DS 203 Assignment

Exercise 9
SQL

Kattirsitti Jeet Govindrao
(22B0010)

October 22, 2023

Tasks

1. Do a manual review of the table nsedata and describe its contents (no SQL to be executed for this task)
   The Stockdata database has a table named nsedata with records of each affiliated company's closing and opening prices of stocks. The fields consists of symbols, series, open, close, low, high, last, prevclose, isin, extra, anum, timestamp, tottrdqty, and tottrdal. It has 1893059 records and datatypes of various fields associated with the table are varchar(10), decimal, int, mediumint to store data for symbol/extra, open/close, tottrdqty, anum respectively.
2. Select the database stockdata using SQL
   USE stockdata;
   SHOW databases;(can be used to check the selected databases)
3. Get a schema dump of the table nsedata
   Get a count of the total number of records in nsedata
   SELECT * FROM nsedata;
   Answer - 1893059 no. of records
4. Get the total count of the records for the month "October 2012"
   SELECT count(*) FROM nsedata
   WHERE timestamp like '%-OCT-2012';
   Answer - 33244
5. Repeat '4', but only for the stock with symbol "GEOMETRIC"
   SELECT * FROM nsedata
   WHERE symbol = 'GEOMETRIC';

Tasks

1. Repeat ‘6’, but only display the first 10 records
   ```
   SELECT * FROM nsedata
   WHERE symbol = ’GEOMETRIC’
   LIMIT 10;
   ```
2. Totally, how many records of “INFY” does the table contain?
   ```
   SELECT count(*) FROM nsedata
   WHERE symbol = ’INFY’
   ```
3. 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
   ```
   select symbol,open,high,low,close,timestamp from nsedata
   where symbol=’3IINFOTECH’
   limit 10;
   ```
4. Repeat ’9’, but this time use the results to create a temporary table t1
   ```
   create temporary table t1 as
   select symbol,open,high,low,close,timestamp as t1 from nsedata
   where symbol=’3IINFOTECH’
   limit 10;
   ```
5. Using t1 find out the following for the column close: max, min, mean. standard deviation and variance
   ```
   select max(close),min(close),stddev(close),variance(close) from t1;
   ```

Tasks

1. How will you find out the value of the median?
   Median can be calculated by first ordering in ascending or descending order the particular field and then accessing the mid record of that field.
   ```
   select close from t1
   order by close;
   ```
2. Delete table t1 `drop table t1;`
3. 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
   ```
   create table t2 select symbol,avg(close) as average from nsedata group by
   symbol;
   ```
4. 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.
   ```
   create table t3 as select symbol,open,close,(open+close)/2 as average_o_c from
   nsedata
   where symbol='GEOMETRIC' and timestamp like '%-OCT-2012';
   ```

Tasks

1. 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

   ```
   CREATE TABLE t4 AS
   SELECT n.timestamp, tcs.close AS close_tcs, geometric.close AS close_geometric FROM nsedata n
   LEFT JOIN nsedata tcs ON n.timestamp = tcs.timestamp AND tcs.symbol = 'TCS'
   LEFT JOIN nsedata geometric ON n.timestamp = geometric.timestamp AND geometric.symbol = 'GEOMETRIC'
   WHERE n.symbol IN ('TCS', 'GEOMETRIC');
   ```

2. Find out the maximum and minimum difference in the daily closing prices of these two companies.

   ```
   select max(abs(close_tcs-close _geometric)) as M1,min(abs(close_tcs-close_geometric)) as M2 from t4;
   ```

3. 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.

