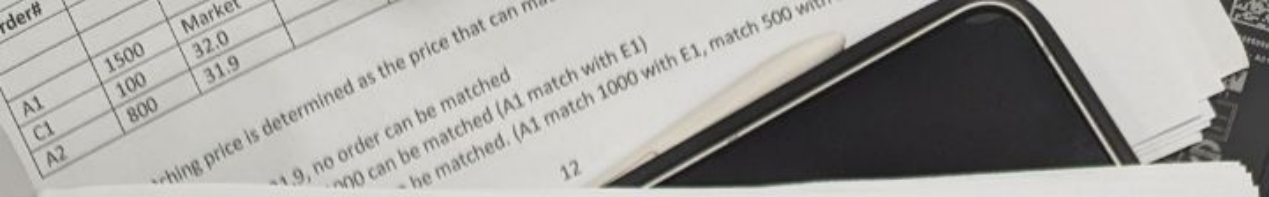
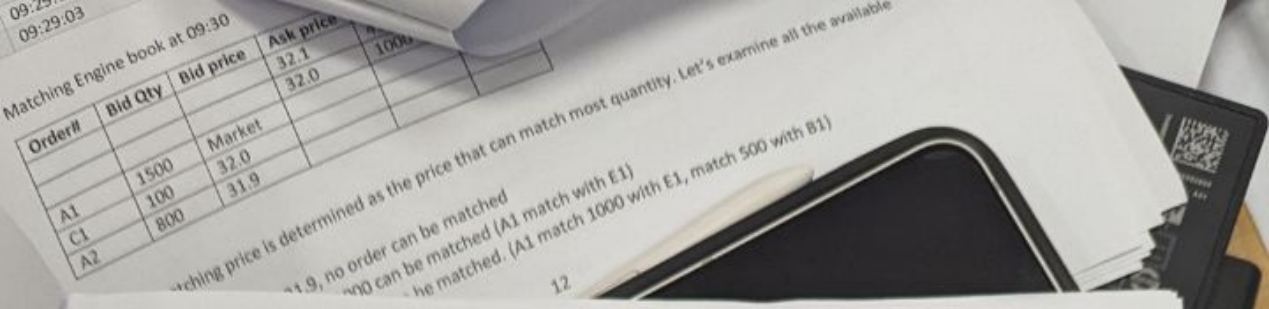


Contents



1 Background and Introduction

The concept of a Matching Engine is pivotal in the financial markets, serving as the core technology that facilitates the execution of trades by matching buy and sell orders based on defined criteria. Matching engine (ME) is the core concept that serves many subdivisions within Global Market Technology (GMT). Its origin traces back to the transformation of financial markets from physical trading floors to electronic systems, necessitating an efficient, automated solution for order matching to ensure market liquidity and efficiency. Today, MEs are integral to not only stock exchanges but to any sort of exchange businesses.

With the advent of high-frequency trading (HFT), the demand for faster and more reliable matching engines has intensified, pushing the limits of current technologies. The primary challenge in designing a Trade Matching Engine to potentially process millions of orders with minimal latency, ensuring that trades are executed fairly and swiftly.

In this problem statement, we will delve into Trade Matching Engine and its core components such as Order book, Matching Logic, Execution and Reporting.

Order book maintains a real-time, organized list of buy and sell orders. The order book is crucial for transparency, allowing traders to make informed decisions based on current market demand and supply.

The matching logic is the algorithm responsible for determining how orders should be matched to execute a trade. This component must be meticulously designed to ensure fairness and adherence to market rules.

The basic version of Trade Matching Engine should be able to validate and accepts orders based on client configuration. It should match the orders based on different algorithms in the trading session.

At the end of trading day, consolidated reports are required to be generated as post trade records.

