail rahulsbnt_gmail          0 2020-08-06 09:33 /user/rahulsbnt_gmail/STOCK_COMPANIES/_SUCCESS
-rw-r--r--   2 rahulsbnt_gmail rahulsbnt_gmail      80004 2020-08-06 09:33 /user/rahulsbnt_gmail/STOCK_COMPANIES/part-m-00000
[rahulsbnt_gmail@ip-10-0-1-10 ~]$
```

- Copy the files to Project Folder.

```
[rahulsbnt_gmail@ip-10-0-1-10 ~]$ hadoop fs -ls /user/rahulsbnt_gmail/Project1/
Found 2 items
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-08-07 03:56 /user/rahulsbnt_gmail/Project1/company
drwxr-xr-x   - rahulsbnt_gmail rahulsbnt_gmail          0 2020-08-07 02:09 /user/rahulsbnt_gmail/Project1/price
[rahulsbnt_gmail@ip-10-0-1-10 ~]$
```

- Created hive DB

```
hive> create database rahulsbnt_Project;
OK
Time taken: 0.028 seconds
hive> use rahulsbnt_project
    > ;
OK
Time taken: 0.012 seconds
hive> show tables
    > ;
OK
Time taken: 0.08 seconds
hive>
```

- Create an EXTERNAL table for PRICE

create external table stock_price (DATE string,SYMBOL string,open double,close double,low double,high double, volume double) row format delimited fields terminated by ',' location '/user/rahulsbnt_gmail/Project1/price'

```
hive> create external table stock_price (DATE string,SYMBOL string,open double,close double,low double,volume double) row format deli
mited fields terminated  by ',' location '/user/rahulsbnt_gmail/Project1/price';
OK
Time taken: 0.06 seconds
hive>
```

- Selected first two rows from both the tables

create external table stock_companies(SYMBOL string,security string,sector string,sub_industry string, headquarter string) row format delimited fields terminated by ',' location '/user/rahulsbnt_gmail/Project1/company'

```
hive> select * from stock_companies  limit 2;
OK
MMM     3M Company      Industrials     Industrial Conglomerates        St. Paul; Minnesota
ABT     Abbott Laboratories     Health Care     Health Care Equipment   North Chicago; Illinois
Time taken: 0.053 seconds, Fetched: 2 row(s)
hive> select * from stock_price limit 2;
OK
2016-01-05      WLTW    123.43  125.839996      122.309998      126.25
2016-01-06      WLTW    125.239998      119.980003      119.940002      125.540001
Time taken: 0.057 seconds, Fetched: 2 row(s)
hive>
```

- Created a MANAGED table "Stock_Details" from price and companies merged columns

CREATE TABLE sd as SELECT p.Trading_year,p.Trading_month,cast(p.Month_sort as int) as Month_sort,p.symbol,c.security as company_name, split(c.headquarter,'\;')[1] as state, c.sector,c.sub_industry,p.open,p.close,p.low,p.high,p.volume FROM stock_companies c right outer join (SELECT date_format(p.date,'YYYY') as Trading_year,date_format(p.date,'MMM') as Trading_month,date_format(date,'MM') as Month_sort, symbol, AVG(CAST(open AS DECIMAL(9,2))) open , AVG(CAST(close AS DECIMAL(9,2))) as close, AVG(CAST(low AS DECIMAL(9,2))) as low, AVG(CAST(high AS DECIMAL(9,2))) as high , AVG(CAST(volume AS DECIMAL(9,2))) as volume from stock_price p group by date_format(p.date,'YYYY'),date_format(p.date,'MMM'),date_format(date,'MM'),symbol) p on (p.symbol = c.symbol);

```
hive> describe stock_details;
OK
trading_year            string
trading_month           string
month_sort              int
symbol                  string
company_name            string
state                   string
sector                  string
sub_industry            string
open                    double
close                   double
low                     double
high                    double
volume                  double
Time taken: 0.063 seconds, Fetched: 13 row(s)
hive> select * from stock_details limit 5;
OK
2010    Apr     4       MMM     3M Company       Minnesota      Industrials     Industrial Conglomerates        85.24   85.33   84.66
   85.88   4590809.52
2010    Aug     8       MMM     3M Company       Minnesota      Industrials     Industrial Conglomerates        83.66   83.63   82.84
   84.34   3396413.64
2010    Dec     12      MMM     3M Company       Minnesota      Industrials     Industrial Conglomerates        85.83   85.91   85.23
   86.52   4048352.94
2010    Feb     2       MMM     3M Company       Minnesota      Industrials     Industrial Conglomerates        79.91   79.9    79.1
   80.46   3948442.11
2010    Jan     1       MMM     3M Company       Minnesota      Industrials     Industrial Conglomerates        83.2    83.0    82.28
   83.74   3958321.05
Time taken: 0.05 seconds, Fetched: 5 row(s)
hive>
```

- # Exploratory Data Analysis

- **Find the top five companies that are good for investment**

  Logic: Took 5 top companies who sold max number for shares between these years. Prepared script and the result screenshot provided below.

  select company_name from (select a. from (select company_name, round(sum(volume),2) vol from sd group by company_name) a where a.company_name is not null order by a.vol desc limit 5) b*

```
Ended Job = job_1594878743366_5931
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 6.87 sec   HDFS Read: 5186332 HDFS Write: 20872 SUCCESS
Stage-Stage-2: Map: 1  Reduce: 1   Cumulative CPU: 3.16 sec   HDFS Read: 25946 HDFS Write: 90 SUCCESS
Total MapReduce CPU Time Spent: 10 seconds 30 msec
OK
Delta Air Lines
Procter & Gamble
Johnson & Johnson
Merck & Co.
Charles Schwab Corporation
Time taken: 31.815 seconds, Fetched: 5 row(s)
hive>
```

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

  As mentioned above, best growing industry can be find using the number of shares sold in the past. Please see the below script and the resulted screenshot attached.