   ```
   SELECT * FROM t4
   WHERE ABS(close_tcs - close_geometric) > ( (SELECT MAX(ABS(close_tcs - close_geometric)) FROM t4) + (SELECT MIN(ABS(close_tcs - close_geometric)) FROM t4) ) / 2;
   ```

Tasks

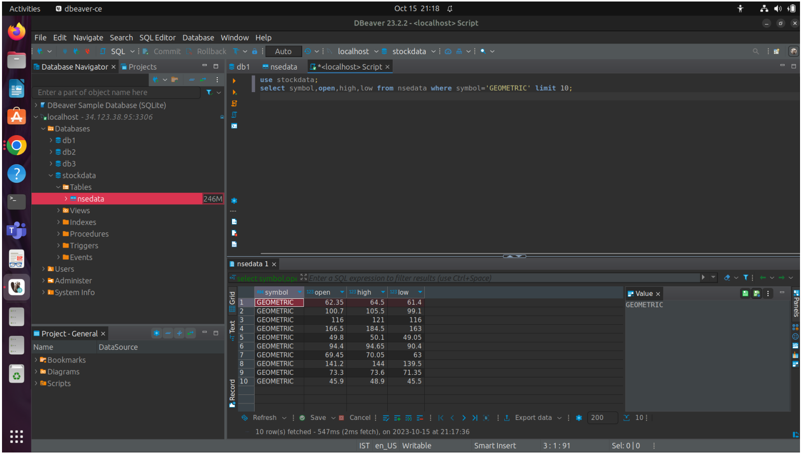
1. Based on nsedata, create table t5 such that it contains the average close price of each company traded in the month of April 2012. The table should be sorted in descending order of the average close price.

   ```
   CREATE TABLE t5 AS SELECT symbol, AVG(close) AS close_price FROM
   nsedata
   WHERE timestamp like ’%-APR-2012’ GROUP BY symbol
   ORDER BY close_price DESC;
   ```

2. 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.

   ```
   CREATE TABLE t5 AS SELECT symbol, COUNT(DISTINCT timestamp) AS
   trading_days FROM nsedata
   GROUP BY symbol
   ORDER BY trading_days DESC;
   ```

First 10 records with symbol='GEOMETRIC'

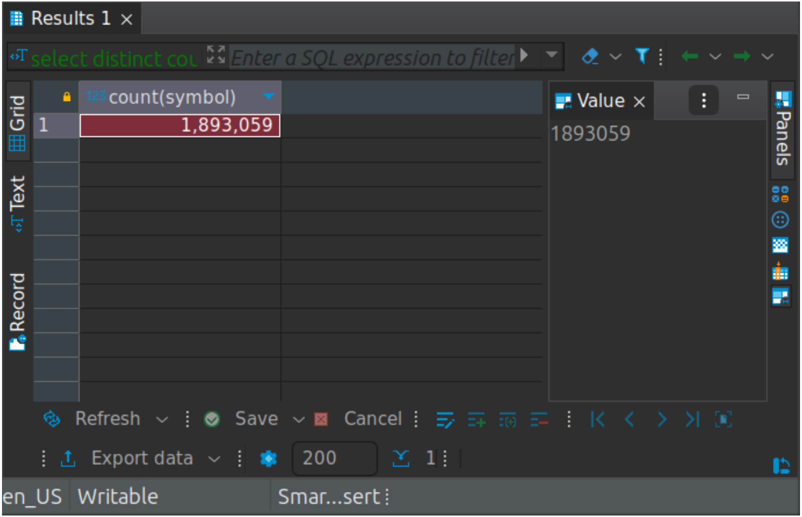


```
use stockdata;
select symbol,open,high,low from nsedata where symbol='GEOMETRIC' limit 10;
```

	symbol	open	high	low
1	GEOMETRIC	62.35	64.5	61.4
2	GEOMETRIC	100.7	105.5	99.1
3	GEOMETRIC	116	121	116
4	GEOMETRIC	166.5	184.5	163
5	GEOMETRIC	49.8	50.1	49.05
6	GEOMETRIC	94.4	94.65	90.4
7	GEOMETRIC	69.45	70.05	63
8	GEOMETRIC	141.2	144	139.5
9	GEOMETRIC	73.3	73.6	71.35
10	GEOMETRIC	45.9	48.9	45.5

```
10 row(s) fetched - 547ms (2ms fetch), on 2023-10-15 at 21:17:36
```