Your Task for this challenge is to:

1. Develop a Matching Engine.
2. Generate consolidated reports at the end of matching day.

Above 2 tasks are essential for the submission to be considered a completed challenge. However, the submission could be enhanced by building nice GUI based dashboards for visualization of the trades and reports – **while these are not necessary, they may be a bonus in the final submission leaderboard.**

Matching Engine book at

Order#	Bid Qty	Bid price	Ask p
			32.1
	1500	Market	32.0
A1	100	32.0	
C1	800	31.9	
A2			

1000

matching price is determined as the price that can match most quantity. Let's examine all the

31.9, no order can be matched

ong can be matched (A1 match with E1)

be matched. (A1 match 1000 with E1, match 500 with B1)

2 Matching Engine

2.1 Data Specification

2.1.1 Client Data specification:

Client profile data consist of 4 components:

ClientID	Unique identifier of the client
Currencies	A list of currencies that the client has opened account with us. Clients are only able to trade instruments issued with these currencies.
PositionCheck	This is a flag to identify whether the client is required to perform position check. A position is the amount of asset that is owned or sold by the client. "Short Sell" refer to the activity that client wants to sell a stock without buying it. Short sell is NOT allowed for the clients who requires position checks.
Rating	A number from 1 to 10, with 1 as highest ranking and 10 as lowest. Higher rating will give client priority in matching compared to clients of lower rating.

Client Data sample:

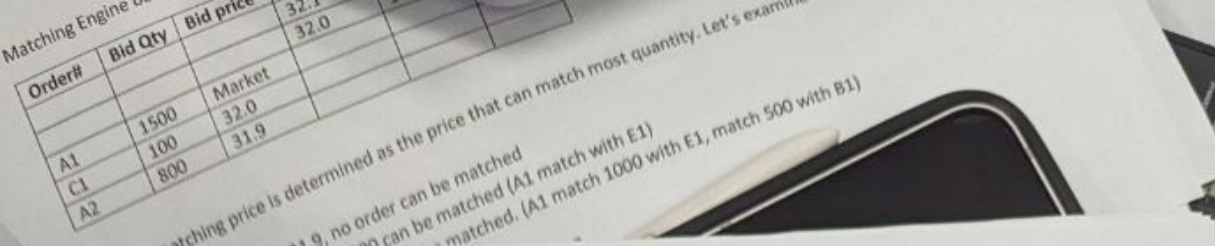
ClientID	Currencies	PositionCheck	Rating
A	USD, SGD	Y	2
B	USD, SGD, JPY	N	1
C	SGD	Y	3
D	USD	Y	4
E	SGD	N	3
...			

2.1.2 Instruments Data specification:

Financial Instruments are monetary contracts that can be traded, such as stock, bond, derivatives.

Instrument data consists of 3 parts:

InstrumentID	Unique identifier of the instrument
Currency	Currency that the instrument is issued with.
LotSize	The lowest size multiplier of instrument units that client can trade for this instrument in one order. For example, if an instrument has lot size 100, an order with quantity 200 or 1000 is valid, but 60 or 1030 is not valid.



Instrument Data sample:

InstrumentID	Currency	LotSize
0001	USD	1
0003	SGD	100
0504	USD	50
...		

2.1.3 Order Data specification:

Time	Time that order arrives at matching engine
OrderID	Unique identifier to identify the order
Client	The client who sent the order
Instrument	The ID of instrument to trade in
Side	Action of the order, buy or sell
Price	A **Limit Order** will have a numerical value in this field, specifying a price at or better than which to buy or sell. Buy limit orders can only be executed at the limit price or lower. Sell limit orders can only be executed at the limit price or higher. A **Market Order** will be labeled with text value "Market" under this column. The actual numerical execution price of market orders will be determined based on current best available prices on the order book to match and execute.
Quantity	Number of instrument units to trade for this order

Order data sample:

Time	OrderID	Client	Instrument	Side	Price	Quantity
09:30:01	O1	ABC001	0001	Buy	Market	100
09:30:02	O2	XYZ056	0001	Sell	101	500
09:30:04	O3	XYZ056	0003	Buy	102	200
09:30:05	O4	ABC001	0001	Sell	Market	300
09:30:05	O5	XYZ056	0504	Buy	10.1	5
...	....	...		...	...	...

Matching Engi

Order#	Bid Qty	Bid price
		32.1
		32.0
	Market	
A1	1500	32.0
C1	100	31.9
A2	800	

Auction matching price is determined as the price that can match most quantity. Let's examine all the available prices.

