

## DS 203 E9

### Task 1-

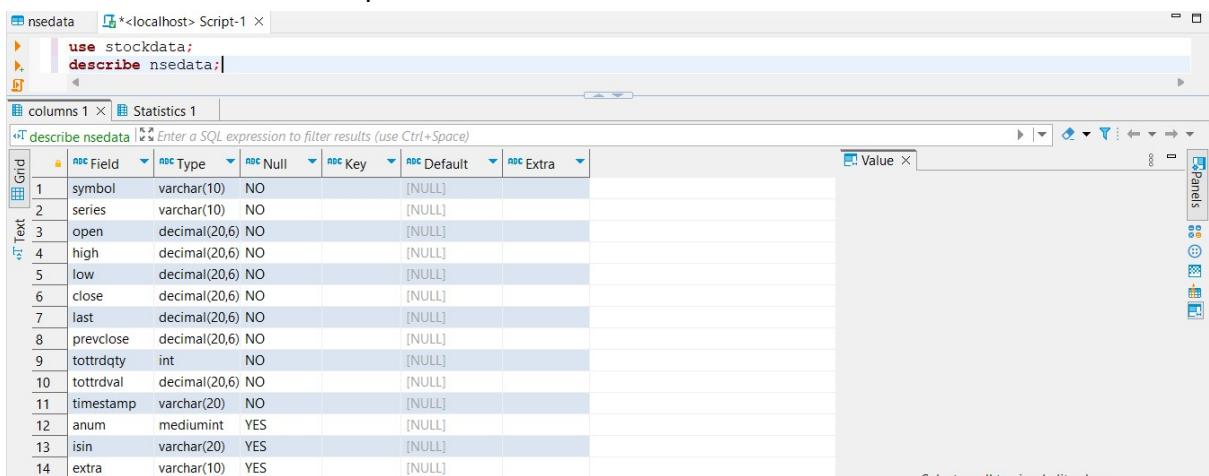
The table nsedata contains data of national stock exchange for various trading companies for a particular period of time. The data includes statistics like the dates of the days when a particular company participated in trade, opening and closing price of the day, highest and lowest price of the data,etc.

### Task 2 - Select the database **stockdata** using SQL



```
nsedata *localhost Script-1
use stockdata
```

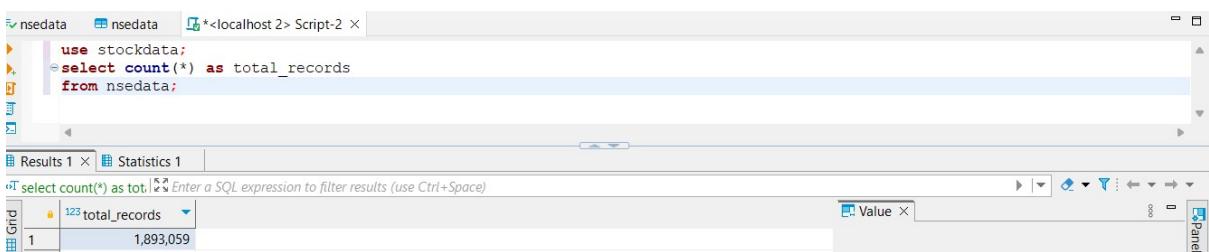
### Task 3 - Get a schema dump of the table **nsedata**



```
nsedata *localhost Script-1
use stockdata;
describe nsedata;
```

Field	Type	Null	Key	Default	Extra
symbol	varchar(10)	NO		[NULL]	
series	varchar(10)	NO		[NULL]	
open	decimal(20,6)	NO		[NULL]	
high	decimal(20,6)	NO		[NULL]	
low	decimal(20,6)	NO		[NULL]	
close	decimal(20,6)	NO		[NULL]	
last	decimal(20,6)	NO		[NULL]	
prevclose	decimal(20,6)	NO		[NULL]	
tottrdqty	int	NO		[NULL]	
tottrdval	decimal(20,6)	NO		[NULL]	
timestamp	varchar(20)	NO		[NULL]	
anum	mediumint	YES		[NULL]	
isin	varchar(20)	YES		[NULL]	
extra	varchar(10)	YES		[NULL]	