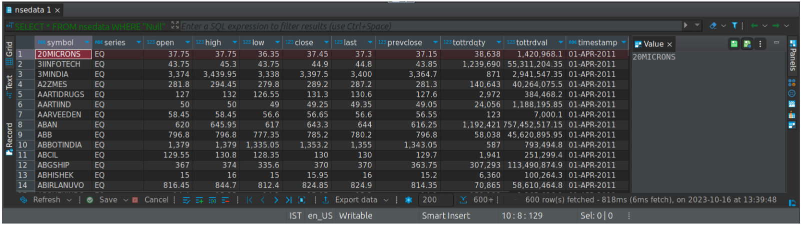
Number of records



	count(symbol)
1	1,893,059

1893059

Data schema of nsedata



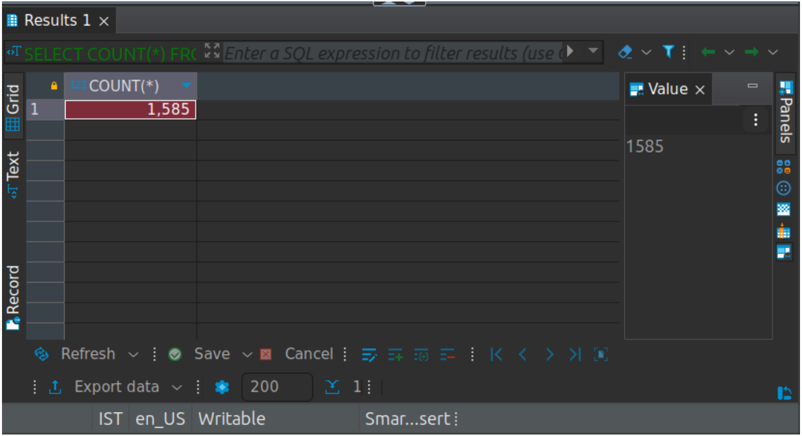
```
SELECT * FROM nsedata WHERE "Null"
```

Enter a SQL expression to filter results (use Ctrl+Space)

	symbol	series	open	high	low	close	last	prevclose	tottrdqty	tottrdval	timestamp
1	20MICRONS	EQ	37.75	37.75	36.35	37.45	37.3	37.15	38,638	1,420,968.1	01-APR-2011
2	3IINFOTECH	EQ	43.75	45.3	43.75	44.9	44.8	43.85	1,239,690	55,311,204.35	01-APR-2011
3	3MINDIA	EQ	3,374	3,439.95	3,338	3,397.5	3,400	3,364.7	871	2,941,547.35	01-APR-2011
4	A2ZMES	EQ	281.8	294.45	279.8	289.2	287.2	281.3	140,643	40,264,075.5	01-APR-2011
5	AARTIDRUGS	EQ	127	132	126.55	131.3	130.6	127.6	2,972	384,468.2	01-APR-2011
6	AARTIIND	EQ	50	50	49	49.25	49.35	49.05	24,056	1,188,195.85	01-APR-2011
7	AARVEEDEN	EQ	58.45	58.45	56.6	56.65	56.6	56.55	123	7,000.1	01-APR-2011
8	ABAN	EQ	620	645.95	617	643.3	644	616.25	1,192,421	757,452,517.15	01-APR-2011
9	ABB	EQ	796.8	796.8	777.35	785.2	780.2	796.8	58,038	45,620,895.95	01-APR-2011
10	ABBOTINDIA	EQ	1,379	1,379	1,335.05	1,353.2	1,355	1,343.05	587	793,494.8	01-APR-2011
11	ABCIL	EQ	129.55	130.8	128.35	130	130	129.7	1,941	251,299.4	01-APR-2011
12	ABGSHIP	EQ	367	374	335.6	370	370	363.75	307,293	113,490,874.9	01-APR-2011
13	ABHISHEK	EQ	15	16	15	15.95	16	15.2	6,360	100,264.3	01-APR-2011
14	ABIRLANUVO	EQ	816.45	844.7	812.4	824.85	824.9	814.35	70,865	58,610,464.8	01-APR-2011

```
20MICRONS
```

