



Все финансовые рынки мира в API Яху Финанс

Python, Алгоритмы, Node.JS, API, Финансы в IT

В этой статье я расскажу об API агрегатора финансовых данных [Yahoo! Finance](#). В рассказе есть один нюанс — официальное API Яху Финанс было [закрыто три года назад](#), однако практически сразу же появилась его недокументированная работоспособная версия, которая жива до сих пор. Хочу в исследовательских целях рассказать об использовании этой работоспособной версии подробнее.

Тем более, что список рынков, данные с которых можно получать через Яху Финанс [огромен](#). На текущий момент в нем 79 стран, включая и Россию.

The screenshot shows the Yahoo Finance homepage. At the top, there's a navigation bar with links for Home, Mail, News, Finance, Sports, Entertainment, Search, Mobile, and More... Below the navigation is the Yahoo Finance logo and a search bar. The main content area features a large banner with the headline "Leadership Development". Below the banner, there are several data boxes: "COVID-19 Confirmed Cases" (1,841,691 +16,520 US | 6,396,683 Global), "Stock futures tick down, pausing after rally" (with a small image of a person holding a protest sign that says "FLATTEN THE CURVE OF UNEMPLOYMENT"), and a "Quote Lookup" section. To the right, there are boxes for "U.S. markets open in 8 hours 11 minutes", "S&P Futures" (3,113.59 -4.25 (-0.14%)), "Dow Futures" (26,230.00 +2.00 (+0.01%)), "Nasdaq Futures" (9,674.50 -10.25 (-0.11%)), "Russell 2000 Futures" (1,447.90 -6.50 (-0.46%)), "Crude Oil" (36.60 -0.69 (-1.85%)), "Gold" (1,704.90 +0.10 (+0.01%)), and a "My Portfolio & Markets" section with recent activity for VTI, CSCO, and SPY.

Apple Inc. (AAPL) на сайте и в API Яху Финанс

Взгляд на данные с позиции долгосрочного частного инвестора

Ниже рассмотрю получение только тех параметров, которые, на мой взгляд, могут быть интересны долгосрочному частному инвестору:

1. Название бумаги
2. Цена бумаги
3. Доход с начала года
4. Дивидендная доходность
5. Дата предыдущего дивиденда
6. Значение предыдущего дивиденда
7. Годовая плата для фондов

ЧИТАЮТ СЕЙЧАС

[Как увидеть ??? Нужно швырнуть ?? в стену](#)

3,3k 19

[9 тяжелых уроков, которые я усвоил за 18 лет разработки](#)

21,3k 43

[Обзорщик обвинил CD Projekt Red в скрытии информации о реальной производительности Cyberpunk 2077 на Xbox One и PS4](#)

2,5k 20

[Что в стоматологии поменялось за 30 лет, и почему заниматься наукой в России так больно](#)

3,3k 17

[Косплей и как это работает](#)

2,6k 4

8. Категория бумаг

API Яху Финанс предоставляет ответы в формате JSON. На скриншотах с использованием API я использую расширение для браузера Google Chrome: [JSON-handle](#).

Название бумаги / Name

Имея только тикер всегда можно получить огромное число параметров, первым в списке, на мой взгляд стоит наименование инструмента. Сначала [найдем его на сайте](#) на примере SPDR S&P 500 ETF Trust (SPY):

SPDR S&P 500 ETF Trust (SPY)
NYSEArca - Nasdaq Real Time Price in USD

305.45 **+1.13 (+0.37%)**

At close: 4:00PM EDT

	Previous Close	Open	Bid	Ask	Day's Range	52 Week Range	Volume	Avg. Volume
	304.32	303.62	0.00 x 1100	0.00 x 1400	303.06 - 306.20	218.26 - 339.08	51,765,787	169,145,495
	Net Assets	NAV	PE Ratio (TTM)	Yield	YTD Daily Total Return	Beta (5Y Monthly)	Expense Ratio (net)	Inception Date
	261.75B	304.41	N/A	1.99%	-4.90%	1.00	0.09%	1993-01-22

Summary Chart Conversations Historical Data Profile Options Holdings Performance Risk

Full screen

Advertisement: The Latest Sedan Models

People Also Watch:

- DIA 255.14 +0.85 +0.33%
- SPDR Dow Jones Industrial Average 234.06 +0.70 +0.30%
- QQQ 234.06 +0.70 +0.30%
- IWM 140.14 +1.24 +0.89%

Имя ETF на сайте Яху Финанс

После этого найдем имя уже через API — в формате JSON оно выводится в двух вариантах: `shortName` и `longName`. Ссылка на данные, включающая в себя имя выглядит следующим образом:

<https://query1.finance.yahoo.com/v10/finance/quoteSummary/SPY?modules=price>

Path: `JSON.quoteSummary.result[0].price.longName`

Key: `longName`

"SPDR S&P 500 ETF Trust"

Имя ETF через API Яху Финанс

Свойство с именем longName содержит строковое значение SPDR S&P 500 ETF Trust.