### Task 4 - Get a count of the total number of records in **nsedata**



```
nsedata *localhost 2 Script-2
use stockdata;
select count(*) as total_records
from nsedata;
```

total_records
1,893,059

### Task 5 - Get the total count of the records for the month “October 2012”



```
nsedata <none> Script-2 nsedata *localhost 3 Script-3
use stockdata;
select count(*) from nsedata where timestamp like '%Oct-2012'.
```

count(*)
33,244

### Task 6 - Repeat '4', but only for the stock with symbol "GEOMETRIC"

```
use stockdata;
select count(*) from nsedata where symbol = 'GEOMETRIC';
```

Results 1 × Statistics 1

select count(\*) from nsedata where symbol = 'GEOMETRIC'

123 count(*)	Value
1	1,237

### Task 7 - Repeat '6', but only display the first 10 records

```
use stockdata;
select * from nsedata where symbol = 'GEOMETRIC' limit 10;
```

nsedata 1 × Statistics 1

select\* from nsedata where symbol = 'GEOMETRIC'

Grid	abc symbol	abc series	123 open	123 high	123 low	123 close	123 last	123 prevclose	123 tottrdqty	123 tottrdval	abc timestamp	123 a
1	GEOMETRIC	EQ	62.35	64.5	61.4	63.25	63.25	61.3	82,246	5,179,345.65	01-APR-2011	000
2	GEOMETRIC	EQ	100.7	105.5	99.1	103.5	102.55	100.2	124,482	12,753,266.8	01-APR-2013	000
3	GEOMETRIC	EQ	116	121	116	120	120.2	115.55	644,060	77,015,430	01-APR-2014	000
4	GEOMETRIC	EQ	166.5	184.5	163	177.55	177.4	167.15	2,398,121	426,671,089.7	01-APR-2015	000
5	GEOMETRIC	EQ	49.8	50.1	49.05	49.9	50	48.9	55,376	2,765,041.05	01-AUG-2011	000
6	GEOMETRIC	EQ	94.4	94.65	90.4	91.8	91.95	94.2	887,542	82,336,516.15	01-AUG-2012	000
7	GEOMETRIC	EQ	69.45	70.05	63	64.1	63.3	69.45	319,336	20,718,691.55	01-AUG-2013	000
8	GEOMETRIC	EQ	141.2	144	139.5	140.2	140.5	142.6	291,911	41,292,337.35	01-AUG-2014	000
9	GEOMETRIC	EQ	73.3	73.6	71.35	72.35	72.4	72.15	130,567	9,479,917.2	01-DEC-2010	000
10	GEOMETRIC	EQ	45.9	48.9	45.5	47.2	47	45	124,440	5,906,044.7	01-DEC-2011	000

### Task 8 - Totally, how many records of "INFY" does the table contain?

```
select count(*) from nsedata where symbol = 'INFY';
```

Results 1 × Statistics 1

select count(\*) from nsedata where symbol = 'INFY'

123 count(*)	Value
1	1,023

### Task 9 - Get a listing of the first 10 records of "3IINFOTECH", but the listing should contain only the following columns: symbol, open, high, low, close, and timestamp

```
use stockdata;
select symbol,open,high,low,close,timestamp from nsedata where symbol = '3IINFOTECH' limit 10;
```

nsedata 1 ×

select symbol,open,h

Grid	abc symbol	123 open	123 high	123 low	123 close	abc timestamp
1	3IINFOTECH	43.75	45.3	43.75	44.9	01-APR-2011
2	3IINFOTECH	5.65	6.1	5.65	6.1	01-APR-2013
3	3IINFOTECH	7.85	7.9	7.45	7.65	01-APR-2014
4	3IINFOTECH	5.9	6.3	5.8	6.2	01-APR-2015
5	3IINFOTECH	41.6	42.45	40.2	40.45	01-AUG-2011
6	3IINFOTECH	10.8	10.8	10.5	10.8	01-AUG-2012
7	3IINFOTECH	3.95	4.15	3.85	4	01-AUG-2013
8	3IINFOTECH	8.75	9.1	8.6	8.65	01-AUG-2014
9	3IINFOTECH	55.9	59.4	55.55	58.35	01-DEC-2010
10	3IINFOTECH	20	20	18.5	18.65	01-DEC-2011

### Task 10 - Repeat '9', but this time use the results to create a temporary table t1

The screenshot shows the MySQL Workbench interface. In the top tab bar, there are two tabs: 'nsedata' and 'nsedata'. Below them is a script tab titled 'Script-3' containing the following SQL code:

```
use stockdata;
create table t1 as select symbol,open,high,low,close,timestamp from nsedata where symbol ='3IINFOOTECH' limit 10;
select * from t1;
```