...ce match price = 31.9, no order can be matched

...ce = 32.0, 1000 can be matched (A1 match with E1)

...500 can be matched. (A1 match 1000 with E1, match 500 with B1)

2.2 Matching Policies

Matching Engine should adhere to the following policies. All orders sent to matching engine needs to be validated based on the policies. Only orders that passed all validations can be added to order book and get potentially matched.

Orders that didn't pass the policy checking will be rejected by the matching engine and will need to be reported in the end of report with rejection reason.

Please follow the below sequence of polices in the table while processing the orders - if check #1 fails, there is no need to perform check #2.

#	Policy Check	Description	Rejection Reason
1	Instrument	Instrument sent with the order is not registered in instrument data	**REJECTED – INSTRUMENT NOT FOUND**
2	Currency	Clients are not allowed to trade stock out of their allowed currency list.	**REJECTED – MISMATCH CURRENCY**
3	Lot Size	If order quantity is not a rounded lot, order should be rejected. **Example:** Referring instrument data, #0003 has lot size of 100, and if order has been received for 80, or 180 shares, then this order should be rejected.	**REJECTED – INVALID LOT SIZE**
4	Position	If client is set to perform position check, sell order that exceeds client's current position of this instrument should be rejected.	**REJECTED – POSITION CHECK FAILED**

Order#					
A1	1500	Market			
C1	100	32.0			
A2	800	31.9			

Auction matching price is determined as the price that can match most quantity. Let's examine all the available prices.

- match price = 31.9, no order can be matched
- 32.0, 1000 can be matched (A1 match with E1)
- can be matched. (A1 match 1000 with E1, match 500 with B1)

2.3 Matching Priority

Matching Engine maintains an order queue for each registered instrument, based on ***Price-Rating-Time*** priority.

1st priority: order price. For buy orders, higher price has high priority. For sell side, lower price has higher priority. Market orders have higher priority over limit order on both sides.

2nd priority: client rating. Client with higher rating has higher priority in matching.

3rd priority: order arrival time. Order that arrives earlier has higher priority.

Suppose we have the below 5 sell-side orders in the order book for Stock id 0001:

OrderID	ClientID	Rating	Arrival time	Price	Quantity
1	B	1	09:30:00	101	500
2	C	2	09:30:01	100	100
3	A	2	09:31:00	100	150
4	B	1	09:35:00	100	50
5	E	3	09:40:00	Market	10

When another buy order with price=103, quantity=400 is received, the correct matching sequence should be:

#1 | Order 5, matched 10 shares, buy order left with 390, Market order has higher priority.
#2 | Order 4, matched 50 shares, buy order left with 340, at price level 100, client B has highest rating.
#3 | Order 2, matched 100 shares, buy order left with 240, at price level 100 and rating 2, Order 2 was earlier.
#4 | Order 3, matched 150 shares, buy order left with 90.
#5 | Order 1, matched 90 shares, 410 shares open in the queue, buy order fully executed.

Order#					
A1	1500	Market			
C1	100	32.0			
A2	800	31.9			

Auction matching price is determined as the price that can match most quantity. Let's examine all the available prices.

...tch price = 31.9, no order can be matched

...n 1000 can be matched (A1 match with E1)

...be matched. (A1 match 1000 with E1, match 500 with B1)

2.4 Matching Algorithms

Matching Engine should have capability to handle 2 types of matching algorithms, **Auction** and **Continuous** within 3 different trading sessions namely Open Auction, Continuous Trading, Closed Auction.

1. Open Auction (09:00 – 09:30)
    i. Client orders are queued.
    ii. Matching only happens at 09:30, that's when open price is determined.
    iii. Open price is determined based on the bid or ask price in the order book that can achieve maximum matched order quantity.
    iv. Orders are matched at open price and trades are generated at 09:30. Any unfilled orders will be carried to the next trading session.
    v. Crossing in open auction will not happen if:
        i. Only market orders available at both buy and sell side of the book.
        ii. OR No bid and ask price overlap.

2. Continuous Trading (09:30 – 16:00)
    i. Client orders are continuously inserted, matched, and filled.
    ii. Unfilled orders at the end of session will be carried to Close Auction session.
    iii. Match price is determined by the order that arrives earlier (see examples in appendix)
    iv. Notes on Market order:
        i. Market order can't be matched with Market order (Market order can only be matched with limit order).
        ii. Unfilled market order will be queued until next available opposite limit order.

