

Хабр



КАК СТАТЬ АВТОРОМ



Зарплаты IT-специалистов



Войти



empenoso

25 мар в 05:20

# Поиск ликвидных облигаций с использованием Python

Простой

5 мин

15K

Python\*, Проектирование API\*, Open source\*, Финансы в IT

Кейс

Сезон Open source

На Московской бирже торгуется более 2500 облигаций, но большая часть из них неликвидна - в стакане почти нет предложений и сделок совершается крайне мало. Это затрудняет покупку и продажу таких бумаг. При этом известные мне публичные сервисы не суммируют объемы торгов за период, поэтому сложно быстро найти облигации с высокой ликвидностью.

Пять лет назад написал Node.js-скрипт, затем адаптировал его для Google Таблиц, а теперь разрабатываю Python версию. При помощи сообщества на GitHub эта Python версия идёт к созданию полноценной библиотеки с расширенными возможностями: автоматический поиск ликвидных облигаций, расчет денежных потоков, сбор новостей по эмитентам и вычисление оптимального объема покупки. Все это направлено на помощь простым инвесторам, вроде нас с вами, чтобы оперативно находить выгодные инвестиционные инструменты и принимать решения на основе актуальной информации.

РЕКЛАМА

**Аукцион серверов**

Цены падают каждую секунду

```

18 class MOEX:
216 def search_volume(self, security_id: str, threshold_value: int) -> dict[str, int]:
220     # в этот день и день перед
227 board_id = self.board_id(security_id)
228 if not board_id:
229     self.log.info(
230         f"⚠ Не удалось получить board_id для {security_id}. Поиск объема прерван."
231     )
232     return {"low_liquid": 1, "value": 0}
233
234 url = (
235     f"https://iss.moex.com/iss/history/engines/stock/markets/bonds/boards/{board_id}/securities/{sec"
236     f"iss.meta=off&iss.only=history&history.columns=SECID,TRADEDATE,VOLUME,NUMTRADES&limit=20&from={
237 }
238 # numtrades - Минимальное количество сделок с бумагой
239 # VOLUME - оборот в количестве бумаг (Объем сделок, шт)
240 self.log.info(
241     f"🔗 {foo_name}. Ссылка для поиска объема сделок {security_id}: {url}"
242 )
243 try:
244     time.sleep(self.API_DELAY)
245
246     response = requests.get(url)
247     response.raise_for_status()
248     json_data = response.json()
249     history_data = json_data["history"]["data"]
250
251     count = len(history_data)
252     volume_sum = 0
253     low_liquid = 0
254     for i in range(count):
255         volume = history_data[i][2]
256         volume_sum += volume
257         if threshold_value > volume: # если оборот в конкретный день меньше
258             low_liquid = 1

```

## Критерии выбора ликвидных облигаций на Московской Бирже

Ликвидность это один из ключевых параметров, поскольку даже высокодоходная бумага бесполезна, если её невозможно купить. В моём скрипте для поиска облигаций используются несколько основных критериев:

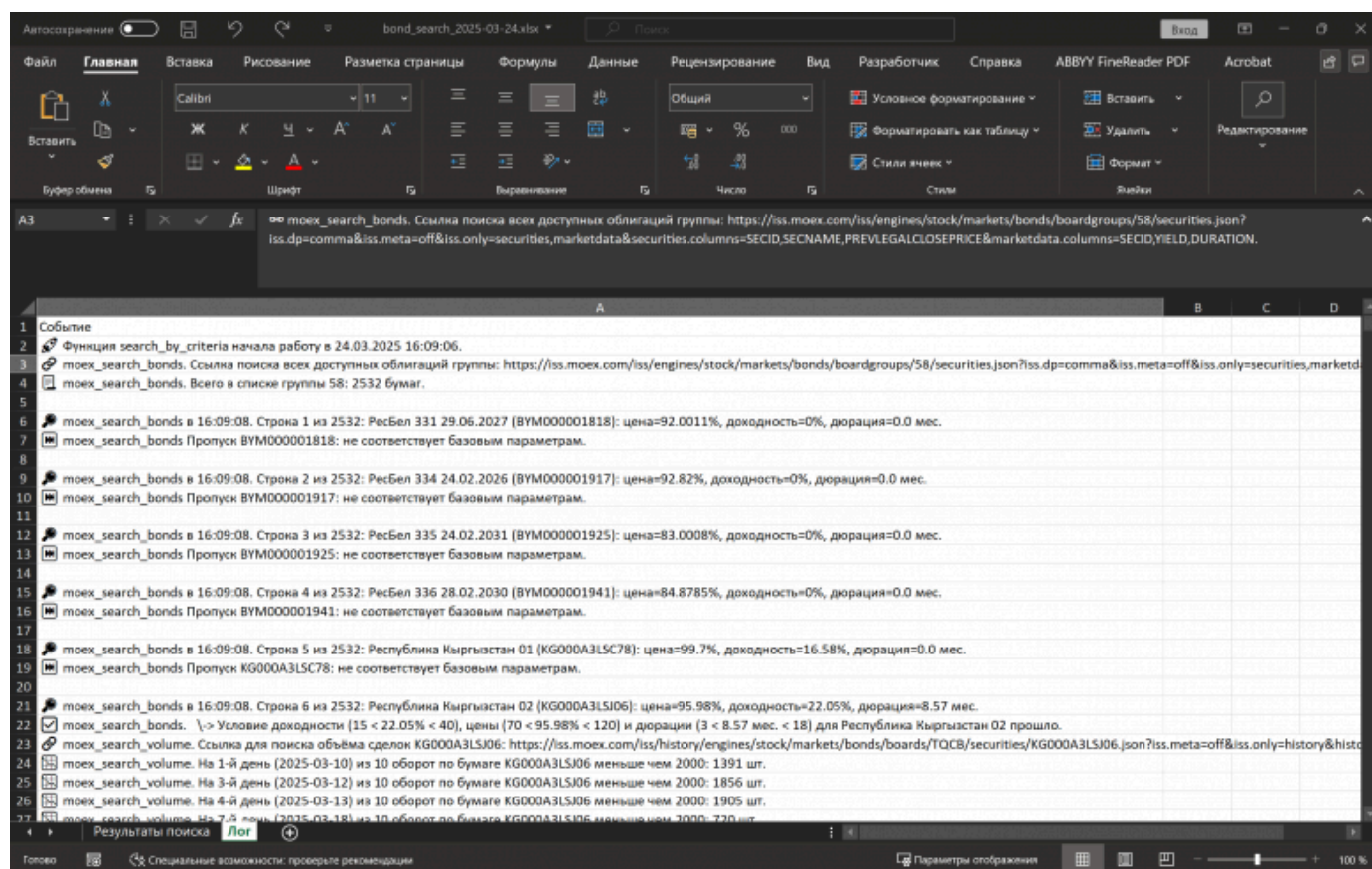
### Доходность

Эффективная доходность облигации — один из главных параметров. В фильтре задаётся диапазон, например, от 15% до 30%. Важно учитывать, что этот показатель не включает налог с купонов и комиссии брокера.



**Аукцион серверов**

Цены падают каждую секунду



## Текущая цена

Облигации торгуются по разным ценам относительно номинала, все цены облигаций указаны в процентах, и этот параметр позволяет фактически выбрать стратегию получения дохода:

- Если облигация торгуется сильно ниже номинала, основная доходность будет сформирована к моменту погашения (за счёт разницы между ценой покупки и номиналом).
- Если облигация торгуется близко к номиналу, основная доходность складывается из купонных выплат в течение срока жизни.



**Аукцион серверов**

Цены падают каждую секунду

**Пример:** облигация ЕвроТранс БО-001Р-03 на 10 марта 2025 года (код RU000A1061K1, [ссылка](#)):

- Текущая цена: 86% от номинала
- Купонная доходность: 13,6% годовых
- Доходность к погашению: 26,78% годовых. Доходность к погашению предполагает, что вы держите облигацию до погашения и что все купонные выплаты будут произведены в срок.

Откуда такая разница? Дело в том, что облигация сейчас торгуется ниже номинала, а при погашении инвестор получит 100% номинальной стоимости. То есть, кроме купонов, инвестор дополнительно зарабатывает на разнице в цене. Именно поэтому параметр текущей цены помогает выбрать, когда получать основную доходность — постепенно в течение срока или разово в момент погашения.