Below the script is a results grid titled 't1' with the following data:

	symbol	open	high	low	close	timestamp
1	3IINFOOTECH	43.75	45.3	43.75	44.9	01-APR-2011
2	3IINFOOTECH	5.65	6.1	5.65	6.1	01-APR-2013
3	3IINFOOTECH	7.85	7.9	7.45	7.65	01-APR-2014
4	3IINFOOTECH	5.9	6.3	5.8	6.2	01-APR-2015
5	3IINFOOTECH	41.6	42.45	40.2	40.45	01-AUG-2011
6	3IINFOOTECH	10.8	10.8	10.5	10.8	01-AUG-2012
7	3IINFOOTECH	3.95	4.15	3.85	4	01-AUG-2013
8	3IINFOOTECH	8.75	9.1	8.6	8.65	01-AUG-2014
9	3IINFOOTECH	55.9	59.4	55.55	58.35	01-DEC-2010
10	3IINFOOTECH	20	20	18.5	18.65	01-DEC-2011

### Task 11 - Using t1 find out the following for the column close: max, min, mean, standard deviation and variance

The screenshot shows the MySQL Workbench interface. In the top tab bar, there are two tabs: 'nsedata' and 'nsedata'. Below them is a script tab titled 'Script-3' containing the following SQL code:

```
select
min(close)as min_value,
max(close)as max_value,
avg(close)as mean,
stdev(close)as standard_deviation,
variance(close)as var from t1;
```

Below the script is a results grid titled 'Results 1' with the following data:

	min_value	max_value	mean	standard_deviation	var
1	4	58.35	20.575	18.7432287773	351.308625

### Task 12 - How will you find out the value of the median?

The screenshot shows the MySQL Workbench interface. In the top tab bar, there are two tabs: 'nsedata' and 'nsedata'. Below them is a script tab titled 'Script-3' containing the following SQL code:

```
SET @rowindex := -1;
SELECT
AVG(d.close) as Median
FROM
(SELECT @rowindex:=@rowindex + 1 AS rowindex,
t1.close AS close
FROM t1)
ORDER BY t1.close) AS d
WHERE
d.rowindex IN (FLOOR(@rowindex / 2), CEIL(@rowindex / 2));
```

Below the script is a results grid titled 'Results 1' with the following data:

	Median
1	9.725

### Task 13 - Delete table t1

The screenshot shows the MySQL Workbench interface. In the top tab bar, there are two tabs: 'nsedata' and 'nsedata'. Below them is a script tab titled 'Script-3' containing the following SQL code:

```
drop table t1;
```

### Task 14 - Use nsedata. Using the GROUP BY functionality of SQL create a table t2 containing the average value of close for each and every symbol in the table. Hint: the table will have the columns: symbol, average

The screenshot shows the MySQL Workbench interface. In the top tab bar, there are two tabs: 'nsedata' and 'nsedata'. Below them is a script tab titled 'Script-3' containing the following SQL code:

```
create table t2 as
select symbol, avg(close) as average
from nsedata group by symbol;
```

**Task 15** - Create a table t3 such that it contains the following columns: symbol, open, close, "average of open and close". Fill up this table for the company GEOMETRIC, for the month of October 2012.

The screenshot shows the Oracle SQL Developer interface. In the top tab bar, there are two tabs: 'nsedata' and 'nsedata'. The active tab is 'nsedata' with the sub-tab 'Script-3'. Below the tabs, a code editor window displays the following SQL script:

```

create table t3 as
select symbol,close,open,(close + open /2) as average
from nsedata
where symbol ='GEOMETRIC'and timestamp like '%Oct-2012';
select * from t3;

```

Below the code editor is a results grid titled 't3 1'. The grid has four columns: 'symbol', 'close', 'open', and 'average'. The data consists of 21 rows, each representing a record for the 'GEOMETRIC' company in October 2012. The 'average' column is calculated as the sum of 'close' and 'open' divided by 2. A tooltip on the right side of the grid says: 'Select a cell to view/edit value' and 'Press F7 to hide this panel'.