Путь для получения этого ключа: JSON.quoteSummary.result[0].price.longName

Цена бумаги / Price

Следующий важный параметр — конечно цена. Найдем её для акций [Berkshire Hathaway Inc. \(BRKA\)](#):

The screenshot shows the Yahoo Finance interface for the stock BRKA. The main headline displays the price as **275,600.00** with a change of **-3,040.00 (-1.09%)**. A red circle highlights the current price value. Below the headline, there's a summary table with various financial metrics like Previous Close, Open, Bid, Ask, Day's Range, 52 Week Range, Volume, Avg. Volume, Fair Value, and Related Research. To the right of the summary is a candlestick chart showing price movement throughout the day. Further down, there's a section for 'Chart Events' with a note about a 'Neutral pattern detected'. On the far right, there's a sidebar for 'People Also Watch' with a small chart and some other stock information.

Цена акций на сайте Яху Финанс

Дальше получим цену через API. Раздел где содержится цена, аналогичен получению имени:

<https://query1.finance.yahoo.com/v10/finance/quoteSummary/BRKA?modules=price>

The screenshot shows the browser developer tools (F12) with the 'Elements' tab open. It displays a hierarchical tree of JSON objects. The path to the price is highlighted with a red circle, showing the key `regularMarketPrice` with its raw value `275600` and formatted value `275,600.00`. To the right of the tree, there's a detailed view of the selected object, showing the raw value `275600`, the key `raw`, and several buttons for copying or modifying the value. At the bottom, there are checkboxes for 'Show value', 'Show img', 'Show array leng', and 'Show ico'.

Цена акций через API Яху Финанс

Ключ с именем raw для regularMarketPrice можно получить как с форматированием,

так и без него. Путь для получения этого свойства:

```
JSON.quoteSummary.result[0].price.regularMarketPrice.raw
```

Доход с начала года / YTD Daily Total Return

Этот параметр можно посмотреть только для фондов. На примере [Vanguard Total Stock Market Index Fund ETF Shares \(VTI\)](#):

Vanguard Total Stock Market Index Fund ETF Shares (VTI)

155.52 +0.77 (+0.50%)

As of 10:38AM EDT, Market open.

Fund Overview

Category	Large Blend
Fund Family	Vanguard
Net Assets	\$ 2,78
YTD Daily Total Return	-5.58%
Yield	1.88%
Legal Type	Exchange Traded Fund

Fund Operations

Attributes	VTI	Category Average
Annual Report Expense Ratio (net)	0.03%	0.36%
Holdings Turnover	4.00%	5,076.00%
Total Net Assets	76,435.80	76,435.80

Fund Summary

The investment seeks to track the performance of a benchmark index that measures the investment return of the overall stock market. The fund employs an indexing investment approach designed to track the performance of the CRSP US Total Market Index, which represents

Доход с начала года на сайте Яху Финанс

Посмотрим этот параметр через API. Раздел где содержится доход с начала года:

```
https://query1.finance.yahoo.com/v10/finance/quoteSummary/VTI?
modules=defaultKeyStatistics
```

```
Path: JSON.quoteSummary.result[0].defaultKeyStatistics.ytdReturn fmt
Key: fmt : Copy deURI deBase64 aLine
"-5.58%"
```

Доход с начала года через API Яху Финанс

Свойство с именем `fmt` для `ytdReturn` можно получить как с форматированием, так и без. Путь для получения этого свойства:

```
JSON.quoteSummary.result[0].defaultKeyStatistics.ytdReturn(fmt
```

Дивидендная доходность / Dividend Yield

Важный параметр, финансовые консультанты даже складывают доходность бумаги с дивидендной доходностью и показывают получившуюся цифру как потенциал роста бумаги. Найдем её для VanEck Vectors Russia ETF (RSX):

VanEck Vectors Russia ETF (RSX)

21.23 +0.61 (+2.96%)

At close: 4:00PM EDT

Summary **Chart** **Conversations** **Historical Data** **Profile** **Options** **Holdings** **Performance** **Risk**

Previous Close	20.62	Net Assets	871.66M	1D	5D	1M	6M	YTD	1Y	5Y	Max	Full screen	
Open	20.75	NAV	20.58										
Bid	0.00 x 2200	PE Ratio (TTM)	N/A										
Ask	0.00 x 1100	Yield	7.42%										
Day's Range	20.75 - 21.23	YTD Daily Total Return	-17.33%										
52 Week Range	13.14 - 26.57	Beta (SY Monthly)	1.25										
Volume	3,807,464	Expense Ratio (net)	0.67%										
Avg. Volume	9,026,806	Inception Date	2007-04-24										