## Дюрация

Показатель дюрации позволяет выбрать облигации с нужным сроком жизни. Например, если мне нужна бумага на ближайшие 3–18 месяцев, фильтр исключает слишком



**Аукцион серверов**

Цены падают каждую секунду

## Прозрачность выплат

Наличие полной информации о будущих купонных выплатах или наличие оферты.

Также я исключаю флоатеры, поскольку Московская биржа не передаёт по ним данные о будущих платежах.

## Ликвидность

Ликвидность - основной критерий, ради которого создавался этот инструмент. В скрипте анализируются:

- Минимальное число сделок за каждый из последних 15 дней - чтобы исключить облигации, которые могут внезапно «замереть».
- Общий объём сделок за 15 дней - параметр, который позволяет выявлять бумаги с устойчивым спросом. Этот скрипт позволяет гибко подстраивать фильтры и находить действительно ликвидные облигации, подходящие под конкретно Вашу стратегию инвестирования.

## Как работает скрипт

Скрипт использует API Московской биржи для получения актуальных данных об облигациях. Данные скачиваются для ознакомительных целей и это позволяет оперативно находить ликвидные облигации.



**Аукцион серверов**

Цены падают каждую секунду

**Ограничение запросов.** С сентября 2024 года API Московской биржи начало периодически разрывать соединение без объяснения причин. Это продолжалось около полугода, поэтому в коде был установлен лимит — не более 50 запросов в минуту. Сейчас, в марте 2025, эта проблема больше не наблюдается, но ограничение оставлено для стабильности.

Московская биржа периодически меняет формат данных, что требует оперативного обновления скрипта.

## Пошаговое руководство по запуску

Если вы не разбираетесь в программировании, но хотите воспользоваться этим Python-скриптом, следуйте инструкции:

### Шаг 1. Скачайте скрипт

1. Откройте ссылку: [GitHub проекта](#).
2. Нажмите "Code" → "Download ZIP".
3. Разархивируйте ZIP в удобную папку.



**Аукцион серверов**

Цены падают каждую секунду

Если Python не установлен:

- Windows: скачайте и установите [Python с официального сайта](#). В установке отметьте "Add Python to PATH".
- MacOS: скачайте [Python для macOS](#) и установите.

### Шаг 3. Установите зависимости проекта

1. Откройте папку с проектом.
2. Дважды кликните файл `install_requirements.bat` (Windows) или `install_requirements.command` (MacOS).

### Шаг 4. Запустите скрипт

Дважды кликните файл `1_bonds_search by criteria.py`.

Во время выполнения отображается лог выполнения.



**Аукцион серверов**

Цены падают каждую секунду

Будет создан файл с текущей датой: `bond_search_2025-03-25.xlsx`

## Преимущества open source и шаги к Python-библиотеке

Идея - набор из четырёх скриптов для личного использования. Разработал их как частный инвестор, понимая какие задачи стоят передо мной:

1. Поиск ликвидных облигаций
2. Автоматический расчёт денежных потоков
3. Сбор новостей по эмитентам
4. Расчёт оптимального объёма покупки

### От одиночного скрипта к полноценной библиотеке

Так как проект открыт, к нему подключилось сообщество. Одним из первых с pull request пришёл Imasikl, который заинтересовался темой облигаций и предложил преобразовать набор скриптов в полноценную Python-библиотеку. Уже было внесено множество улучшений:

Метки: [Python](#), [Финансы](#)



**Аукцион серверов**

Цены падают каждую секунду



- #11 – Добавлена диаграмма взаимодействия пользователя со скриптом (автор: lmasikl, одобрено 17 марта)
- #7 – Отформатирован код с дефолтными настройками Ruff (автор: lmasikl, одобрено 14 марта)
- #6 – Внесен в приложение скрипт расчета оптимального объема покупки облигаций (автор: lmasikl, одобрено 13 марта)
- #5 – Перенесен в приложение скрипт поиска новостей (автор: lmasikl, одобрено 12 марта)
- #4 – Перенесен в приложение скрипт поиска облигаций (автор: lmasikl, одобрено 10 марта)
- #3 – Начат переход к полноценной библиотеке (автор: lmasikl, одобрено 6 марта)