3. Close Auction (16:00 – 16:10)
    i. Close price is determined the same mechanism as open auction mentioned in point 2.4.1.
    ii. Orders matching only happens at 16:10. Close price is determined, and close trades are generated.
    iii. Like open auction, close auction crossing will not occur in the following scenario:
        i. Only market orders available at both buy and sell side of the book.
        ii. OR No bid and ask price overlap.

Refer appendix for detailed examples.

A2	800

Auction matching price is determined as the price that can match most quantity. Let's examine all the available prices.

If choose match price = 31.9, no order can be matched

If choose match price = 32.0, 1000 can be matched (A1 match with E1)

If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

3 Consolidated Reports

Matching Engine will generate reports at the end of each trading day as a summary of today's trading activities. There are three kinds of reports namely Exchange, Client, and Instrument.

1. Exchange Report: A list of orders that didn't pass policy checks in the trading day.
2. Client Report: Position of each client at the end of the trading day, for each instrument.
3. Instrument Report: For each instrument, what is the:
    a. Open Price
        i. Open price is the matching price at open auction. If no match happens at open auction, this is determined to be the first trade price of the day after open.
    b. Closed Price (the trade price of the instrument in Close Auction).
    c. Total Traded Volume – Sum of total buy and sell quantity for the day.
    d. Day High/Low – Highest and Lowest matched price of the day.
    e. Volume Weighted Average Price (VWAP) of the day.
    Matching engine should have kept the record of every matching that happened in the day, each matching consisting of matched price and matched volume. With this info, we can calculate:

$$\boldsymbol{VWAP} = \frac{\sum \boldsymbol{Matched\ Price} \times \boldsymbol{Matched\ Volume}}{\sum \boldsymbol{Matched\ Volume}}$$

Suppose 4 matches happened for stock #0001 in the day:

#1 | 100 shares, 101.5
#2 | 5000 shares, 100
#3 | 50 shares, 103
#4 | 105 shares, 102.5

$$VWAP_{0001} = \frac{100 \times 101.5 + 5000 \times 100 + 50 \times 103 + 105 \times 102.5}{100 + 5000 + 50 + 105} = 100.1070$$

Matching scenarios output can be used as test case input for generating consolidate reports.

C1		
A2	800	31.9

Auction matching price is determined as the price that can match most quantity. Let's examine all the available prices.

If choose match price = 31.9, no order can be matched
If choose match price = 32.0, 1000 can be matched (A1 match with E1)
If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

12

4 Requirement for the challenge

4.1 Essential Components

1. Read data from CSV files **instruments.csv, clients.csv, orders.csv** and create components listed below.
    a. **Matching Engine:** Implement a real-world like exchange per the guideline provided in section 2.
    b. **Consolidated Report:** End of day reporting. Refer to section 3.
2. Share GitHub repo with final implementation at the end of the challenge for evaluation purpose.
3. A presentation deck, consisting of no more than 7 slides, to present your solution. Suggestion is provided in section 4.5.

4.2 Implementation Boundaries and Assumptions

1. Number of currencies are not limited to initial data set.
2. Assume **single trading day.**
3. All **final output data** related to price should be round up to 4 decimal places.
4. You are free to use any language and libraries.
5. **Do not use any code generation platform**.

4.3 Requirements

1. Provide clear instructions to run the application and other implementation details.
2. Produce logs when needed.
3. We will provide one set of sample input and output files besides the example provided in this document.
4. We will be giving testing input files in similar format as CSV files mentioned in section 4.1 before final submission. You are required to **process all inputs and send back your result output files (CSV)** via email provided on the challenge day. Output files will be used to evaluate your work.
5. We are expecting 3 output files in csv format named as: -
    - **output_exchange_report.csv**
    - **output_client_report.csv**
    - **output_instrument_report.csv**
6. Please create a public github repo and upload your code folder, output files and presentation deck. Kindly share the github repo link in your submission email.

