The operation of the script can be divided into following points:

a) **Initialization:**

1. **Fetching currency exchange rates for EUR/PLN, USD/PLN, CHF/PLN for the last 30 days.**

   Retrieving demanded data utilizing https://api.nbp.pl/ website and storing them in pandas dataFrame. If fetching fails, user will be informed.

2. **Calculating EUR/USD, CHF/USD rates based on retrieved data.**

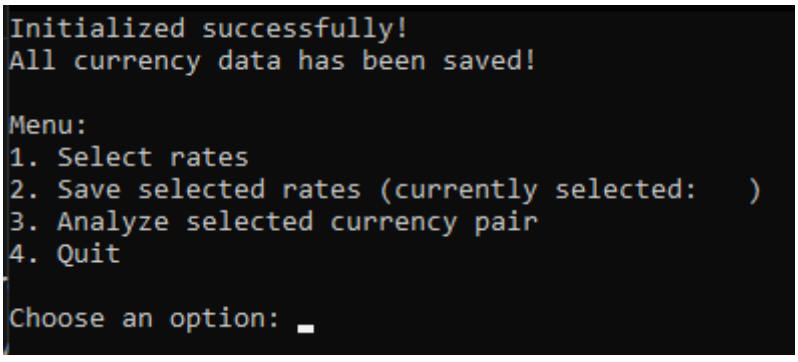
   Calculating EUR/USD and CHF/USD rates is done with simple operations on dataFrame columns.

3. **Saving dates and rates for five pairs into CSV file**

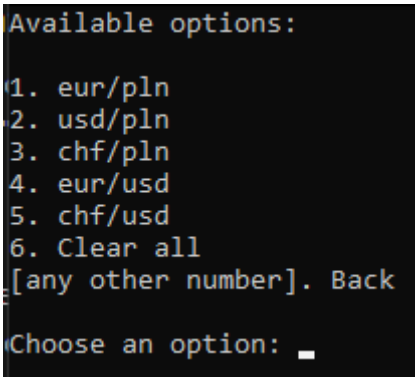
   When rates and dates are successfully fetched and calculated, they are being saved into „all_currency_data.csv".

b) **User interaction:**

User operates the program by choosing a number corresponding to each operation. Since such an approach requires valid user inputs, appropiate error-handling mechanisms were developed.



Main menu



Submenu (rates selection)

User can select multiple exchange rates and save them into "selected_currency_data.csv" file. There is also possibility to display statistical metrics such as: average, median, minimum, maximum.

Confirmation messages are being shown during every step of interaction with script.

c) Reflection

I decided to design this mini user-interface so that it is being operated by choosing an number corresponding to a number because data amount (in this particular case amount of distinct exchange rates) isn't high. Menus are being created dynamically, so there is a possibility to add a more rates quickly. Due to these reasons I believe that this approach is more user-friendly than for example typing currencies manually (user needs to be informed which currencies are available anyway). However, if the amount of data would be highly increased – there is a possibility, that current design would become slightly cluttered.