**Task 16** - It is required to create a table t4 such that it contains the data for two companies GEOMETRIC and TCS. The columns of this table should be as follows: timestamp, close\_tcs, close\_geometric. Hint: use JOIN

The screenshot shows the Oracle SQL Developer interface. In the top tab bar, there are two tabs: 'nsedata' and 'nsedata'. The active tab is 'nsedata' with the sub-tab 'Script-3'. Below the tabs, a code editor window displays the following SQL script:

```

create table t4 as
select
    n1.timestamp,
    n1.close as close_tcs,
    n2.close as close_geometric
from nsedata n1
join nsedata n2 on n1.timestamp = n2.timestamp
where n1.symbol = 'TCS' and n2.symbol = 'GEOMETRIC';
select * from t4;

```

Below the code editor is a results grid titled 't4 1'. The grid has three columns: 'timestamp', 'close\_tcs', and 'close\_geometric'. The data consists of 22 rows, each representing a record for the 'TCS' and 'GEOMETRIC' companies at different timestamps. A tooltip on the right side of the grid says: 'Select a cell to view/edit value' and 'Press F7 to hide this panel'.

**Task 17 -** Find out the maximum and minimum difference in the daily closing prices of these two companies.

```

nsedata nsedata *<localhost 3> Script-3 X
| select
|   max(close_tcs - close_geometric) as max_difference,
|   min(close_tcs - close_geometric) as min_difference
| from t4;

Results 1 X
iT select max(close_tcs - close_geometric) as ma| Enter a SQL expression to filter results (use Ctrl+Space)
Grid 123 max_difference 123 min_difference
1 2,631.65 770.35
Value X
Pan

```

**Task 18 -** Based on t4 can you identify those days on which the difference in their closing price was more than the average of the minimum and maximum difference.

```

nsedata nsedata *<localhost 3> Script-3 X
| select timestamp
| from t4
| where abs(close_geometric - close_tcs) > (
|   (select max(close_tcs - close_geometric) from t4) +
|   (select min(close_tcs - close_geometric) from t4)
| ) / 2;

t4 1 X
iT select timestamp from t4 where abs(close_gec| Enter a SQL expression to filter results (use Ctrl+Space)
Grid abc timestamp
1 01-APR-2014
2 01-APR-2015
3 01-AUG-2013
4 01-AUG-2014
5 01-DEC-2014
6 01-JAN-2014
7 01-JAN-2015
8 01-JUL-2014
9 01-JUL-2015
10 01-JUN-2015
11 01-NOV-2013
12 01-OCT-2013
13 01-OCT-2014
14 01-SEP-2014
15 02-APR-2014
16 02-AUG-2013
17 02-DEC-2013
18 02-DEC-2014
19 02-FEB-2015
20 02-JAN-2014
21 02-JAN-2015
22 02-JUL-2014
23 02-JUL-2015
24 02-JUN-2014
Value X
Panels
Select a cell to view/edit value
Press F7 to hide this panel

```

**Task 19** - Based on nsedata, create table t5 such that it contains the average close price of each company traded in the month of April 2012. The table should be sorted in descending order of the average close price.

The screenshot shows the MySQL Workbench interface. In the top tab bar, there are three tabs: 'nsedata', 'nsedata', and '\*<localhost 3> Script-3'. The 'Script-3' tab contains the following SQL code:

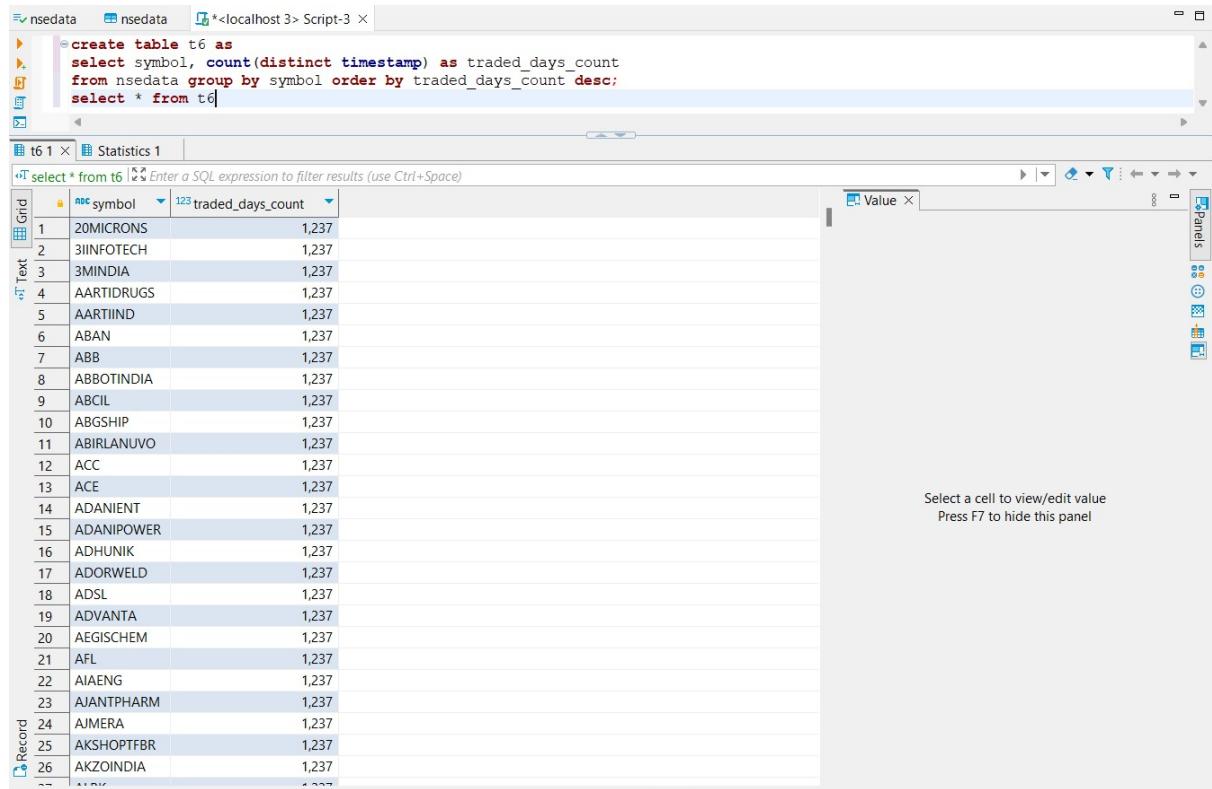
```
create table t5 as
select symbol, avg(close)as average_close
from nsedata where timestamp like '%Apr-2012' group by symbol order by average_close desc;
select * from t5;
```

Below the script, a results grid titled 't5 1' is displayed. It has two columns: 'symbol' and 'average\_close'. The data is sorted by 'average\_close' in descending order. The results are as follows:

	symbol	average_close
1	ORISSAMINE	34,041.3868421053
2	MRF	10,993.7
3	SBIN	8,620.1764646465
4	BOSCHLTD	8,504.1
5	TIDEWATER	7,691.1236842105
6	NESTLEIND	4,813.3921052632
7	3MINDIA	4,157.8105263158
8	ALFALALV	3,935.85
9	GODFRYPHLP	3,574.0394736842
10	ASIANPAINT	3,359.1184210526
11	TTKPRESTIG	3,355.0684210526
12	SHREECEM	2,911.0263157895
13	CRMGETF	2,904.695
14	HONAUT	2,902.7473684211
15	PAGEIND	2,900.8026315789
16	MGOLD	2,853.6631578947
17	IDBIGOOLD	2,842.9526315789
18	BSLGOLDETF	2,830.4131578947
19	GSKCONS	2,804.2263157895
20	RELIGAREGO	2,801.8763157895
21	SBIGETS	2,792.0578947368
22	IPGETF	2,790.2473684211
23	HDFCMFGETF	2,790.0236842105
24	AXISGOLD	2,788.3947368421
25	GOLDBEES	2,726.0526315789
26	KOTAKGOLD	2,722.1236842105
27	COLGATEP	2,700.4526315789

A message at the bottom right of the grid panel says: "Select a cell to view/edit value" and "Press F7 to hide this panel".

**Task 20** - Not all companies are traded every day. It is required to create a table that contains a count of the days each company has been traded. The table should be sorted in descending order of the count.



The screenshot shows the MySQL Workbench interface. A script editor window at the top contains the following SQL code:

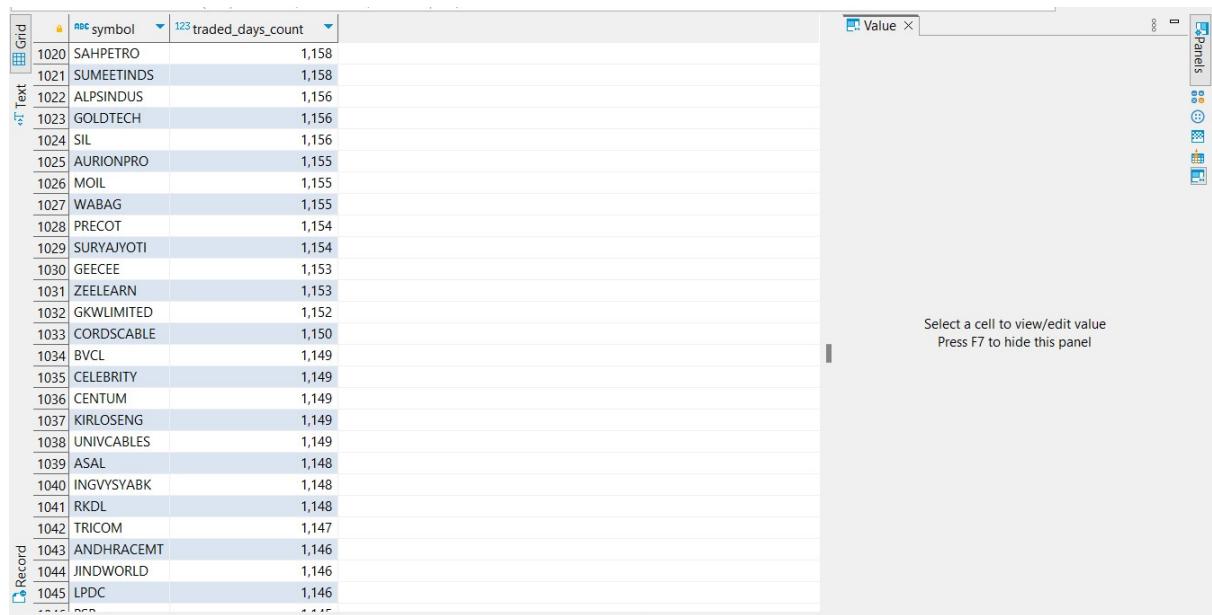
```

create table t6 as
select symbol, count(distinct timestamp) as traded_days_count
from nsedata group by symbol order by traded_days_count desc;
select * from t6

```

Below the script editor is a results grid titled "t6 1". The grid displays two columns: "symbol" and "traded\_days\_count". The data shows 26 records, all with a value of 1,237 for "traded\_days\_count". The "Grid" tab is selected. A tooltip on the right side of the grid says "Select a cell to view/edit value" and "Press F7 to hide this panel".

	symbol	traded_days_count
1	20MICRONS	1,237
2	3INFOTECH	1,237
3	3MINDIA	1,237
4	AARTIDRUGS	1,237
5	AARTIIND	1,237
6	ABAN	1,237
7	ABB	1,237
8	ABBOTINDIA	1,237
9	ABCIL	1,237
10	ABGSHIP	1,237
11	ABIRLANUVO	1,237
12	ACC	1,237
13	ACE	1,237
14	ADANIENT	1,237
15	ADANIPOWER	1,237
16	ADHUNIK	1,237
17	ADORWELD	1,237
18	ADSL	1,237
19	ADVANTA	1,237
20	AEGISCHEM	1,237
21	AFL	1,237
22	AIAENG	1,237
23	AJANTPHARM	1,237
24	AJMERA	1,237
25	AKSHOPTFBR	1,237
26	AKZOINDIA	1,237



The screenshot shows the MySQL Workbench interface. A script editor window at the top contains the following SQL code:

```

create table t6 as
select symbol, count(distinct timestamp) as traded_days_count
from nsedata group by symbol order by traded_days_count desc;
select * from t6

```

Below the script editor is a results grid titled "t6 1". The grid displays two columns: "symbol" and "traded\_days\_count". The data shows 45 records, all with a value of 1,158 for "traded\_days\_count". The "Grid" tab is selected. A tooltip on the right side of the grid says "Select a cell to view/edit value" and "Press F7 to hide this panel".

	symbol	traded_days_count
1020	SAHPETRO	1,158
1021	SUMEETINDS	1,158
1022	ALPSINDUS	1,156
1023	GOLDTECH	1,156
1024	SIL	1,156
1025	AURIONPRO	1,155
1026	MOIL	1,155
1027	WABAG	1,155
1028	PRECOT	1,154
1029	SURYAJYOTI	1,154
1030	GEECEE	1,153
1031	ZEELEARN	1,153
1032	GKWLIMITED	1,152
1033	CORDSCABLE	1,150
1034	BVCL	1,149
1035	CELEBRITY	1,149
1036	CENTUM	1,149
1037	KIRLOENG	1,149
1038	UNIVCABLES	1,149
1039	ASAL	1,148
1040	INGVSYABK	1,148
1041	RKDL	1,148
1042	TRICOM	1,147
1043	ANDHRACEMT	1,146
1044	JINDWORLD	1,146
1045	LPDC	1,146