

CodeIT Suisse 2022 Entry Challenge

Input

A comma-separated-value (CSV) stream of ticks in the format:

```
timestamp,ticker,quantity,price
```

For the purpose of this challenge:

1. timestamp will just be hh:mm, you can also treat this as a string.
2. price is any positive decimal value greater than 0.0, we only need to handle 1 decimal place for input and output.

Please state any other assumptions your solution makes. This could be in the form of code comments or a well-written README.md file.
:)

Part One

Aggregate the stream by time in chronological order, with each output record in the format:

```
timestamp,ticker1,cumulative_quantity1,cumulative_notional1,ticker2,cumulative_quantity2,cumulative_notional2...
```

Example: 00:00,A,5,5.5,B,4,4.4

The group ticker,cumulative_quantity,cumulative_notional repeats for each ticker with a tick at the timestamp. Tickers should be sorted alphabetically as well.

The `notional` is the product of `quantity` and `price` at each tick, and the `cumulative_notional` is the running sum of `notional` values for each ticker up till the timestamp.

The implementation should be done in a `to_cumulative` function (or equivalent) that takes in a list of `string`, and returns a list of `string`.

Part Two

Aggregate the stream by time in chronological order, but this time each output record is ‘delayed’ by only reporting cumulative quantities in blocks of `quantity_block`.

Take note that if only a portion of the current tick is applied for reporting the next quantity block, the `notional` calculation should factor the correct quantity, with the leftover quantity effectively ‘hidden’ from the true `cumulative_notional`.

The implementation should be done in a `to_cumulative_delayed` function (or equivalent) that takes in a list of `string` and `integer`, and returns a list of `string`.

Example:

`quantity_block: 5`

Input:

```
[
  "00:06,A,1,5.6",
  "00:05,A,1,5.6",
  "00:00,A,1,5.6",
  "00:02,A,1,5.6",
  "00:03,A,1,5.6",
  "00:04,A,1,5.6"
]
```

Result:

```
[  
  "00:05,A,5,28.0"  
]
```

"00:06,A,1,5.6" is not included in the output as the cumulative quantity is not a multiple of `quantity_block` which is 5 in this case.

Challenge Submission

Please fork from one the following templates to implement your challenge submission.

There is a simple `README.md` in each of them to explain what needs to be done.

- Python:
 - <https://replit.com/@astitemi/codeit-2022-entry-challenge-python>
- Java:
 - <https://replit.com/@astitemi/codeit-2022-entry-challenge-java>
- Javascript:
 - <https://replit.com/@astitemi/codeit-2022-entry-challenge-js>

We are using Repl.it's [unit testing](#) feature for evaluating your submission!

If you are using other languages, please feel free to consult the sample inputs in the templates and DIY. :)