EVENT

Is coding a part of your DNA?

Register now for CodeIT Suisse 2022

Inspiring possibilities

Endless potential.

That's empowered engineering.

As a leading global bank, our success is a direct result of our ability to adapt in an everchanging landscape. Technology is at the core of our business model.

We invite you to take part in CodeIT Suisse, a Coding Challenge designed to help you demonstrate your skills, personality and aptitude for technology.

CodeIT Suisse will give you the opportunity to not only solve coding challenges covering a wide range of topics, but also to set up an end-to-end server & development pipeline.

Event details

Date: September 17-18, 2022 (Saturday morning – Sunday afternoon)

Format: Hybrid (virtual & in-person)

All participants are invited to the Credit Suisse office on Sunday for event conclusion

Register

Prizes

Winning teams / individuals will be awarded with prizes that are worth of the below values.

	Team	Individual
Champion	HKD 12,500	HKD 4,000
First Runner Up	HKD 7,800	HKD 2,500
Second Runner Up	HKD 5,100	HKD 1,600

How to apply

1. To participate, you may apply on an individual basis or form a team of 2-3 eligible members.
2. Each participant (each team member and individual participant) is required to submit both an application via the **Register** button above and a **solution to the entry challenge on Repl.it**.

Entry challenge

You may refer to the attached CodeIT Suisse 2022 entry challenge and submission guide for instructions.

Registration deadline

Extended to September, 11, 2022, Sunday

Further event details will be sent to successful event applicants. We look forward to your entries.

Connect with us: campus.recruitment-ap@credit-suisse.com

To learn more about the Technology careers at Credit Suisse, please visit here

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!

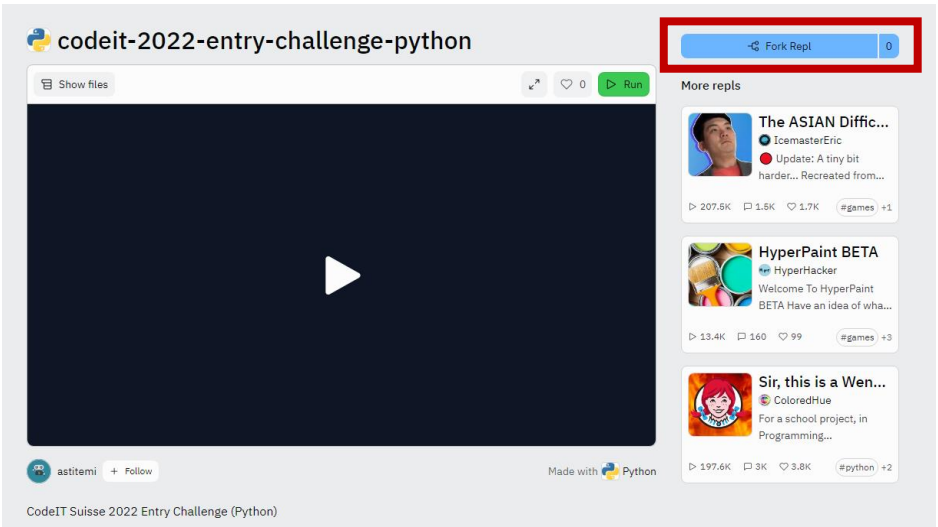
If you are using other languages, please feel free to consult the sample inputs in the templates and DIY. :)

How to submit your entry challenge solution using Repl.it?

1. Sign up for an account on Repl.it: https://replit.com/

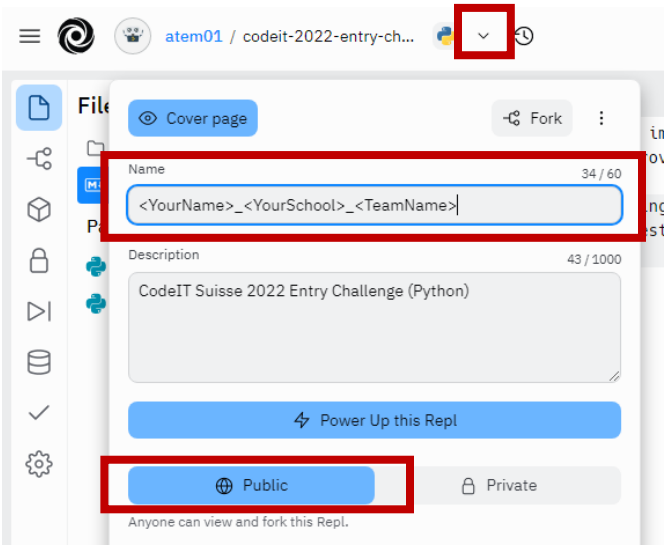
2. Fork from one of the following templates to implement your submission.
   - **Python**
     - https://replit.com/@astitemi/codeit-2022-entry-challenge-python
   - **Java**
     - https://replit.com/@astitemi/codeit-2022-entry-challenge-java
   - **Javascript**
     - https://replit.com/@astitemi/codeit-2022-entry-challenge-js