Barons.com • 2 months ago
Russia's Stocks Are a Buy Only for Very Patient Investors

The Latest Sedan Models

People Also Watch

Symbol	Last Price	Change	% Change
EWZ	27.51	+0.32	+1.18%

Дивидендная доходность ETF на сайте Яху Финанс

Получим этот параметр через API. Раздел где содержится дивидендная доходность:

<https://query1.finance.yahoo.com/v10/finance/quoteSummary/RSX?modules=defaultKeyStatistics>

```

{
  "quoteSummary": {
    "result": [
      {
        "defaultKeyStatistics": {
          "yield": {
            "raw": 0.074200004,
            "fmt": "7.42%"
          }
        }
      }
    ]
  }
}
  
```

Дивидендная доходность ETF через API Яху Финанс

Свойство с именем fmt для yield можно получить как с форматированием, так и без. Путь для получения этого свойства:

`JSON.quoteSummary.result[0].defaultKeyStatistics.yield(fmt)`

Дата предыдущего дивиденда / Dividend Date

Найдем дату последней выплаты дивиденда. Для этого придется обращаться к истории и можно будет получить все дивидендные выплаты. На сайте эти данные в разделе [Historical Data](#), возьмем например акции Microsoft Corporation (MSFT):

Date	Dividend
May 20, 2020	0.51 Dividend
Feb 19, 2020	0.51 Dividend
Nov 20, 2019	0.51 Dividend
Aug 14, 2019	0.46 Dividend

Дата предыдущего дивиденда акции на сайте Яху Финанс

Получить их через API задача уже немного сложнее, потому что ссылка будет иметь вид:

<https://query1.finance.yahoo.com/v8/finance/chart/MSFT?symbol=MSFT&period1=1559457037&period2=1591079437&interval=1mo&includePrePost=true&events=div%7Csplit>

Где:

- period1 — начальная дата в виде [Unix Timestamp](#).
- period2 — конечная дата в виде [Unix Timestamp](#).
- interval=1mo — укрупненные свечи, меня интересуют только дивиденды.
- events=div%7Csplit — добавляет информацию о дивидендах и сплитах в вывод.

The screenshot shows a JSON viewer interface with the following structure:

```

{
  "events": {
    "dividends": [
      {
        "amount": 0.46,
        "date": 1564632000
      },
      {
        "amount": 0.51,
        "date": 1572580800
      },
      {
        "amount": 0.51,
        "date": 1574260200
      },
      {
        "amount": 0.51,
        "date": 1582122600
      },
      {
        "amount": 0.51,
        "date": 1588305600
      }
    ]
  },
  "indicators": {
    "quote": {
      "0": {
        "low": [
          119.01000213623047,
          134.6699981689453,
          130.7799877929688,
          134.50999450683594,
          133.22000122070312
        ]
      }
    }
  }
}

```

Дата предыдущего дивиденда акции через API Яху Финанс

Для получения даты возможны два варианта:

1. Считывать ключи `JSON.chart.result[0].timestamp` и перебирать по этим ключам даты дивидендов.
2. Более предпочтительный — получать массив значений перечисляемых свойств объекта `JSON.chart.result[0].events.dividends`.

Значение предыдущего дивиденда / Next Dividend

Полностью аналогично предыдущему разделу. Только ищем не дату, а значение.

Найдем значение прошлого дивиденда [для ETF iShares MSCI Mexico Capped ETF \(EWW\)](#):

The screenshot shows the Yahoo Finance page for EWW. Key information includes:

- Current Price:** 32.82 +0.93 (+2.92%)
- Historical Data:** Shows a table with the following data:

Date	Dividends
Dec 16, 2019	0.434 Dividend
Jun 17, 2019	0.882 Dividend
- Advertisement:** Leadership Development
- Advertisement:** The Latest Sedan Models

Значение предыдущего дивиденда ETF на сайте Яху Финанс

В API ссылка будет выглядеть:

<https://query1.finance.yahoo.com/v8/finance/chart/EWW?symbol=MSFT&period1=1559457037&period2=1591079437&interval=1mo&includePrePost=true&events=div%7Csplit>

Расшифровка запроса аналогична получению даты выше.

The screenshot shows a browser window with the URL <https://query1.finance.yahoo.com/v8/finance/chart/EWW?symbol=MSFT&period1=1559457037&period2=1591079437&interval=1mo&includePrePost=true&events=div%7Csplit>. A JSON viewer extension is open, displaying a tree structure of the API response. A specific node under the 'events/dividends' section is highlighted with a red circle, showing the value 'amount: 0.434'. The right panel of the JSON viewer shows the path 'JSON.chart.result[0].events.dividends[1575176400].amount' and the value '0.434'.