## Февраль 2025

- #2 – Исправлены некорректные значения в рублях (автор: gogbajbobo, одобрено 21 февраля)
- #1 – Выполнен рефакторинг кода (автор: lmasikl, одобрено 20 февраля)

Добавлен план схемы работы:



**Аукцион серверов**

Цены падают каждую секунду



### Аукцион серверов

Цены падают каждую секунду

## Почему open source — это важно?

Открытый код даёт возможность сообществу вносить улучшения, исправлять ошибки и расширять функциональность. Гибкость библиотеки позволяет каждому настроить поиск облигаций под собственные нужды, создавая индивидуальные стратегии отбора.

## Как практически использовать эту библиотеку

Допустим, у нас есть 300 000 рублей, которые мы хотим вложить в облигации. Чтобы минимизировать риски, разделим сумму на 10 разных облигаций.



### Аукцион серверов

Цены падают каждую секунду

Первая часть скрипта анализирует рынок и отбирает бумаги с хорошей ликвидностью,

подходящие под заданные критерии (доходность, дюрация, цена и т. д.).

## 2. Проверка эмитента

Запускаем вторую часть скрипта — он собирает последние новости по эмитентам.

Если обнаружены негативные публикации (например, судебные иски или финансовые проблемы компании), такие облигации исключаем из списка.

## 3. Расчёт денежных потоков

Далее, используя третью часть скрипта, можно заранее рассчитать будущие выплаты по купонам и спрогнозировать точную доходность портфеля.

## 4. Расчёт объёма покупки

Последний скрипт поможет рассчитать, сколько именно облигаций можно приобрести с учётом доступного капитала и НКД. Это позволяет эффективно распределить средства и избежать недостатка ликвидности.

Раз в месяц достаточно просматривать портфель, анализировать новые облигации через первый скрипт и при необходимости докупать бумаги. Такой алгоритм можно повторять бесконечно, постепенно увеличивая капитал.

## Заключение

Использование этого скрипта позволяет частному инвестору систематизировать процесс подбора облигаций, минимизировать риски и упростить управление портфелем.



**Аукцион серверов**

Цены падают каждую секунду

Этот проект развивается благодаря усилиям энтузиастов и разработчиков, заинтересованных в автоматизации инвестирования. Если у вас есть идеи по улучшению функциональности или вы хотите протестировать новые возможности, присоединяйтесь к обсуждению на [GitHub](#)!

Любые предложения, правки и новые модули помогут сделать библиотеку ещё более мощным инструментом для инвесторов.

**Автор:** Михаил Шардин

 [Моя онлайн-визитка](#)

 [Telegram «Умный Дом Инвестора»](#)

25 марта 2025 г.

**Теги:** [облигации](#), [московская биржа](#), [moex](#), [Сезон Open source](#), [алгоритмическая торговля](#)

**Хабы:** [Python](#), [Проектирование API](#), [Open source](#), [Финансы в IT](#)

## Редакторский дайджест



Присылаем лучшие статьи раз в месяц



Оставляя свою почту, я принимаю [Политику конфиденциальности](#) и даю согласие на получение рассылок

**208**

Карма

**167.2**

Рейтинг

**Михаил Шардин** [@empenoso](#)

[Автоматизация](#) / [Данные](#) / [Финансы](#) / [Умные дома](#)

[Подписаться](#)

[Сайт](#) [Сайт](#) [Github](#)

**Аукцион серверов**

Цены падают каждую секунду

Комментарии 43

## Публикации

ЛУЧШИЕ ЗА СУТКИ

ПОХОЖИЕ



km1337

17 часов назад

### Чисто русский переезд в другой дата-центр

7 мин

5.6K

+74

22

6



### Аукцион серверов

Цены падают каждую секунду

замедлился

 6 мин  12K +38 15 44**Dmitry\_zm**

19 часов назад

## Koda: AI-помощник разработчика – бесплатно, без VPN, с поддержкой русского языка



Простой



3 мин



4K

 Обзор +35 67 37**M\_Y\_L\_T**

19 часов назад

## Чтобы построить корабль из дерева, мне нужно 350 часов и 50 тысяч рублей. Но готовый я не продам: для меня он бесценный



