Summary

of

Assignment - Stock Market Analysis

By

Rachit Dev

Prerequisites for making the assignment:

1. Import all the required .csv files as a table using the import table wizard of mysql and choose only the date column (text) and Close Price column for further analysis.
2. We now create the database using query:

```
create database if not exists sql_assignment;
```

3. We now use the above-mentioned database using query:

```
use sql_assignment;
```

4. Now we convert the Date column which is in text format in Date format for all the tables using query. The below mentioned query is used for one table, similarly we do it for other tables too.

```
alter table bajaj add column new_date DATE;
update bajaj
set new_date = str_to_date(`Date`, '%d-%M-%Y');
ALTER TABLE bajaj DROP Date;
```

Steps to create the desired results using queries:

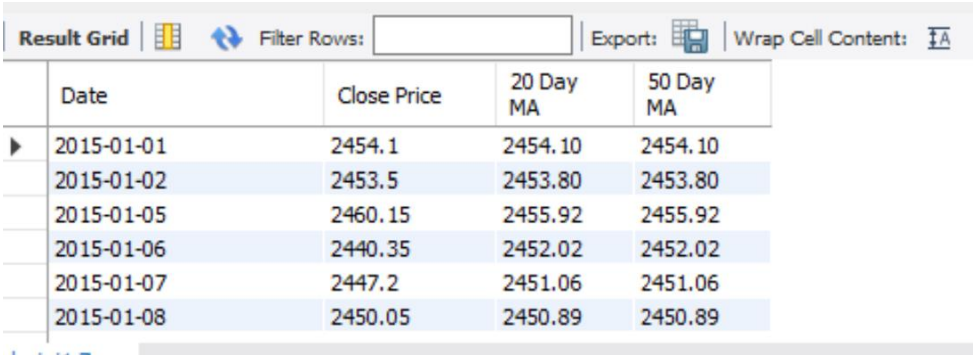
1. Our first step is to create tables for all the stocks whose data has been given to us in the below mentioned format

Date	Close Price	20 Day MA	50 Day MA

```sql
-- Lets create a window function for MA20 and MA50 and save them in table bajaj1
CREATE TABLE bajaj1
SELECT new_date as `Date`, `Close Price`,
       ROUND(AVG(close_price) OVER (ORDER BY new_date ASC ROWS 19 PRECEDING), 2) AS `20 Day MA`,
       ROUND(AVG(close_price) OVER (ORDER BY new_date ASC ROWS 49 PRECEDING), 2) AS `50 Day MA`
FROM   bajaj;

-- Test the above window function using the below commented query.
-- select *
-- from bajaj1;
```

The output will be like below mentioned:



Date	Close Price	20 Day MA	50 Day MA
2015-01-01	2454.1	2454.10	2454.10
2015-01-02	2453.5	2453.80	2453.80
2015-01-05	2460.15	2455.92	2455.92
2015-01-06	2440.35	2452.02	2452.02
2015-01-07	2447.2	2451.06	2451.06
2015-01-08	2450.05	2450.89	2450.89

Similarly we perform for all the other tables.

2. To create the master table and it should appear like below mentioned:

Date	Bajaj	TCS	TVS	Infosys	Eicher	Hero

```
-- Let's create the window function for creating the master table.
CREATE TABLE `master`
SELECT bajaj1.Date, bajaj1.`Close Price` AS Bajaj, eicher1.`Close Price` AS Eicher,
        hero1.`Close Price` AS Hero, infosys1.`Close Price` AS Infosys,
        tcs1.`Close Price` AS TCS, tvs1.`Close Price` AS TVS
FROM (((((bajaj1
INNER JOIN eicher1 ON bajaj1.`Date` = eicher1.`Date`)
INNER JOIN hero1 ON bajaj1.`Date` = hero1.`Date`)
INNER JOIN infosys1 ON bajaj1.`Date` = infosys1.`Date`)
INNER JOIN tcs1 ON bajaj1.`Date` = tcs1.`Date`)
INNER JOIN tvs1 ON bajaj1.`Date` = tvs1.`Date`);

-- Test the above window function using the below commented query.
-- SELECT *
-- FROM `master`
```

The output is like the below mentioned table:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Date	Bajaj	Eicher	Hero	Infosys	TCS	TVS
2015-01-01	2454.1	15239.15	3107.3	1975.8	2548.2	276.85
2015-01-02	2453.5	15065.55	3097.75	2013.45	2570.85	267.8
2015-01-05	2460.15	15133.55	3118.65	1996.6	2537.9	271.2
2015-01-06	2440.35	14753.1	3011.8	1956.85	2446.55	268
2015-01-07	2447.2	14945.55	2977.1	1964.8	2416.8	278.55
2015-01-08	2450.05	14877.8	2984.45	1974.55	2443.85	295.8
2015-01-09	2381.5	14727.1	2993.15	2073.6	2512.2	296.7

master 8

3. Creating bajaj2 as below mentioned format:

Date	Close Price	Signal

```
create table bajaj2 as
select `Date`, `Close Price`,
case when first_value(short_term_greater) over w = nth_value(short_term_greater,2) over w then 'Hold'
     when nth_value(short_term_greater,2) over w = 'Y' then 'Buy'
     when nth_value(short_term_greater,2) over w = 'N' then 'Sell'
     else 'Hold'
     end
As `Signal`
from
(
select `Date`, `Close Price`,
        if (`20 Day MA` > `50 Day MA`, 'Y', 'N') short_term_greater
    from bajaj1
) temp_table

window w as (order by `Date` rows between 1 preceding and 0 following);
```

The table bajaj2 is below mentioned and we do it for all the stocks.

Result Grid | Filter Rows:

Date	Close Price	Signal
2015-12-09	2390.45	Hold
2015-12-10	2441.45	Hold
2015-12-11	2411.9	Sell
2015-12-14	2409.75	Hold
2015-12-15	2435.45	Hold
2015-12-16	2451.65	Hold

bajaj2 9

4. Create a User defined function:

signalOfDay - Routine

Name: signalOfDay

DDL:

```
1  CREATE DEFINER=`root`@`localhost`
2  FUNCTION `signalOfDay`(input_date DATE)
3  RETURNS varchar(4) CHARSET utf8mb4
4      DETERMINISTIC
5  begin
6  declare signal_value varchar(4);
7  SELECT
8      `Signal`
9  INTO signal_value FROM
10     bajaj2
11 WHERE
12     `Date` = input_date;
13 return signal_value;
14 end
```

The output of the above-mentioned function comes by using the below mentioned query:

```
285 -- So we call the above mentioned function from the stored function signalOfDay in the below mentioned query.
286 Select `signalOfDay`('2015-05-18');
287 -- The above query returned "Buy".
288
289 -- End of Part 4 of the assignment and assignment completed.
290
291
292
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

`signalOfDay`('2015-05-18')
Buy

Result Grid