Значение предыдущего дивиденда ETF через API Яху Финанс

Годовая плата / Expense Ratio

Годовая плата, которую все фонды или ETF взимают со своих акционеров. На сайте можно посмотреть в разделе Summary:

The screenshot shows the Yahoo Finance website at <https://finance.yahoo.com/quote/HYD>. The 'Summary' tab is selected. Key information displayed includes the current price of \$57.69, a change of +0.13 (+0.22%), and the expense ratio of 0.35%. The expense ratio is circled in red.

Годовая плата, которую все фонды или ETF взимают со своих акционеров на сайте Яху Финанс

В API ссылка будет выглядеть:

<https://query1.finance.yahoo.com/v10/finance/quoteSummary/HYD?modules=fundProfile>

The screenshot shows a JSON viewer interface with a tree view on the left and a details panel on the right. The tree view shows a nested structure of fund profile fields, including 'maxAge', 'styleBoxUrl', 'family', 'categoryName', 'legalType', 'managementInfo', and 'feesExpensesInvestment'. The 'feesExpensesInvestment' node has a child 'annualReportExpenseRatio' which is circled in red. This node contains a 'raw' key with the value '0.0034999999' and a 'fmt' key with the value '0.35%'. The details panel on the right shows the path 'JSON.quoteSummary.result[0].fundProfile.feesExpensesInvestment.an' and the value '0.35%'.

Годовая плата, которую все фонды или ETF взимают со своих акционеров через API Яху Финанс

Свойство с именем fmt для annualReportExpenseRatio можно получить как с форматированием, так и без. Путь для получения этого свойства:

`JSON.quoteSummary.result[0].fundProfile.feesExpensesInvestment.annualReportExpenseRatio(fmt)`

Категория акций / Sector и Industry

На сайте можно посмотреть в разделе [Profile](#). Для примера возьмем [Cisco Systems, Inc. \(CSCO\)](#):

The screenshot shows the Yahoo Finance profile page for Cisco Systems, Inc. (CSCO). The top navigation bar includes links for Finance Home, Coronavirus, Watchlists, My Portfolio, Screeners, Markets, News, Personal Finance, Videos, Industries, and Premium. The main content area shows the company's name, address, phone number, and website. Below this is a table of key executives. To the right, there is a sidebar with advertisements for 'The Latest Sedan Models' and sections for Upcoming Events (Cisco Systems, Inc. Earnings Call on Aug 12, 2020) and Recent Events (10-Q: Quarterly Report on May 18, 2020, and 8-K: Results of Operations and Financial Condition, Financial Statements and Exhibits on May 13, 2020). A map of San Jose, CA is also present.

Категория акции на сайте Яху Финанс

Эти данные можно посмотреть только для акций и возможных вариантов секторов не так уж и много:

1. Basic Materials

2. Consumer Cyclical

3. Financial Services

4. Real Estate

5. Consumer Defensive

6. Healthcare

7. Utilities

8. Communication Services

9. Energy

10. Industrials

11. Technology

Подкатегории для секторов — Industry.

В API ссылка будет выглядеть:

<https://query1.finance.yahoo.com/v10/finance/quoteSummary/CSCO?modules=assetProfile>

Категория акции через API Яху Финанс

Свойство с ключом sector можно получить по следующему пути:

JSON.quoteSummary.result[0].assetProfile.sector

Общие правила работы с API Яху Финанс

Хост

query1.finance.yahoo.com для HTTP / 1.0

query2.finance.yahoo.com для HTTP / 1.1

Основные данные

/v10/finance/quoteSummary/GOOGL?modules= (Полный список модулей ниже)

(замените GOOGL на любой символ)

Входные данные для ?modules= запроса:

1. assetProfile
2. incomeStatementHistory
3. incomeStatementHistoryQuarterly
4. balanceSheetHistory
5. balanceSheetHistoryQuarterly
6. cashflowStatementHistory
7. cashflowStatementHistoryQuarterly
8. defaultKeyStatistics
9. financialData
10. calendarEvents
11. secFilings
12. recommendationTrend
13. upgradeDowngradeHistory
14. institutionOwnership
15. fundOwnership
16. majorDirectHolders
17. majorHoldersBreakdown
18. insiderTransactions
19. insiderHolders
20. netSharePurchaseActivity
21. earnings
22. earningsHistory
23. earningsTrend
24. industryTrend
25. indexTrend
26. sectorTrend

и возможно что-то ещё...