Простой



3 мин



7.8K

 +33 17 52**DAN\_SEA**

15 часов назад

## Про гидроабразивную резку



Средний



12 мин



1.5K

 Обзор +24 12 1**vazhendima**

23 часа назад

## Что такое FinOps, зачем он нужен в 2025 году и с чем его едят



Простой



8 мин



1.7K

**Аукцион серверов**

Цены падают каждую секунду

 +24 17 32**Lunathecat**

19 часов назад

## Компактный девятивольтовый повербанк для гитарного педалборда (или роутера)



Простой



8 мин



1.2K

Тutorial

 +21 7 4**AndreyZolotykh**

17 часов назад

## Жизненный цикл фичи в коммутаторе: от идеи через QA до прода



Простой



11 мин



811

Обзор

 +18 12 0**GrishinAlex**

20 часов назад

## Как настроить Kafka в DBaaS от Selectel: подробный разбор параметров конфигурации



Простой



7 мин



685

Обзор

 +18 5 0**AndreyButov**

6 часов назад

## Про IT в 2025 году



Средний



13 мин



2.7K

**Аукцион серверов**

Цены падают каждую секунду



## Рационализируйте это: тест на то, какой вы в отпуске

Турбо

Показать еще

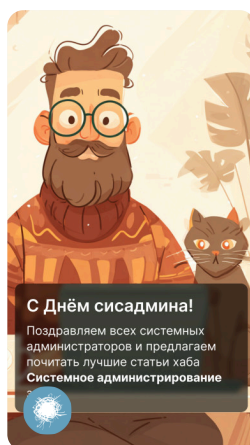
### ИСТОРИИ



**Как изменить реальность за окном за секунду?**



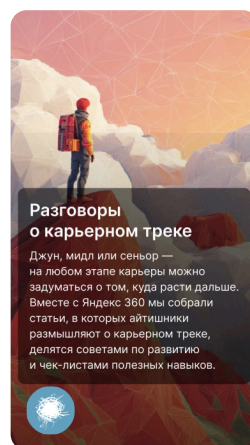
**Топ-7 годных статей из блогов компаний**



**С Днём сисадмина!**



**История Яндекс Почты**



**Как расти в ИТ: советы, гайды и опыт сеньоров**



**Кур... до 6**

### КУРСЫ

**SE** Проектирование и реализация очередей в брокерах RabbitMQ и Apache Kafka

24 августа 2025 · Systems Education

**SE** Проектирование интеграции с REST API

31 августа 2025 · Systems Education

**SE** Systems Analyst Bootcamp: Проектировщик корпоративных информационных систем

4 сентября 2025 · Systems Education

**rsu.ru** Data science на языке Python

9 сентября 2025 · РЭУ им. Г. В. Плеханова

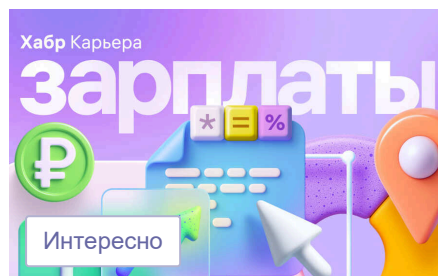


**Аукцион серверов**

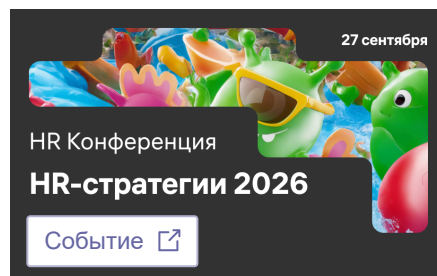
Цены падают каждую секунду

Больше курсов на Хабр Карьере

## МИНУТОЧКУ ВНИМАНИЯ



Зарплаты IT-специалистов в первой половине 2025



Го с нами на онлайн-конфу для HR в сентябре?



Сколько хостов выдерживает один админ? Вот и узнаем

## БЛИЖАЙШИЕ СОБЫТИЯ



15 августа

BigData MeetUp от 43Tech



Аукцион серверов

Цены падают каждую секунду

[Больше событий в календаре](#)

Хабр



 [Настройка языка](#)

[Техническая поддержка](#)

© 2006–2025, Habr



**Аукцион серверов**

Цены падают каждую секунду