Optiver

Optibook Guide

2025

Table of Contents

1 Introduction

Optibook is a fully functional virtual exchange, built in-house at Optiver for educational purposes. Through Optibook, we teach about financial markets, exchanges, market making and derivatives, and aim to show why market makers are essential for the functioning of (financial) trading and with that the economy as a whole.

In the trading challenges we will supply, you will develop automated trading algorithms to trade on the Optibook exchange. Each of the these is intended to demonstrate a specific aspect important in automated trading. Depending on the design and time-availability of your course, you may trade all, or a subset of the products available on the exchange.

The Instrument reference section of this document contains an overview of all the tradable products listed on the exchange, and you can read these definitions as they become relevant.

The questions and challenges for Optibook typically do not have a single correct answer. As in real financial markets, your algorithm will have to deal with the uncertain and unpredictable behaviour of other market participants, your fellow participants and a suite of trading bots. So, your algorithms ideally adjust easily to the situation at hand. Rather than trying to derive the optimal algorithm once-and-for-all, the algorithms should be built up by a combination of first principles and what-if style thinking. An algorithm you develop will be an engineering project, with many moving parts, all working in tandem. Choices you make in one part will relate to what the optimal choices in another are.

This text is usually, depending on the course setup, accompanied by at least one introduction lecture, where you will be guided through the process of logging in to and interacting with the exchange for the first time. Additional reference materials, such as recorded lectures of financial concepts, API documentation, etc. can be found on the Optibook webpage hosted on https://<your-course>.optibook.net/, under *docs* and *resources*. If you are presented with an Optibook Intro Lecture, that supplants the *setup instructions* and *intro exercises*, so you can safely skip over those sections of the page.

2 Programming Environment

Algorithms that trade on Optibook are written in Python. Python is an interpreted programming language. This means that a Python program is defined in a text file, which is read and executed by a Python interpreter.

To avoid any differences between setups, we have opted for participants of Optibook courses and challenges to write and run their code through a cloud-based setup hosted on Amazon Web Services (AWS). What this provides is an environment that is correctly setup from the first time you launch it, offering a pre-installed version of Python, a browser-based IDE (Visual Studio) supporting .py files and Jupyter notebooks, and the ability for us to preload the Optibook libraries and any example code files.

For simplicity, we ask participants to use this web-hosted service from a personal laptop or PC with a public/ordinary internet connection, as some university and/or company networks disallow certain required internet traffic to AWS. It is highly recommended to use a Chrome browser for access.

You will be provided with a login username and password by one of the course hosts.

2.1 Cloud-Based Development Environment (DEVENV)

2.1.1 Logging In

To log in to the development environment (devenv) browse to <your-course>.optibook.net. Then:

1. Click on the **devenv** button on the top-right of the page
2. On the page, login using the provided details
3. Click '*Start devenv!*' and refresh after approximately 1 minute until it's started up
4. Click the link '*Go to your devenv*'
5. (If you are too fast this may show while the server is loading, in sequence a 503 error, a 502 error, the IDE, stubbornly refresh until the Visual Studio IDE shows up)
6. When prompted, select Yes to trusting all files
7. Get coding!

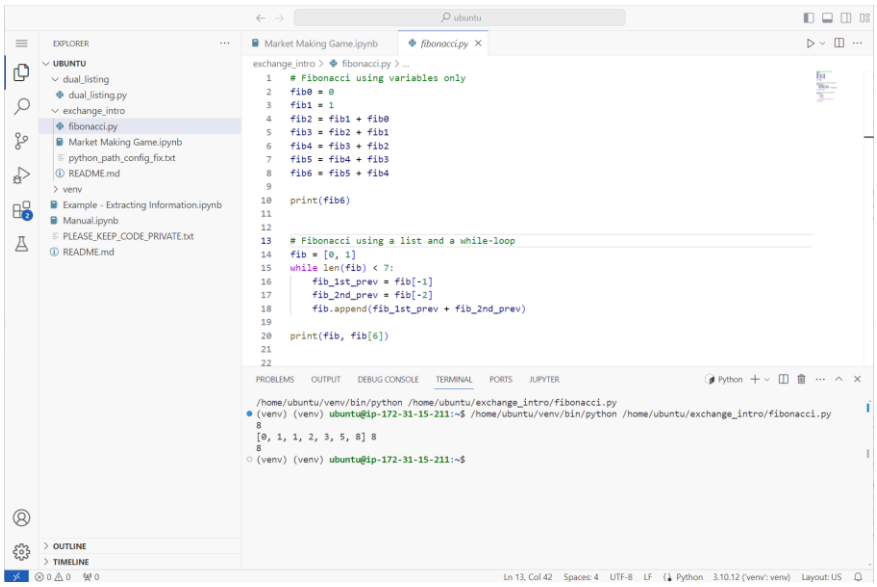
Troubleshooting login process:

If you are not able to log in, it is typically because of one of the following causes:

- You browsed to the wrong server URL, this should be <your-course>.optibook.net and not dev.optibook.net
- You copied the password incorrectly, for example including a trailing space

2.1.2 Visual Studio Usage

Below is a screenshot of the Visual Studio IDE (Integrated Development Environment).



The large area in the top-right is where text and code are edited. If the tab that is currently selected in the main area is a .py file, it can be ran directly by hitting the “play” button located at the top of the screen. The output of the file is then displayed at the bottom of the screen.

The column on the left is a file browser, which should contain the specific trading challenge files relevant for your course or challenge. Double clicking a file opens it as a tab in the code editor.

Feel free to make use of any other functionality supported in the IDE. However, please do not use public git repositories for sharing solutions, or share them online otherwise, to keep the challenges interesting for future participants.

2.1.3 Jupyter Notebooks

Jupyter notebooks are supported directly from the IDE. The *Manual.ipynb* notebook is a good starting point, it contains examples of all interactions that are possible to have with the Optibook exchange through the Exchange object.

When first running a notebook it is required to configure which version of Python (*kernel*) is used. To do so:

- Open a Jupyter notebook
- Wait until the loading icon on the top right of the notebook, ‘Loading kernels...’, is replaced by ‘Select Kernel’
- Click ‘Select Kernel’
- Above the notebook, in the centre, there is now a list of options to select, select ‘*Python environments...*’ followed by ‘*venv*’

Your notebook is now configured correctly and you’re able to run the cells as normal. Visual Studio should remember which Kernel/Python version you selected for any future notebooks you open, but if not, simply repeat the above.

3 Instrument reference

This section contains a full reference of the products available for use as you start working through the challenges. Depending on your course setup, it may not be necessary to have a full understanding of all the instruments listed; so look up specific instrument types as convenient and relevant.

On the Optibook Visualizer, the Instrument Overview shows all *listed instruments*, or, in other words, the tradeable products. We refer to these instruments by their unique *instrument_id*. The instrument_ids always have a specific format which can be interpreted to understand what specific instrument or derivative it refers to, for example, the expiry date or strike of an option. But, this information can also be obtained directly through code. An example of how to do so is provided in the *'Example – Extracting Information.ipynb'* notebook in the home folder of your instance.

3.1 Stocks and stock-derived instruments

3.1.1 Stocks

The instrument_ids of the stocks take the form of only a stock ticker/name: **<stock ticker>** (TSLA, NVDA, ASML, etc). They are so-called *cash instruments*, in that they are not a derivative but represent ownership of a single share of a stock.

As is the case for all instruments on Optibook, the stocks can be *sold short*. That is, one is allowed to have a negative position in them, you can sell before you buy.

The real-world process of selling a stock short is an involved process, whereby one borrows the stock from someone else first to return later. On Optibook, this is all happily abstracted away, we can simply enter into a negative position; leading to the opposite profit-and-loss (*PnL*) as a positive (*long*) position would.

The stock names are fictious, their values are not in any way related to the real-world equivalents.