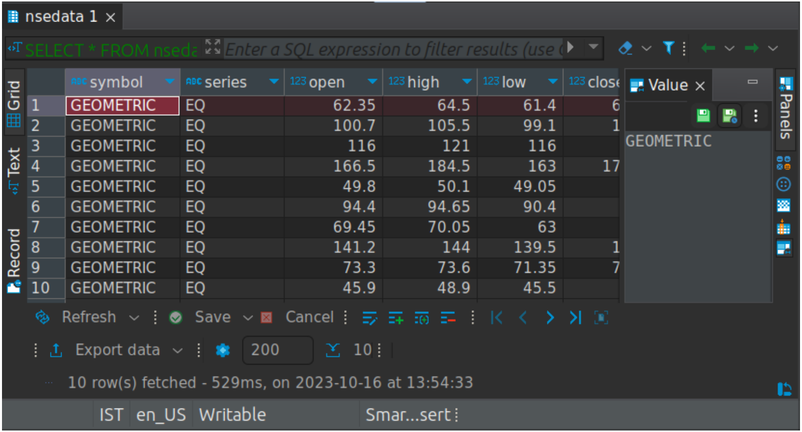
```
600 row(s) fetched - 818ms (6ms fetch), on 2023-10-16 at 13:39:48
```



```
SELECT COUNT(*) FRC
```

	COUNT(*)
1	1,585

Value: 1585



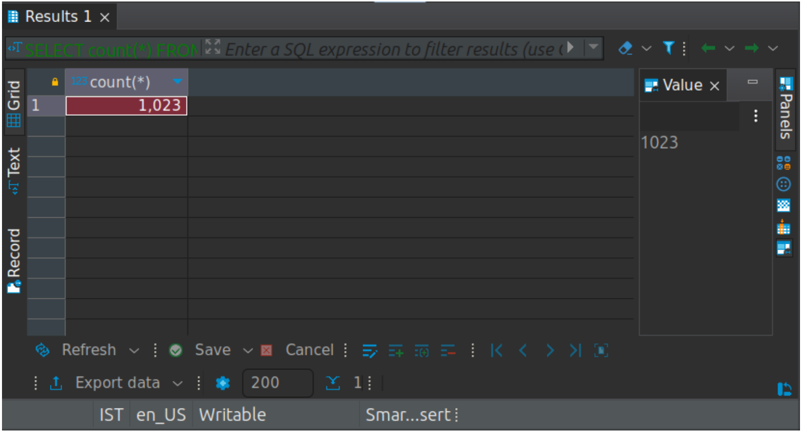
```
SELECT * FROM nseda
```

Enter a SQL expression to filter results (use

	symbol	series	open	high	low	close
1	GEOMETRIC	EQ	62.35	64.5	61.4	6
2	GEOMETRIC	EQ	100.7	105.5	99.1	1
3	GEOMETRIC	EQ	116	121	116	
4	GEOMETRIC	EQ	166.5	184.5	163	17
5	GEOMETRIC	EQ	49.8	50.1	49.05	
6	GEOMETRIC	EQ	94.4	94.65	90.4	
7	GEOMETRIC	EQ	69.45	70.05	63	
8	GEOMETRIC	EQ	141.2	144	139.5	1
9	GEOMETRIC	EQ	73.3	73.6	71.35	7
10	GEOMETRIC	EQ	45.9	48.9	45.5	

GEOMETRIC

10 row(s) fetched - 529ms, on 2023-10-16 at 13:54:33

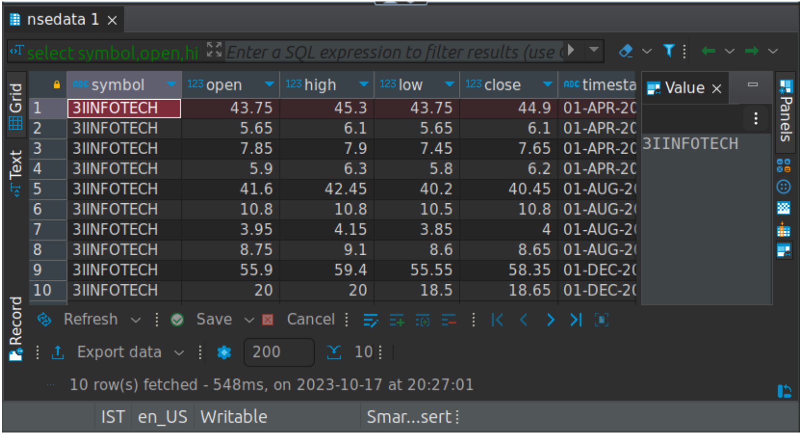


```
SELECT count(*) FROM
```

	count(*)
1	1,023

Value: 1023

First 10 records with symbol='3IINFOTECH'