Пример URL:

```
https://query1.finance.yahoo.com/v10/finance/quoteSummary/GOOGL?modules=assetProfile%2CearningsHistory
```

Запрос для: assetProfile и earningsHistory.

История цен, сплитов и дивидендов

```
/v8/finance/chart/GOOGL?symbol=GOOGL&period1=0&period2=9999999999&interval=3mo
```

Интервалы:

&interval=3mo это 3 месяца.

&interval=1d это 1 день.

&interval=5m это 5 минут, возвращает 80 дней.

&interval=1m это 1 минута, возвращает 4-5 дней.

period1= unix timestamp представление даты, с которой вы хотите начать. Значения ниже начальной торговой даты будут округлены до начальной торговой даты.

period2= unix timestamp представление даты, на которой вы хотите закончить.

Значения, превышающие последнюю торговую дату, будут округлены до последней доступной отметки времени.

Добавить данные до и после рынка: &includePrePost=true

Добавить дивиденды и сплиты: &events=div%2Csplit

Пример полного запроса:

```
https://query1.finance.yahoo.com/v8/finance/chart/AAPL?symbol=AAPL&period1=0&period2=9999999999&interval=1d&includePrePost=true&events=div%2Csplit
```

Приведенный выше запрос вернет все данные о цене тикера GOOGL с интервалом в 1 день, включая данные до и после рынка, а также дивиденды и сплиты.

Всю информацию об API можно получить изучая код страницы Яху Финанс, но первоначально информация была взята с [Stack Overflow](#) и [GitHub](#).

Пример кода на Node.js

```

const fetch = require('node-fetch');
async function USAStockGetName(ID) { //получаем имя бумаги
    const url = `https://query1.finance.yahoo.com/v10/finance/quoteSummary/${ID}?modules=price`;
    // console.log("USAStockGetName. url для %s: %s", ID, url);
    try {
        const response = await fetch(url)
        const json = await response.json()
        const value = json.quoteSummary.result[0].price.longName
        console.log("USAStockGetName. Название для %s: %s", ID, value)
        if (value == 0) return 'нет'
        return value
    } catch (e) {
        console.log('Ошибка в USAStockGetName')
    }
}
module.exports.USASockGetName = USAStockGetName;

```

```

const fetch = require('node-fetch');
async function USAStockGetName(ID) { //получаем имя бумаги
    const url = `https://query1.finance.yahoo.com/v10/finance/quoteSummary/${ID}?modules=price`;
    // console.log("USAStockGetName. url для %s: %s", ID, url);
    try {
        const response = await fetch(url)
        const json = await response.json()
        const value = json.quoteSummary.result[0].price.longName
        console.log("USAStockGetName. Название для %s: %s", ID, value)
        if (value == 0) return 'нет'
        return value
    } catch (e) {
        console.log('Ошибка в USAStockGetName')
    }
}
module.exports.USASockGetName = USAStockGetName

```

Пример кода на Python

→ Можно посмотреть на [GitHub](#)

Итог

Я написал эту статью, желая в исследовательских целях разобраться в работающем API Яху Финанс, содержащем подробные данные о десятках тысяч ценных бумаг по всему миру, включая Россию.

Автор: [Михаил Шардин](#),

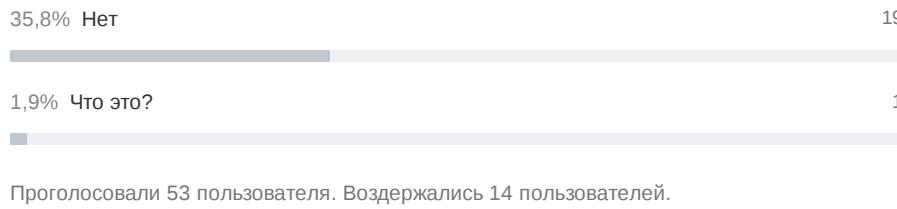
8 июня 2020 г.

Только зарегистрированные пользователи могут участвовать в опросе. [Войдите](#), пожалуйста.

Используете финансовые API?

62,3% Да

33



Теги: парсинг, котировка, биржа, инвестиции, статистика, ценные бумаги, облигации, биржевая торговля, JavaScript, Node.JS, Python

Хабы: Python, Алгоритмы, Node.JS, API, Финансы в IT

↑ +15 ↓ 111 📒 10,9k 💬 36 ⚡ Поделиться



Михаил Шардин @empenoso

Разработчик

Github

ПОХОЖИЕ ПУБЛИКАЦИИ

27 ноября 2016 в 19:08

[Настольный пульт управления на JavaScript/Node.js для робота на Ардуине](#)

↑ +14 ↓ 13k 📒 94 💬 14

24 февраля 2016 в 13:46

[Тонкости Javascript/Node.js. Увеличиваем производительность в десятки раз](#)

↑ +23 ↓ 41,2k 📒 310 💬 38

14 декабря 2014 в 01:27

[Выразительный JavaScript: Node.js](#)

↑ +39 ↓ 137k 📒 499 💬 8

ВОПРОСЫ И ОТВЕТЫ

Как устранить парсинг одной и той же картинки?