3.1.2 Stock dual listings

A *dual listing* is the *listing* of any security on two different exchanges. That means that the same instrument, in the same currency, is traded in two different locations, for example both in Paris and Amsterdam. As they represent the exact same rights of ownership; it follows that they should have the same value. However, due to the disjoint nature of the exchanges; the available prices might (temporarily) deviate. The strategies surrounding these interesting discrepancies are explored in the first trading challenge.

Our simulated exchange Optibook is a single exchange in a single location, listing different instruments. To simulate the dual listings on Optibook we introduce a second instrument on some of the stocks, with the instrument_id **<stock ticker>_DUAL**. These dual listings are very similar to the dual listings on real exchanges, in that they represent the same stock, but with disjoint order books, leading to the same interesting dynamics between them.

For dual listings on real exchanges it is usually possible to transfer a stock holding from one exchange to the other. This ability completes the *arbitrage*, so to speak, as one might buy a stock on one exchange, transfer it, and sell it at the other. This is not possible on Optibook, but you may assume it to be possible for other market participants/the broader market. That is, you may assume that the two products are truly perfectly *fungible* and should have/will converge to the same price.

3.1.3 Stock futures

The stock futures have instrument_ids of the form **<stock ticker>_<year><month>_F**; the year and month refer to the expiry date of the future. By convention the expiry always takes place on the third Friday of the month at 12:00 UTC.

A *futures contract* is a type of exchange-traded financial derivative. It is a financially engineered product, the value of which is derived from an underlying product. It is designed with a later-date purchase of the underlying stock in mind, to ensure that through the set of cashflows it provides, one can, in a specific sense, "lock in the future price of the stock".

You may have heard of a so-called margining and netting procedure, related to managing the available funds of a future trading market participant. On Optibook, this process is simplified. All the futures contracts are modelled as simple assets with the direct pricing relationship valid for futures from theory: $F_t = S_t \cdot exp(r\tau)$ for some interest rate r, time to expiry τ (in years) and underlying stock price S_t.

3.1.4 Stock Options

Like stock futures, *stock options* are exchange-traded financial derivatives. They give a conditional right to buy (*call options*) or sell (*put options*) the underlying stock at a specific agreed upon *strike price*, a right which the option buyer may then choose to exercise or not. Determining the correct price of an option is involved and, if relevant, will be covered by the materials and/or lectures related to the trading challenge.

Options may be European-style or American-style, which has nothing to do with the physical location it is traded in, but rather with the expiry procedure. European-style options may be exercised once, exactly when the options expire, whereas American-style options may be exercised at any time before the expiry. All options on Optibook are European-style.

The stock options have instrument_ids of the form **<stock ticker>_<year><month>_<strike price><call/put>**. By convention the expiry always takes place on the third Friday of the month at 12:00 UTC.

3.2 Index-derived instruments

Individual stocks are often combined into a basket to create so-called *stock indices*. An *index value* is calculated as the weighted average of the *constituent stock* values, for some pre-decided weighting scheme which can vary per index. Tracking the value of these indices allows investors to focus on broad market developments, e.g. in specific sectors or countries, rather than the idiosyncrasies of each individual companies' performance. Some examples of indices are the S&P

500 (USA), NASDAQ (USA), the EURO STOXX 50 (Eurozone), the CAC 40 (France), the DAX (Germany) and the AEX (The Netherlands).

There are a few common weighting schemes for indices, some more appropriate than others. They are typically a variation of a price-weighting, or a market capitalization-weighting. Capitalization-weighting is preferred, as the price of a stock is not very representative of its importance (compare a stock worth €100,- with 1 million outstanding shares to a stock worth €10,- with 100 million outstanding shares).

It is crucial to understand that indices themselves are not traded directly. The index value is only a number that is defined, calculated and officially published by some financial institution, often an exchange. For investors it is possible to get a direct monetary exposure to this number either by buying the stocks in the right proportion to their weighting (inefficient), or by trading one of the index-derived products (efficient).