```
SELECT state,sector FROM (
        SELECT state,sector, vol,s_count,RANK() OVER (PARTITION BY state ORDER BY vol DESC)
top_sector FROM (
                SELECT state,sector,SUM(volume) vol,
                COUNT(sector)  OVER (PARTITION BY state) s_count
                FROM stock_details GROUP BY state,sector
        ) sd WHERE s_count > 1
) main  WHERE main.top_sector = 1 ;
```

```
Stage-Stage-3: Map: 1  Reduce: 1   Cumulative CPU: 5.36 sec
Total MapReduce CPU Time Spent: 18 seconds 490 msec
OK
 Alabama        Financials
 Arizona        Information Technology
 Arkansas       Consumer Staples
 Bermuda        Financials
 California     Information Technology
 Colorado       Materials
 Connecticut    Financials
 Delaware       Materials
 Florida        Consumer Discretionary
 Georgia        Consumer Discretionary
 Illinois       Industrials
 Indiana        Health Care
 Iowa   Financials
 Ireland        Health Care
 Kentucky       Consumer Discretionary
 Louisiana      Telecommunications Services
 Maryland       Consumer Discretionary
 Massachusetts  Consumer Discretionary
 Michigan       Consumer Discretionary
 Minnesota      Consumer Discretionary
 Missouri       Health Care
 Nebraska       Industrials
 Netherlands    Health Care
 New Jersey     Health Care
 New York       Financials
 North Carolina Consumer Discretionary
 Ohio   Financials
 Oregon Consumer Discretionary
 Pennsylvania   Health Care
 Rhode Island   Consumer Staples
 Switzerland    Energy
 Tennessee      Materials
 Texas  Energy
 Virginia       Consumer Discretionary
 Washington     Consumer Discretionary
 Wisconsin      Consumer Discretionary
Time taken: 49.817 seconds, Fetched: 36 row(s)
hive>
```

- **For each sector find the Worst, best and stable year**

  Logic: For these analyses, we need to find out the average share growth for each sector. This can be achieved by using below formula.

  Growth Percentage = ((**close-open)/open) *100**

  I took three queries for each scenario (best, worst and stable) and joined together to achieve the result. Please see the below result screenshot and the script for the reference.

```
select w.sector sector,w.trading_year worst_year,w.growth as Worst_Growth,b.trading_year
as Best_Year,b.growth as Best_Growth,s.trading_year as Stable_Year from (
        select sector,trading_year,growth,growth_rank from (SELECT
sector,trading_year,round(((sum(close)-sum(open))/sum(open))*100,2)  growth,dense_rank()
over (partition by sector order by round(((sum(close)-
sum(open))/sum(open))*100,2),trading_year asc) growth_rank frOM stock_details  GROUP BY
sector,trading_year) worst where growth_rank = 1) w
left join
        (select sector,trading_year,growth,growth_rank from (SELECT
sector,trading_year,round(((sum(close)-sum(open))/sum(open))*100,2)  growth,dense_rank()
over (partition by sector order by round(((sum(close)-
sum(open))/sum(open))*100,2)desc,trading_year desc) growth_rank frOM stock_details
GROUP BY sector,trading_year) best where growth_rank = 1) b
on (b.sector = w.sector)
left join
        (select sector,trading_year,growth,growth_rank from (SELECT
sector,trading_year,abs(round(((sum(close)-sum(open))/sum(open))*100,2))  growth,
dense_rank() over (partition by sector order by abs(round(((sum(close)-
sum(open))/sum(open))*100,2)),trading_year desc) growth_rank frOM stock_details  GROUP BY
sector,trading_year) stable where growth_rank = 1) s
on (w.sector = s.sector) ;
```

```
Total MapReduce CPU Time Spent: 32 seconds 940 msec
OK
sector  worst_year      worst_growth    best_year       best_growth     stable_year
NULL    2014    -0.13   NULL    NULL    NULL
Consumer Discretionary  2015    -0.05   2010    0.09    2014
Consumer Staples        2015    0.01    2013    0.07    2015
Energy  2011    -0.03   2016    0.07    2015
Financials      2011    -0.07   2013    0.09    2014
Health Care     2015    -0.02   2013    0.07    2016
Industrials     2011    -0.03   2013    0.09    2015
Information Technology  2011    -0.03   2013    0.07    2015
Materials       2011    -0.06   2012    0.07    2015
Real Estate     2013    -0.02   2014    0.08    2015
Telecommunications Services     2011    -0.13   2016    0.01    2016
Utilities       2012    -0.02   2016    0.09    2010
Time taken: 109.369 seconds, Fetched: 12 row(s)
hive>
```