`select symbol,open,hi`

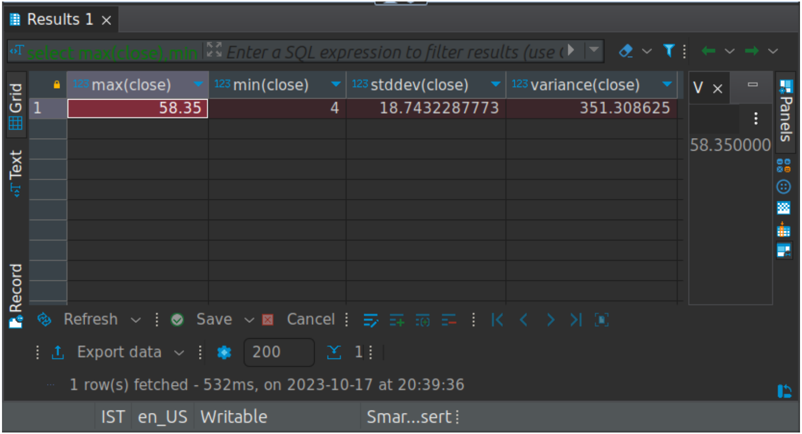
Enter a SQL expression to filter results (use

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

3IINFOTECH

```
10 row(s) fetched - 548ms, on 2023-10-17 at 20:27:01
```

max,min,variance



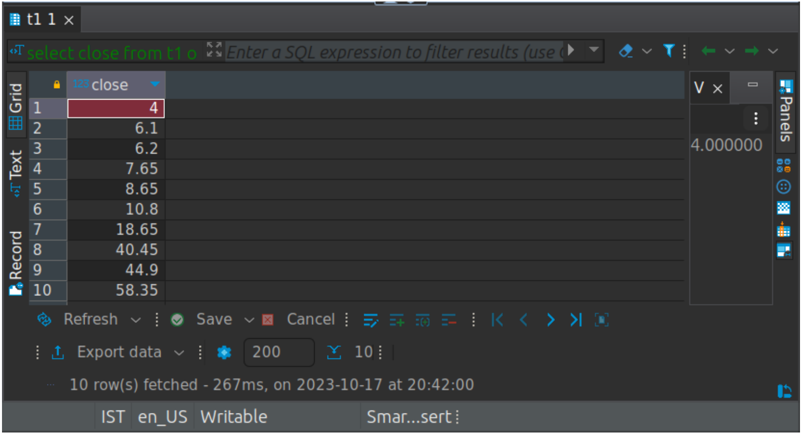
select max(close),min

	max(close)	min(close)	stddev(close)	variance(close)
1	58.35	4	18.7432287773	351.308625

58.350000

1 row(s) fetched - 532ms, on 2023-10-17 at 20:39:36

Average of the 5th and 6th record will be the median



`select close from t1 o`

Enter a SQL expression to filter results (use

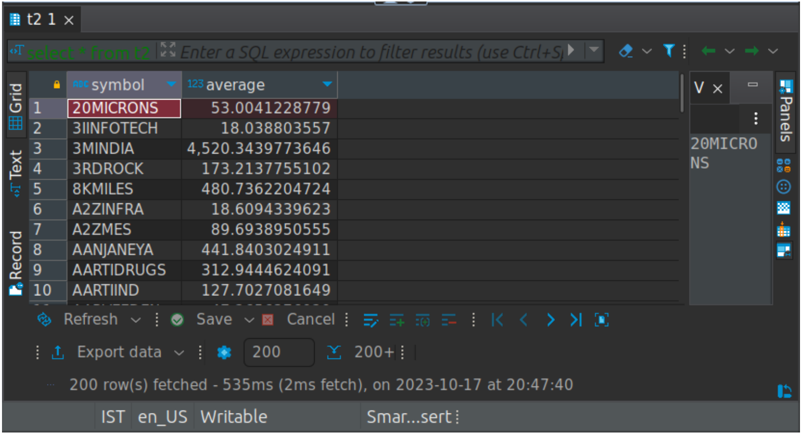
	close
1	4
2	6.1
3	6.2
4	7.65
5	8.65
6	10.8
7	18.65
8	40.45
9	44.9
10	58.35