Python • Простой • 1 ответ

Почему страница не скачивается на сервере, а с локального скачивается с одним и тем же прокси?

Парсинг • Простой • 0 ответов

Как можно осуществить парсинг всех ссылок сайта?

Парсинг • Простой • 1 ответ

Как steam может идентифицировать парсер?

Python • Средний • 1 ответ

МИНУТОЧКУ ВНИМАНИЯ



Новогодние скидки на мерч Хабра



Листья опали, а промокод активен до Нового года

Кто знает скрипт который бы с дзена (парсер) брал все статьи автора?
Парсинг • Средний • 1 ответ

[Больше вопросов на Хабр Q&A](#)

Реклама



Хочешь Иметь Содержанку?

Место Встречи Успешных Мужчин и Красивых Девушек. Выбирай Отношения на Твоих Условиях!

emilydates.com



[Открыть >](#)

Комментарии 36

ramef 8 июня 2020 в 16:47 <#> [Bookmark](#)

[↑](#) +1 [↓](#)

Спасибо за статью, но в самом начале сформулирован вопрос, на который нет ответа в целой большой и полезной статье.

официальное API Яху Финанс было закрыто три года назад, однако практически сразу же появилась его недокументированная работоспособная версия, которая жива до сих пор.

Вопрос: что конспирологи и другие исследователи этого мира думают по поводу такого артефакта? Кто платит за этот банкет со стороны Яху и зачем они это делают? И т.п.

emprenoso 8 июня 2020 в 16:51 <#> [Bookmark](#) [h](#) [o](#)

[↑](#) 0 [↓](#)

Скорее всего это баг — то что им можно пользоваться.

ramef 8 июня 2020 в 18:27 <#> [Bookmark](#) [h](#) [o](#)

[↑](#) +1 [↓](#)

Баг — это дыра, а вот платить 3+ года за поддержку офиц. закрытого — это не баг, а фича. Но за ради чего или кого, Холмс?

OnvogSGN 10 июня 2020 в 11:03 <#> [Bookmark](#) [h](#) [o](#)

[↑](#) +1 [↓](#)

Вряд ли там какая-то поддержка в полном смысле слова. Скорее, оставили и больше не трогали. Это ж Яху, оно всегда так работало :) А может, оставили ради немногочисленных действующих клиентов, чтобы не ломать им процессы.

n1k_ivanov 8 июня 2020 в 16:50 <#> [Bookmark](#)

[↑](#) +1 [↓](#)

Так же советую посмотреть тиньков инвестиции API. Тоже интересно, можно делать даже торговые боты с управление в ТГ (как пример).

emprenoso 8 июня 2020 в 16:53 <#> [Bookmark](#) [h](#) [o](#)

[↑](#) 0 [↓](#)

Это другая тема конечно. Среди российских брокеров наверное по пальцам одной руки можно пересчитать кто имеет своё API. И Тинькофф банк один из них.

emprenoso 8 июня 2020 в 17:38 <#> [Bookmark](#) [h](#) [o](#)

[↑](#) 0 [↓](#)

Вот обсуждение было: habr.com/ru/post/496722/#comment_21491254

ЧТО ОБСУЖДАЮТ

[Сейчас](#) [Вчера](#) [Неделя](#)

Абиогенез и алгоритмы естественного отбора

[674](#) [9](#)

Обзорщик обвинил CD Projekt Red в скрытии информации о реальной производительности Cyberpunk 2077 на Xbox One и PS4

[2,5k](#) [20](#)

Мифический человеко-месяц 45 лет спустя

[6,2k](#) [66](#)

PandaDoc просит освободить Виктора Кувшинова

[11,2k](#) [187](#)

Мониторим оборудование в реальном времени. Причем тут ИИ?

[Интересно](#)



OnvogSGN 10 июня 2020 в 01:21

📒 ⌚ 🔍

↑ +1 ↓

Не могу ответить там. Альфа-директ имеет своё API задолго до появления Тинькова



empenoso 10 июня 2020 в 05:07

📒 ⌚ 🔍

↑ 0 ↓

Вы имеете в виду для отчетов или что-то другое?



OnvogSGN 10 июня 2020 в 10:54

📒 ⌚ 🔍

↑ +1 ↓

Для всех операций, которые можно делать через терминал. В том числе, для подачи заявок.



empenoso 10 июня 2020 в 11:05

📒 ⌚ 🔍

↑ 0 ↓

Подскажите, а где подробнее можно про это почитать?