Optibook defines one stock index, the OB5X, it is a value-weighted index with the following weighting scheme:

$$X_t = \frac{\sum_{i=1}^{5} w_i \cdot S_{i,t}}{1000}$$

Here X_t represents the index value at time t, w_i represents the weight of stock price i based on the fictional market capitalization, $S_{i,t}$ represents stock i's value at time t and 1000 is the normalizer/divisor. For simplicity, the weighting factors w_i are kept constant throughout the simulation at the following values:

	ASML	AAPL	SAP	TSLA	NVDA
Weight	908.06	129.24	124.78	2245.39	953.21

3.2.1 Index ETF

An *ETF* is an *exchange-traded fund*. It is, at the basis, just the stock of a publicly listed company. However, the company it represents is not like any ordinary one. It does not do business and distribute the proceeds to the shareholders. Rather, the company is incorporated for a single purpose: to hold balance sheet assets (stocks) in the appropriate ratios to track a pre-defined financial index.

The issue with manually tracking an index like the S&P 500 is that one needs to buy and hold all the 500 different stocks, in the appropriate ratios. This is firstly quite cumbersome and inefficient; one incurs trading costs for each individual transaction in the stocks and would need to rebalance continuously as the ratios change. Secondly, to remain accurate, it is not possible to hold very small amounts of each stock, as it is not possible to hold fractional stocks. If one would need to hold two stocks in a ratio of 30:1, there is no other option than to buy 30 of the second stock for every single one of the first. All in all this exposure may be very large.

These issues are resolved through the concept of an ETF. The ETF issuer manages the trading and rebalancing of the stocks in the correct ratios, and the interested market participant can get (or remove) exposure to the underlying index through a single trade in the ETF stock.

As the ETF is a company, and it needs to rebalance periodically, and as participants buy in and sell out of the fund, it may sometimes be left with a (small) cash position in addition to the stocks it holds, representing the money it has left over from its transactions. Therefore, we can describe the value of the ETF with the following relationship:

$$V_t = NAV_t = C_t + M \cdot X_t,$$

where NAV_t stands for the net-asset value at time t, C_t represents the, potentially time-varying cash component, M represents a fixed multiplier, consistent with the number of outstanding shares relative to the ETF holdings, and X_t represents the time-t index value. In other words, it's simply worth the sum of all of its assets, including the cash holding and replicated index exposure. Optibook lists an ETF on OB5X with instrument_id **OB5X_ETF**; it has multiplier $M = 0.25$ and a time-constant cash component $C_t = C = 2.50$.

3.2.2 Index futures

Index futures are defined analogously to stock futures, with a value of $F_t = X_t \cdot exp(r\tau)$, with X_t now representing the time-t index value in place of the stock value. The instrument_ids have the form **OB5X_<year><month>_F**, and as in the other instruments, by convention, they expire on the third Friday of that month, at 12:00 UTC.

3.2.3 Index options

Index options are defined analogously to the stock options. The pricing is very similar to that of stock options, and details of that will be covered in the relevant lectures. The instrument_ids take the form **OB5X_<year><month>_<strike price><call/put>**. The options are European-style. As in the other instruments, by convention, they expire on the third Friday of that month, at 12:00 UTC.

4 Exchange-enforced limits

The Optibook exchange imposes some exchange-side limits for your algorithm to adhere to:

- Total position (long or short) per instrument cannot ever go over 100
- You are not allowed to have total orders outstanding for over 200 lots per instrument
- You are only allowed to send 25 updates (inserts/deletes/amends) to the market per second

These limits are applied pre-emptively. For an inserted order, if there is any worst-case scenario of order execution conceivable, by matching with any current and/or future opposing orders, that may lead to breaching the limit, the order is rejected by the exchange, and the participant/algorithm directly disconnected.

On disconnect, all outstanding orders are directly deleted.

While you may simply reconnect, it is best to stay in full control of your algorithm to avoid breaching these limits in the first place. Insert timers to avoid breaching the frequency limit, and ensure your outstanding order volume is capped to a value that does not cause breaches of the position and outstanding order volume limits.

There is also a (soft) delta risk limit. If you have an unhedged position of 100 or more deltas for more than 10 seconds, then 10% of these deltas will be hacked out automatically by Optibook against an unfavorable price of 2% worse than the midprice in the orderbook.