BSC – HGP- Assignment 01
Stock Trade Profit Calculator

Helpful Hints

Below are some hints to assist you with adding and interacting with widgets for Assignment 01.

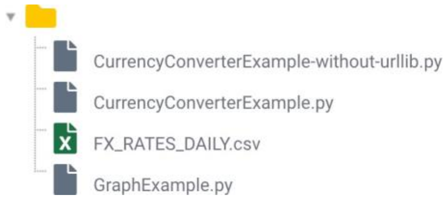
List of Steps

1. Download the required folders: StockTradeCalculator and Assignment 01 - Files
2. Review of the *Currency Converter Tutorial*
3. Complete StockTradeProfitCalculator.py
4. Submit to Moodle

Steps in detail

1. Download CurrencyConverter and Assignment01-Files

a) Download the folder *CurrencyConverter* which contains the files below:



CurrencyConverterExample-without-urllib.py
CurrencyConverterExample.py
FX_RATES_DAILY.csv
GraphExample.py

The functions of the files are as follows:

- *CurrencyConverterExample.py* – code from *Currency Converter Tutorial* with edits to:
    - Accommodate changes in the format of the original data source since the tutorial was published.
- *CurrencyConverterExample-without-urllib.py* – code from *Currency Converter Tutorial* with edits to:
    - Accommodate changes in the format of the original data source since the tutorial was published.
    - Allow the processing of a local file instead of a remote file.
- *FX_RATES_DAILY.csv* – contains the data used in the *Currency Converter Tutorial*
- *GraphExample.py* – contains some sample code showing how to create a graph which you might like to use as your additional feature.

b) Download the folder Assignment01-Files which contains the following files:

The functions of the files are as follows:

- *StockTradeProfitCalculator.py:* This is the file you will edit and submit as your answer.

- *BSC-HGP – Assignment 01 - Specification.pdf:* The assignment specification.

- *BSC-HGP - Assignment 01 - UI Design Document.docx:* The UI design document you will need to complete and submit.

- *Transformed_Stock_Market_Dataset.csv*: This is the dataset provided for your use in the assignment. It contains historical stock price data for companies like Amazon, Tesla, Google, and more, spanning multiple dates. You will use this dataset in your application to calculate the profit/loss based on user inputs.

2. Review of the Currency Converter Tutorial

a. Watch the video https://youtu.be/weKpTw1SjM4 (also on Moodle)

b. Follow the code *CurrencyConverterExample.py* (We are only interested in the GUI elements which feature at the top of the code not the data processing element which are at the bottom)

c. Identify the following sections in the code.

   i. Imports

   ii. Creation of GUI components

   iii. Layout of GUI components

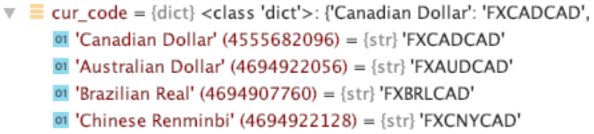
   iv. Signals linking changes in GUI components to *updateUI()*

   v. *updateUI()* which will

      1. get the information from each GUI component.
      2. perform a mathematical calculation.
      3. update a number of labels.

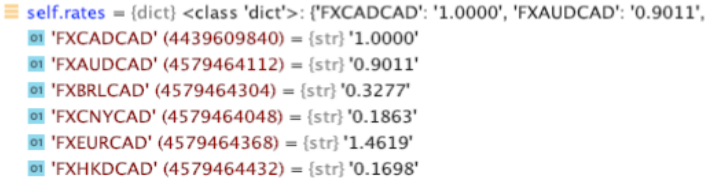
   vi. get_data() which creates 2 data dictionaries from *FX_RATES_DAILY.csv*. You **DO NOT** need to understand this section in detail. The dictionaries are called.

      1. self.cur_code



```
▼ ≡ cur_code = {dict} <class 'dict'>: {'Canadian Dollar': 'FXCADCAD',
    01 'Canadian Dollar' (4555682096) = {str} 'FXCADCAD'
    01 'Australian Dollar' (4694922056) = {str} 'FXAUDCAD'
    01 'Brazilian Real' (4694907760) = {str} 'FXBRLCAD'
    01 'Chinese Renminbi' (4694922128) = {str} 'FXCNYCAD'
```

      2. self.rates



```
≡ self.rates = {dict} <class 'dict'>: {'FXCADCAD': '1.0000', 'FXAUDCAD': '0.9011',
   01 'FXCADCAD' (4439609840) = {str} '1.0000'
   01 'FXAUDCAD' (4579464112) = {str} '0.9011'
   01 'FXBRLCAD' (4579464304) = {str} '0.3277'
   01 'FXCNYCAD' (4579464048) = {str} '0.1863'
   01 'FXEURCAD' (4579464368) = {str} '1.4619'
   01 'FXHKDCAD' (4579464432) = {str} '0.1698'
```

3. Complete Assignment1 *StockTradeCalculator.py* by doing the following:

a. Identifying similarities between the assignment requirements and *CurrencyConverterExample.py*

b. Read the comments in *StockTradeCalculator.py* and ask any questions you have regarding the content.

c. Follow the comments to do the following.

   i. Create widgets – like tutorial, code for QCalendarWidgets is available at the bottom of https://zetcode.com/pyqt6/widgets/

   ii. Layout widgets – code available at bottom of https://zetcode.com/pyqt6/layout/

   iii. Set the purchase and sell calendars default dates inside the date ranges so that there are valid prices. The following code would allow you to make a QDate object to hold the newest date in the dataset and then set the default date for the sell calendar widget.

```
self.sellCalendarDefaultDate = sorted(self.data['Amazon'].keys())[-1]
self.sellCalendar.setSelectedDate(self.sellCalendarDefaultDate)
```

   iv. Link changes in GUI components to updateUI().

   v. Complete *updateUI* which will

   - Get the information from each GUI component (selected stock, dates, and quantity).
   - Perform a mathematical calculation for profit/loss.
   - Update the number of labels.

   vi. At the base of the file are 3 completed data processing methods. You **DO NOT** have to edit these as they are complete.

d. Add additional features. Your additional feature should focus on GUI interaction rather than complex data processing as this is GUI & HCI course not a data analytics course. The following resources may assist you.

   i. https://zetcode.com/pyqt6/widgets/ tutorial - how to add the following widgets to your GUI
      1. QCheckBox
      2. QToggleButton
      3. QSlider
      4. QProgressBar
      5. QCalendarWidget

   ii. https://doc.qt.io/qt-6/qtwidgets-index.html shows many additional widgets of interest

   iii. https://zetcode.com/pyqt6/painting/ shows how to draw items in the background if you like

   iv. GraphExample.py as a basis for adding an additional feature

   v. Remember to document your Extra Feature both in code using comments as well as in your UI Design document.

   vi. The Extra Feature is worth 30 marks, make sure to implement a feature with enough complexity to receive these marks. If you would like, you can choose two medium complexity features, but make sure to document both well.

4. Submit to Moodle

a. The final submission zip should contain the following files
   i. StockTradeProfitCalculator.py
   ii. UI Design Document
   iii. dataset.csv
   iv. Any assets in a folder titled 'Assets'

b. Rename the folder FirstName_LastName_StudentNumber_Ass1 appropriately

c. Complete BSC-HGP - Assignment 01 - UI Design Document and save as a PDF

d. Zip the folder and upload it to Moodle.