OnvogSGN 10 июня 2020 в 11:33

📒 ⌚ 🔍

↑ +1 ↓

А вот это сложный вопрос :))) Вроде, оно закопано где-то в глубинах их сайта. Это частый вопрос новичков в чате АД. Но текущей 4-й версией API я уже не интересовался и документацию не видел. Ну и в текущей версии они делают больше упор на программирование роботов и индикаторов внутри самого терминала, на каком-то C#-подобном языке.



empenoso 10 июня 2020 в 13:39

📒 ⌚ 🔍

↑ 0 ↓

А собственные позиции только в режиме чтения можно через апи в Альфа Директе получать?



OnvogSGN 10 июня 2020 в 14:05

📒 ⌚ 🔍

↑ +1 ↓

Опять же — документацию к текущей версии не видел, но уверен, что да. Это же элементарная операция и в то же время, одна из важнейших.
Точно знаю, что у StockSharp есть коннектор через это API. А уж это полноценная торговая система, соответственно, все эти возможности в API АД есть.



vrangel 8 июня 2020 в 22:10

📒

↑ +1 ↓

Спасибо за статью.

Вопрос: Есть ли какие-нибудь ограничения на количество запросов? Если я, скажем, захочу сделать анализ всех акций сектора и запрошу параллельно несколько тысяч параметров?

Я где-то встречал информацию, что этот API выдает агрегированную информацию за 15 (???) минут.



empenoso 9 июня 2020 в 05:11

📒 ⌚ 🔍

↑ 0 ↓

Никаких специальных тестов я не проводил. Но на глаз несколько десятков запросов за минуту вполне работает.



stepmex 9 июня 2020 в 12:08

📒

↑ +1 ↓

А запрос на получение списка всех ценных бумаг (symbols) есть?

 **empenoso** 9 июня 2020 в 14:07 <#> [Bookmark](#) [h](#) [o](#) ↑ 0 ↓

Мне о таком неизвестно.

 **postgres** 24 июня 2020 в 01:51 <#> [Bookmark](#) [h](#) [o](#) ↑ +1 ↓

я находил такой инструмент, который перебором бешеное количество выкачивает
pypi.org/project/Yahoo-ticker-downloader

Понял, что проще брать то, что нужно (торгуется у брокера)

 **empenoso** 24 июня 2020 в 05:06 <#> [Bookmark](#) [h](#) [o](#) ↑ 0 ↓

Спасибо!

 **empenoso** 24 июня 2020 в 06:17 <#> [Bookmark](#) [h](#) [o](#) ↑ 0 ↓

Похоже в настоящий момент он устарел, потому что в коде используется
отключченное АПИ.

 **postgres** 25 июня 2020 в 09:55 <#> [Bookmark](#) [h](#) [o](#) ↑ +1 ↓

Но при этом работает

▶ [start](#)

 **empenoso** 25 июня 2020 в 10:25 <#> [Bookmark](#) [h](#) [o](#) ↑ 0 ↓

Спасибо, очень полезно. С python не особо знаком, на node.js переделаю.

Скрипт получается перебирает просто буквы алфавита? Но при этом в
выборку попадает только американский рынок?

Например просто [буква g](#) — но в результатах нет gazp.me например, однако
если запрос [gazp](#), то эта российская бумага уже в выборке.

 **postgres** 25 июня 2020 в 10:44 <#> [Bookmark](#) [h](#) [o](#) ↑ +1 ↓

Я особо не вникал, оставил на ночь, и он собрал мне
150к строк, и там нет газпрома)
видимо возьмет его лишь когда дойдет до [gazp.me](#)
Поэтому такой перебор мне не понравился) Может там есть какие-то
параметры... но я подумал, что иду не по тому пути)

finance.yahoo.com/_finance_doubledown/api/resource/searchassist;searchTerm=g&device=console&returnMeta=true

 **empenoso** 25 июня 2020 в 11:08 <#> [Bookmark](#) [h](#) [o](#) ↑ +1 ↓

Вот скрипт выборки всех бумаг с nasdaq — прямо с их ftp:

```
#!/bin/bash

echo -n '[' > stocks.json
curl -o nasdaq.txt ftp://ftp.nasdaqtrader.com/symboldirect
```

```
ory/nasdaqtraded.txt
cat nasdaq.txt | grep -Eo '^\\w\\|\\w*' | sed 's/^\\w//g' | sed 'H;1h;$!d;x;y/\n//' | sed 's/,/"/,\"/g' >> stocks.json
echo '"'"]'" >> stocks.json
sed -i ':a;N;$!ba;s/\n//' stocks.json
rm nasdaq.txt
```



empenoso 25 июня 2020 в 11:11



0

Скрипт на каждой итерации получается добавляет +1 букву?

