

## Programming Group Assignment: Trading Platform

### **Coders:**

Gian Berwert (19-606-441)

Fabian Welti (20-611-729)

Lukas Gebhardt (20-610-986)

### **Objective of the group project:**

This assignment focuses on designing and implementing a program, which uses Python to solve the problem described below.

### **Assignment specification:**

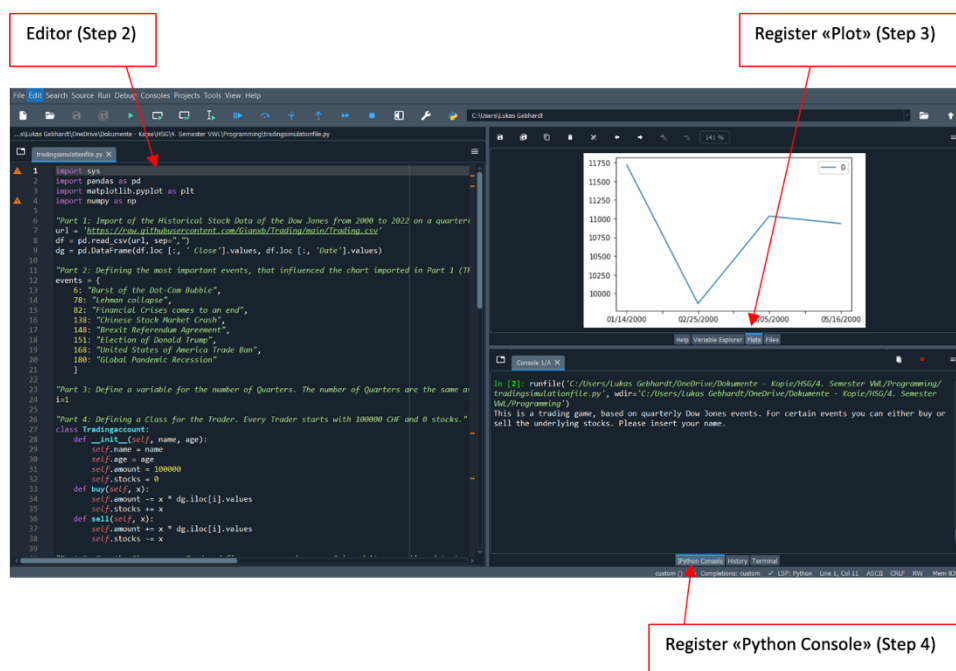
This program is designed to be a trading platform. The user will have the chance to follow real price movements from the Dow Jones Industrial Average (quarterly basis, from 2000 to today), visualized in a graph. Additionally, the user will have the chance to execute simple buy and sell trades at the most influencing events of the market (e.g. Financial crisis 2008). For these events every user is able to trade with 100'000 CHF.

## Programming Group Assignment: Trading Platform

### Step by step guide how to execute the code:

This section provides a step-by-step guide from where to get the code and how to execute it so the trading platform can be used in the most convenient way. After the written step by step manual, we also provide a link to a video executing the steps.

1. Use the following link  
<https://github.com/Gianxb/Trading/blob/a83f5ca3d86ea2367296b1b4351d8bf117c0805b/temp.py> which opens a Github folder where the raw code can be pulled
2. Open Spyder where the pulled code can be pasted in the editor
3. Go to the console in Spyder and click on the register named “plot” (see picture below)
4. Click on the register named “python console”
5. One can run the code via the “python console” and therefore start the trading platform
6. The first step is to choose and enter a trading name
7. The second step will be to insert your age
8. If the (age is over 18) one is able to trade:  
For a given event effecting the market decide if you want to buy, sell, or hold your stocks.
9. If the entire time period is over and all the events are done, one can read the results made from his trading activities
10. One can improve his trading skills by repeating the steps from step 4 onwards



The following link provides a video version of how to successfully run the code:

<https://github.com/Gianxb/Trading/blob/b3cc8645413f8288b6c6c400b0e910b44075a6f8/Tradinginstruction.mp4>

### How the code works:

First, we imported the needed packages to run our program. The panda package and matplotlib.pyplot played a very important role in data manipulation and visualizing the price movements.

We split our code in 10 parts, so it is easier to follow and to understand.

#### Part 1

In this step the aim is to import the historical stock data of the Dow Jones from 2000 until 2022 on a quarterly basis. This is done by pulling the data from Github, containing a spreadsheet with Dow Jones price movements (<https://www.wsj.com/market-data/quotes/index/DJIA/historical-prices>).

#### Part 2

This part defines major events in a dictionary which influenced prices of the Dow Jones Industrial Average in the period from 2000 until 2022, which match the data we imported in part 1. The number which the event is attributed to represents the corresponding quarter of the year (e.g. the number 6 represents the 6<sup>th</sup> quarter, so 2<sup>nd</sup> quarter of the 2<sup>nd</sup> year).

#### Part 3

This step defines the variable i for the number of quarters. The number of quarters are the same as in the dictionary from part 2.

#### Part 4

The program defines class for the trader. This class contains all the functions that are linked to the trader. The first function initiates the name, age (checked to be an integer), starting capital fixed for every trader at 100'000 CHF and a fixed starting amount of stock at 0. The second function defines the buy trade: It subtracts the amount of stocks to buy, times the current value of the stock from the capital and adds the amount of stocks to the stock account. The third function defines the sell trade, which works the opposite way as the buy function.

#### Part 5

In part 5 the program asks the trader for input, specifically the name and the age.

#### Part 6

Part 6 checks with a condition if you are over 18 and thus allowed to trade or not. Traders below the age of 18 cannot trade and receive the message: "By law underage people are to be protected. You are not allowed to trade!". When the if condition is fulfilled, the trader gets a greeting and the trading starts.

#### Part 7

This part is concerned with visualizing the price movements. The for loop generates a moving graph (a new date is added to the graph every 0.0001 second).

## Programming Group Assignment: Trading Platform

### Part 8

Now the interactive part really begins. First the trader gets information about which event occurs and decides to hold, sell or buy stock. If the trader answers hold then 1 is added to j and the while loop is over. If the trader types in buy or sell, then the trader has to enter the amount of stocks he wants to trade (only non-negative numbers possible). The trade only executes if the trader has enough money or stocks in his account respectively. If this condition is exceeded, a message will be displayed that informs the trader about the maximum amount.

### Part 9

This part is very important when it comes to the last date in the data. When this date is reached, all remaining stock are sold automatically at the current market price and transferred to the account.

### Part 10

In this final step the trader is presented with the returns of his trades.

Have fun and enjoy the trading platform!