4.000000

10 row(s) fetched - 267ms, on 2023-10-17 at 20:42:00

IST en_US Writable Smar...sert

Average in particular month



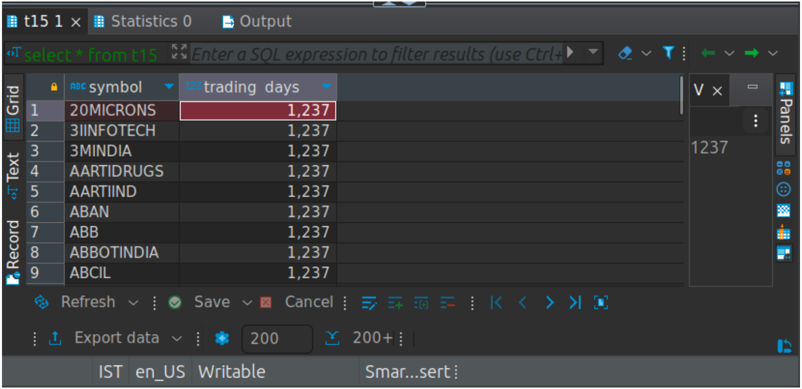
```
select * from t2
```

	symbol	average
1	20MICRONS	53.0041228779
2	3IINFOTECH	18.038803557
3	3MINDIA	4,520.3439773646
4	3RDROCK	173.2137755102
5	8KMILES	480.7362204724
6	A2ZINFRA	18.6094339623
7	A2ZMES	89.6938950555
8	AANJANEYA	441.8403024911
9	AARTIDRUGS	312.9444624091
10	AARTIIND	127.7027081649

```
20MICRONS
```

```
200 row(s) fetched - 535ms (2ms fetch), on 2023-10-17 at 20:47:40
```

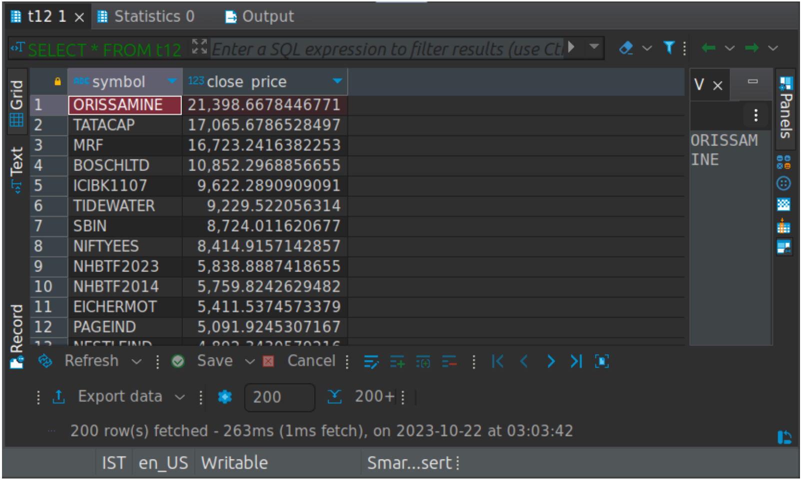
Traded days



```
select * from t15
```

	symbol	trading days
1	20MICRONS	1,237
2	3IINFOTECH	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