буква

буква+буква

буква+буква+буква

...



postgres ⚡ 25 июня 2020 в 11:16



+1



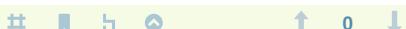
ага(

еще точку, цифры, %

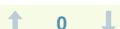
Возможно там чуть умнее перебор, и не возможные комбинации этого дела собираются, но все равно очень долго и много)



empenoso 29 июня 2020 в 14:35



0



Кстати, если вы ищите просто общий список акций с мосбиржи, то [вот он](#). Можно перебирать еще [группы](#).



empenoso 20 июля 2020 в 21:20



0



А можете последовательность как он перебирает символы сбросить?



mrShadow 12 июня 2020 в 14:22



+1



Занятно, что Yahoo.Finance выдаёт Sector и Industry для компаний, торгующихся только в России, тоже. Например:

query1.finance.yahoo.com/v10/finance/quoteSummary/OBUV.ME?modules=assetProfile

```
industry: "Footwear & Accessories",
sector: "Consumer Cyclical",
```



empenoso ⚡ 12 июня 2020 в 15:18



0



Да, но только для акций. Очень удобно кстати.



mrShadow 12 июня 2020 в 14:39



+1

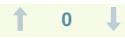


Вообще получение истории цен это супер возможность, спасибо! А то у меня функция GOOGLEFINANCE в Google Spreadsheets периодически глючит, либо начиная выдавать ошибку N/A, либо зависая в Loading... Может, перейду на использование скриптов (Google App Scripts), откуда буду дёргать Yahoo.Finance.

Жаль конечно, что информации о российских облигациях в Yahoo.Finance, видимо, нет, так же как в GOOGLEFINANCE. Остаёмся на API Московской биржи.



empenoso 12 июня 2020 в 15:21



Там нет и американских облигаций, и европейских. С получаем параметров облигаций вообще не очень радужная ситуация.



postgres 24 июня 2020 в 01:45



Где вы раньше были)
не так давно требовалось скачать историю акций которые есть в Тинькофф, случайно наткнулся на Yahoo.
Заметил, что по российским акциям можно лишь к тикеру .ME прилепить

сгенерил для bash такой скрипт скачивания csv (да, можно было бы и цикл на тикер для красоты)

► [only_rus.sh](#)

Заметил, что:

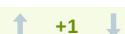
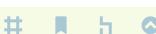
— все же нужна задержка, т.к. запросы повисали наглухо, и как будто банился їр на время
— иногда странные данные, типа 0-е объемы, хотя бумага явно торговалась в этот день,
или цена в какие-то дни якобы падала раз в 10. Но по данным другим источников такого не было

прим.: посмотрите график за 5 лет finance.yahoo.com/quote/BANEP.ME

— не показывается данные текущего дня по «американским» акциям, до начала захода Америки, хотя мы уже вовсю торгуем ими. Или я не туда смотрю?



empenoso 24 июня 2020 в 05:26



Вот [ссылка с запросом](#) на Мосбиржу.



empenoso 24 июня 2020 в 05:08



Вообще, оптимальнее воспользоваться апи Московской биржи для этой задачи.

Только [полноправные пользователи](#) могут оставлять комментарии. [Войдите](#),
пожалуйста.

САМОЕ ЧИТАЕМОЕ[Сутки](#)[Неделя](#)[Месяц](#)**9 тяжелых уроков, которые я усвоил за 18 лет разработки** [+36](#) [21,3k](#) [134](#) [43](#)**Библиотека от AMD стала причиной плохой производительности процессоров AMD в Cyberpunk 2077** [+56](#) [59,4k](#) [28](#) [63](#)**До свидания, Google Fonts. Последний аргумент** [+67](#) [21,3k](#) [144](#) [47](#)**Темно-серая зона экосистемы Telegram** [+8](#) [12,5k](#) [23](#) [56](#)

Прокрастинатор поневоле. Как я написал (нет) крутую статью про видеокарты

[Мегапост](#)

Ваш аккаунт

[Войти](#)[Регистрация](#)

Разделы

[Публикации](#)[Новости](#)[Хабы](#)[Компании](#)[Пользователи](#)[Песочница](#)

Информация

[Устройство сайта](#)[Для авторов](#)[Для компаний](#)[Документы](#)[Соглашение](#)[Конфиденциальность](#)

Услуги

[Реклама](#)[Тарифы](#)[Контент](#)[Семинары](#)[Мегапроекты](#)[Мерч](#)

© 2006 – 2020 «Habr»

[Настройка языка](#)[О сайте](#)[Служба поддержки](#)[Мобильная версия](#)