RDBMS Assignment 2

Avirup Das Roll No: MDS202013
Shiuli Subhra Ghosh Roll No: MDS202035
Suman Roy Roll No: MDS202041

M.Sc. in Data Science

Date : 10.02.2021

Chennai Mathematical Institute

Chennai, India

1. **Query to count the total number of rows in the ‘stock’ table:**

   ```
   select count(*) from stock;
   ```

   Output:

count(*)
2463

---

2. **Query for finding highest overall (across multiple days) gain and highest loss per company ever achieved, assuming that the shares were bought on 02-11-2017:**

   ```
   select Symbol, max(d) as Maximum_Gain, -min(d) as Maximum_Loss from
   (select Symbol, ((ClosePrice-a)) as d from stock,
   (select Symbol as n, ClosePrice as a from stock where Dates=”2017-11-02”) as b
   where Symbol=n) as c group by Symbol;
   ```

   Output:

Symbol	Maximum_Gain	Maximum_Loss
APOLLOHOSP	503.95000000000005	105.74999999999989
ICICIBANK	160.75	57.39999999999998
INDIGO	641.7	523.5
RELIANCE	532.4999999999999	71.35000000000002
SUNTV	224.4000000000001	439.65

---

3. **Query 1**:

   ```
   select symbol, count(distinct Sub.Dates) as ”Number of Days”
   from (select * from stock where ((ClosePrice-PrevClose)/PrevClose)*100>3)
   as Sub group by symbol ;
   ```

   Query 2:

   ```
   select symbol, count(symbol) as ”Number of Days” from stock
   where (ClosePrice - PrevClose)/PrevClose * 100 > 3
   group by symbol;
   ```

   For both the queries, output is:

Symbol	Number of Days
APOLLOHOSP	38
ICICIBANK	34
INDIGO	51
RELIANCE	19
SUNTV	34

---

4. **Query for showing the list of the dates in which all five stocks closed higher than the previous close:**

```
select Dates as All_Five_Stock_Closed_Higher
from stock where ClosePrice > PrevClose group by Dates
having count(Symbol)=5;
```

The output table would have 45 rows:

All_Five_Stock_Closed_Higher
2017-11-09
2017-11-16
2017-12-05
⋮

Query for showing the list of the dates in which all five stocks closed lower than the previous close:

```
select Dates as All_Five_Stock_Closed_Lower
from stock where ClosePrice < PrevClose group by Dates
having count(Symbol)=5;
```

The output table would have 39 rows:

All_Five_Stock_Closed_Lower
2017-11-07
2017-11-08
2017-11-13
⋮

Query for counting the number of days in which all the five stocks closed higher than previous close as well as the number of days in which all five stocks closed lower than previous close:

```
select
((select count(distinct dates) from stock)
– (select count(distinct dates) from stock
where PrevClose>ClosePrice)) as ”All Five stocks closed higher”,
((select count(distinct dates) from stock)
– (select count(distinct dates) from stock
where PrevClose<ClosePrice)) as ”All Five stocks closed Lower” ;
```

The output table is as follows:

All Five stocks closed higher	All Five stocks closed Lower
45	39

5. Assuming we have invested Rs. 5000 on one of the five stocks during 1st week of January 2018 (between ’2018-01-01’ and ’2018-01-05’).

   Assumption : For Max profit we have taken the Lowest price on the Investment Date and Highest Price on the Withdrawal Date.

   The Query for the achievement of the best gain during the last week of the January 2018 (between ’2018-01-22’ and ’2018-01-31’) :

```
select der.symbol, der.InvestmentDate, der.WithdrawalDate,
der.LowPrice as BuyPrice ,der.HighPrice as SellPrice,
(5000 - 5000 % der.LowPrice) / der.LowPrice as num_stock,
(5000 - 5000 % der.LowPrice) / der.LowPrice *
(der.HighPrice - der.LowPrice) as max_profit
from (select c.symbol, c.Dates as InvestmentDate,
c.LowPrice, p.dates as
WithdrawalDate, p.HighPrice ,
(p.HighPrice - c.LowPrice)/c.LowPrice*100 as percentage
from
(select * from stock where Dates between '2018-01-01' and '2018-01-05')
as c inner join
(select * from stock where Dates between '2018-01-22' and '2018-01-31')
as p on c.symbol = p.symbol) as der
order by der.percentage desc limit 1;
```

   The Output table is showing as:

Symbol	InvestmentDate	WithdrawalDate	BuyPrice	SellPrice	num_stock	max_profit
ICICIBANK	2018-01-02	2018-01-29	307.5	365.7	16	931.199

   The Query for the achievement of the best gain during the last week of 2018 (between ’2018-12-24’ and ’2018-12-31’) :

```
select der.symbol, der.InvestmentDate, der.WithdrawalDate,
der.LowPrice as BuyPrice ,der.HighPrice as SellPrice,
(5000 - 5000 % der.LowPrice) / der.LowPrice as num_stock,
(5000 - 5000 % der.LowPrice) / der.LowPrice *
(der.HighPrice - der.LowPrice) as max_profit
from (select c.symbol, c.Dates as InvestmentDate,
c.LowPrice, p.dates as
WithdrawalDate, p.HighPrice ,
(p.HighPrice - c.LowPrice)/c.LowPrice*100 as percentage
from
(select * from stock where Dates between '2018-01-01' and '2018-01-05')
as c inner join
(select * from stock where Dates between '2018-12-24' and '2018-12-31')
as p on c.symbol = p.symbol) as der
order by der.percentage desc limit 1;
```

   The Output table is showing as:

Symbol	InvestmentDate	WithdrawalDate	BuyPrice	SellPrice	num_stock	max_profit
RELIANCE	2018-01-02	2018-12-28	906.4	1135.25	5	1144.25