4.4 Marking Scheme

Item	Points
Matching Engine, implement matching priority, algorithms, and policies correctly	50
Report, calculation, and accuracy of reporting values	20
Overall design, architecture, coding practice, component integration.	10
Presentation deck on thinking process, design considerations, algorithm complexity, solutions and challenges faced.	10
Test Coverage – Write down the unit test and integration test for core components.	10
Subtotal	100

A1	1500	Market			
C1	100	32.0			
A2	800	31.9			

Auction matching price is determined as the price that can match most quantity. Let's examine all the available prices.

If choose match price = 31.9, no order can be matched

If choose match price = 32.0, 1000 can be matched (A1 match with E1)

If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

4.5 Presentation

The following is a suggested format of your presentation. This is for your reference only. Selected teams will get only 5 minutes to do their presentation.

1. Cover Page: include your team's name and member details.
2. Overview: describe the high-level approach taken for the solution and components you have attempted.
3. Design and architecture of your solution.
4. Consolidated Reports demo/ screenshots.
5. Optionally, any additional commentary on your solution.
6. Takeaways and conclusions.

5 Words of Advice

This project is intended to give you an exposure into how engineers work at the bank i.e., you get to work on everything which you like, where you feel challenged and where there is immense learning. Divide the work within your team as per different components and individual strength. Try to work on simple, correct solution first with support of unit tests to verify and then improve it further. Moreover, an important part of working at the Bank is making it easy for the users and fellow engineers, so writing well documented code that is easy to maintain is always encouraged, along with any supportive documentation.

All the very best and make sure you enjoy the project!

A2	800							

Auction matching price is determined as the price that can match most quantity. Let's examine all the available prices.
If choose match price = 31.9, no order can be matched
If choose match price = 32.0, 1000 can be matched (A1 match with E1)
If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

6 Appendix

Example Illustration with 1 instrument:

1. **A list of instruments: input_instruments.csv**

#	InstrumentID	Currency	LotSize
1	SIA	SGD	100

2. **A list of client information: input_clients.csv**

#	ClientID	Currencies	PositionCheck	Rating
1	A	USD, SGD	Y	1
2	B	USD, SGD, JPY	N	2
3	C	SGD	Y	3
4	D	USD	Y	4
5	E	SGD	N	5

3. **A list of orders: input_orders.csv**

1) Open Auction session orders (until 09:30)

#	Time	Client	Instrument	Side	Price	Quantity	OrderID	Result (not part of orders.csv – included here for explanation)
1	09:00:01	A	SIA	Buy	Market	1500	A1	Accept
2	09:02:00	B	SIA	Sell	32.1	4500	B1	Accept
3	09:05:00	C	SIA	Buy	32.0	100	C1	Accept
4	09:10:00	D	SIA	Sell	Market	300	D1	REJECTED – MISMATCH CURRENCY
5	09:29:01	B	SIA	Sell	32.1	5	B2	REJECTED – INVALID LOT SIZE
6	09:29:02	E	SIA	Sell	32.0	1000	E1	Accept
7	09:29:03	A	SIA	Buy	31.9	800	A2	Accept

Matching Engine book at 09:30

Order#	Bid Qty	Bid price	Ask price	Ask Qty	Order#
			32.1	4500	B1
			32.0	1000	E1
A1	1500	Market			
C1	100	32.0			
A2	800	31.9			

Auction matching price is determined as the price that can match most quantity. Let's examine all the available prices.
If choose match price = 31.9, no order can be matched
If choose match price = 32.0, 1000 can be matched (A1 match with E1)
If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

prices.
If choose match price = 31.9, no order can be matched
If choose match price = 32.0, 1000 can be matched (A1 match with E1)
If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

Hence, open price is 32.1, open quantity = 1500 (maximum matched order quantity).
Trades generated (for your reference):

Order Buy	Trade	Order Sell
A1	1000@32.1	E1
A1	500@32.1	B1

After open auction, matching engine transit to continuous trading session. Unmatched orders will be carried forward. So initial snapshot of order book at 09:30 is:

Order#	Bid Qty	Bid price	Ask price	Ask Qty	Order#
			32.1	4000	B1
C1	100	32.0			
A2	800	31.9			

Initial snapshot of client position at 09:30 is:

#	Client ID	Position of instrument SIA
1	A	1500
2	B	-500
3	C	0
4	D	0
5	E	-1000

2) Continuous Trading Session (order get matched continuously)