1237

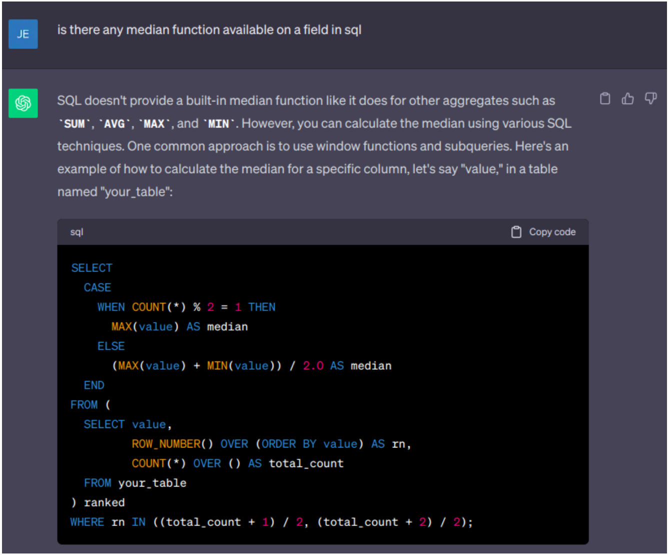


```
SELECT * FROM t12
```

	symbol	close_price
1	ORISSAMINE	21,398.6678446771
2	TATACAP	17,065.6786528497
3	MRF	16,723.2416382253
4	BOSCHLTD	10,852.2968856655
5	ICIBK1107	9,622.2890909091
6	TIDEWATER	9,229.522056314
7	SBIN	8,724.011620677
8	NIFTYEES	8,414.9157142857
9	NHBTF2023	5,838.8887418655
10	NHBTF2014	5,759.8242629482
11	EICHERMOT	5,411.5374573379
12	PAGEIND	5,091.9245307167

ORISSAMINE

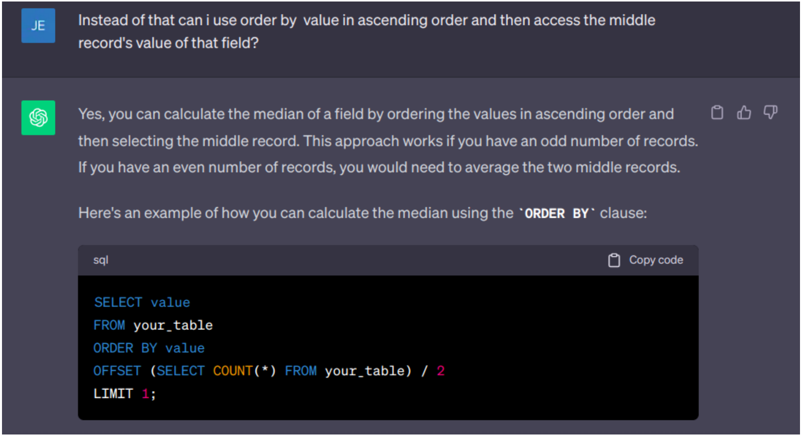
200 row(s) fetched - 263ms (1ms fetch), on 2023-10-22 at 03:03:42



is there any median function available on a field in sql

SQL doesn't provide a built-in median function like it does for other aggregates such as `SUM`, `AVG`, `MAX`, and `MIN`. However, you can calculate the median using various SQL techniques. One common approach is to use window functions and subqueries. Here's an example of how to calculate the median for a specific column, let's say "value," in a table named "your_table":

```sql
SELECT
  CASE
    WHEN COUNT(*) % 2 = 1 THEN
      MAX(value) AS median
    ELSE
      (MAX(value) + MIN(value)) / 2.0 AS median
  END
FROM (
  SELECT value,
         ROW_NUMBER() OVER (ORDER BY value) AS rn,
         COUNT(*) OVER () AS total_count
  FROM your_table
) ranked
WHERE rn IN ((total_count + 1) / 2, (total_count + 2) / 2);
```



Instead of that can i use order by value in ascending order and then access the middle record's value of that field?

Yes, you can calculate the median of a field by ordering the values in ascending order and then selecting the middle record. This approach works if you have an odd number of records. If you have an even number of records, you would need to average the two middle records.

Here's an example of how you can calculate the median using the `ORDER BY` clause:

```sql
SELECT value
FROM your_table
ORDER BY value
OFFSET (SELECT COUNT(*) FROM your_table) / 2
LIMIT 1;
```