3. How to fork – the same steps apply to all templates:
   - Step 1: Open any of the template links above, click on "Fork Repl"

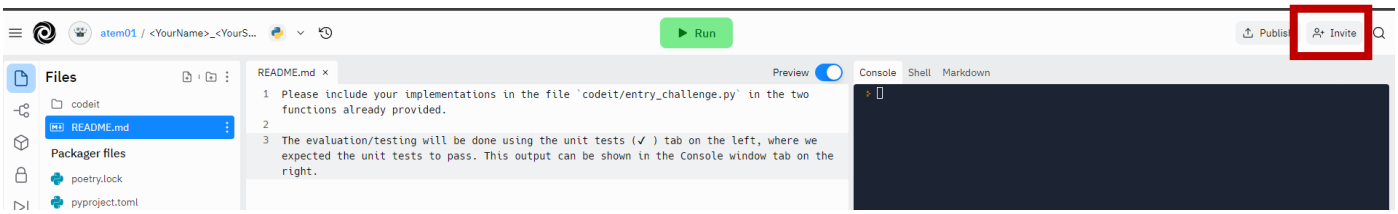


   - Step 2: Rename your repl.it
     - Click on the arrow at the top, beside the current repl's name
     - Ensure that the Repl is set to **Public**
     - In the pop-up window, update the repl name with the following naming conventions
       - If you are signing up as a team**: **YourName_YourSchool_TeamName**
       - If you are signing up as an individual: **YourName_YourSchool_Individual**

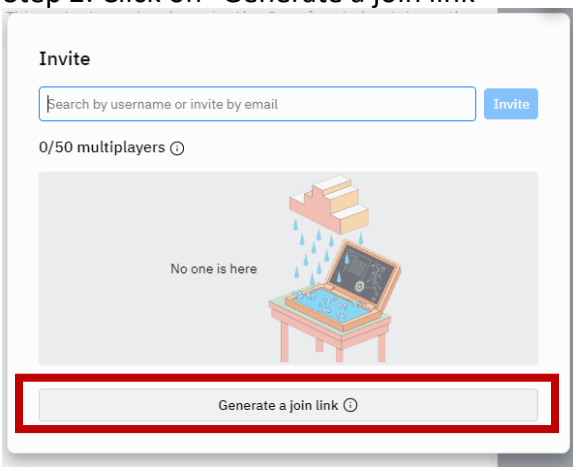
     ** all team members are required to submit an individual submission



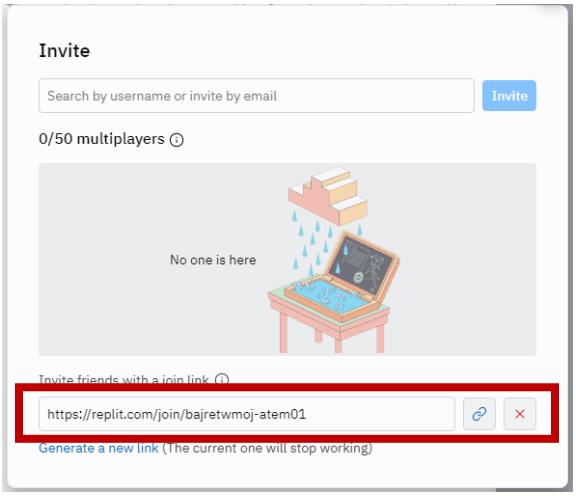
4. Generate link to the submission
    - Step 1: Click on “Invite” button at the top right-hand corner



- Step 2: Click on “Generate a join link”



- Step 3: The generated link is the link that you should submit as part of your application



5. The repl is automatically saved in “My repls”, you can continue working on it (ie: there’s no need to complete it at once)

6. To access your repl, navigate to “Profile” and click on the repl name.