New Order #8

#	Time	Client	Instrument	Side	Price	Quantity	OrderID	Result (not part of orders.csv – included here for explanation)
8	09:30:01	C	SIA	Sell	Market	100	C2	REJECTED – POSITION CHECK FAILED

9

New Order #9:

#9	09:40:00	B	SIA	Sell	32.2	500	B3	Accept

Book after update:

Order#	Bid Qty	Bid price	Ask price	Ask Qty	Order#
			32.2	500	B3
			32.1	4000	B1
C1	100	32.0			
A2	800	31.9			

prices.
If choose match price = 31.9, no order can be matched
If choose match price = 32.0, 1000 can be matched (A1 match with E1)
If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

10

New Order #10

#10	10:50:00	C	SIA	Buy	32.2	4200	C3	Accept

2 trades are generated:

Order Buy	Trade	Order Sell
C3	4000@32.1	B1
C3	200@32.2	B3

Book after update:

Order#	Bid Qty	Bid price	Ask price	Ask Qty	Order#
			32.2	300	B3
C1	100	32.0			
A2	800	31.9			

11

New Order #11

#11	11:10:00	B	SIA	Sell	Market	100	B4	Accept

1 trade is generated:

Order Buy	Trade	Order Sell
C1	100@32	B4

Book after update:

Order#	Bid Qty	Bid price	Ask price	Ask Qty	Order#
			32.2	300	B3
A2	800	31.9			

12

New Order #12

#12	12:30:00	E	SIA	Buy	31.9	600	E2	Accept

No Trade:

Book after update:

Order#	Bid Qty	Bid price	Ask price	Ask Qty	Order#
			32.2	300	B3
A2/E2	800/600	31.9			

13

New Order #13

#13	13:50:00	B	SIA	Sell	31.0	1000	B5	Accept

2 trades generated:

Order Buy	Trade	Order Sell
A2	800@31.9	B5
E2	200@31.9	B5

Book after update:

Order#	Bid Qty	Bid price	Ask price	Ask Qty	Order#
			32.2	300	B3

prices.
If choose match price = 31.9, no order can be matched
If choose match price = 32.0, 1000 can be matched (A1 match with E1)
If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

	E2	400	31.9			

Unmatched orders during continuous trading session will be carried to the closed auction.

3) Close auction:

#	Time	Client	Instrument	Side	Price	Quantity	OrderID	Result (not part of orders.csv – included here for explanation)
14	16:00:01	C	SIA	Sell	32.0	600	C4	Accept
15	16:01:00	B	SIA	Sell	32.1	100	B6	Accept
16	16:05:00	A	SIA	Buy	31.9	300	A3	Accept
17	16:09:59	E	SIA	Buy	31.8	2000	E3	Accept

Book at the closing time:

Order#	Bid Qty	Bid price	Ask price	Ask Qty	Order#
			32.2	300	B3
			32.1	100	B6
			32.0	600	C4
E2/A3	400/300	31.9			
E3	2000	31.8			

No match will happen, close price = null.

All trades in the day:

1000@32.1
500@32.1
4000@32.1
200@32.2
100@32.0
800@31.9
200@31.9

If choose match price = 31.9, no order can be matched
If choose match price = 32.0, 1000 can be matched (A1 match with E1)
If choose match price = 32.1, 1500 can be matched. (A1 match 1000 with E1, match 500 with B1)

4. **Consolidated Reports:**

i. Exchange report:

OrderID	RejectionReason
D1	REJECTED – MISMATCH CURRENCY
B2	REJECTED – INVALID LOT SIZE
C2	REJECTED – POSITION CHECK FAILED

ii. Client report:

Only list the instruments that client has traded. In this case, Client D doesn't have any trade on Instrument 0001, so it's not listed here.

ClientID	InstrumentID	NetPosition
A	SIA	2300
B	SIA	-5800
C	SIA	4300
E	SIA	-800

iii. Instrument Report:

#	Instrument ID	OpenPrice	ClosePrice	TotalVolume	VWAP	DayHigh	DayLow
1	SIA	32.1	NULL	6800	32.0721	32.2	31.9