

Getting Started with Market Data API

This is a quick guide on how to start consuming market data via APIs. Starting from beginning to end, this section outlines how to install Alpaca's software development kit (SDK), create a free alpaca account, locate your API keys, and how to request both historical and real-time data.

Installing Alpaca's Client SDK

In this guide, we'll be making use of the SDKs provided by Alpaca. Alpaca maintains SDKs in four languages: Python, JavaScript, C#, and Go. Follow the steps in the installation guide below to install the SDK of your choice before proceeding to the next section.

[Python](#) [Go](#) [JavaScript](#) [C#](#)

```
pip install alpaca-py
```

Generate API Keys

Go to the [Alpaca dashboard](#) and find the **API Keys** section on the right sidebar. Click on Generate New Keys and save the generated API credentials. If you have previously generated keys there and you lost the secret, you can also regenerate them here.

How to Request Market Data Through the SDK

With the SDK installed and our API keys ready, you can start requesting market data. Alpaca offers many options for both historical and real-time data, so to keep this guide succinct, these examples are on obtaining historical and real-time bar data. Information on what other data is available can be found in the Market Data API reference.

To start using the SDK for historical data, import the SDK and instantiate the crypto historical data client. It's not required for this client to pass in API keys or a paper URL.

[Python](#) [Go](#) [JavaScript](#)

```
from alpaca.data.historical import CryptoHistoricalDataClient

# No keys required for crypto data
client = CryptoHistoricalDataClient()
```

Next we'll define the parameters for our request. Import the request class for crypto bars, `CryptoBarsRequest` and `TimeFrame` class to access time frame units more easily. This example queries for historical daily bar data of Bitcoin in the first week of September 2022.

Python Go JavaScript

```
from alpaca.data.requests import CryptoBarsRequest
from alpaca.data.timeframe import TimeFrame

# Creating request object
request_params = CryptoBarsRequest(
    symbol_or_symbols=["BTC/USD"],
    timeframe=TimeFrame.Day,
    start=datetime(2022, 9, 1),
    end=datetime(2022, 9, 7)
)
```

Finally, send the request using the client's built-in method, `get_crypto_bars`. Additionally, we'll access the `.df` property which returns a pandas DataFrame of the response.

Python Go JavaScript

```
# Retrieve daily bars for Bitcoin in a DataFrame and printing it
btc_bars = client.get_crypto_bars(request_params)

# Convert to dataframe
btc_bars.df
```

Returns

Python Go JavaScript

			open	high	low	close	v
symbol	timestamp						
BTC/USD	2022-09-01 05:00:00+00:00		20055.79	20292.00	19564.86	20156.76	7141.9
	2022-09-02 05:00:00+00:00		20156.76	20444.00	19757.72	19919.47	7165.9
	2022-09-03 05:00:00+00:00		19924.83	19968.20	19658.04	19806.11	2677.6
	2022-09-04 05:00:00+00:00		19805.39	20058.00	19587.86	19888.67	4325.6

2022-09-05 05:00:00+00:00	19888.67	20180.50	19635.96	19760.56	6274.5
2022-09-06 05:00:00+00:00	19761.39	20026.91	18534.06	18724.59	11217.7

Request ID

All market data API endpoint provides a unique identifier of the API call in the response header with `X-Request-ID` key, the Request ID helps us to identify the call chain in our system.

Make sure you provide the Request ID in all support requests that you created, it could help us to solve the issue as soon as possible. Request ID can't be queried in other endpoints, that is why we suggest to persist the recent Request IDs.



Shell

```
$ curl -v https://data.alpaca.markets/v2/stocks/bars
...
> GET /v2/stocks/bars HTTP/1.1
> Host: data.alpaca.markets
> User-Agent: curl/7.88.1
> Accept: */*
>
< HTTP/1.1 403 Forbidden
< Date: Fri, 25 Aug 2023 09:37:03 GMT
< Content-Type: application/json
< Content-Length: 26
< Connection: keep-alive
< X-Request-ID: 0d29ba8d9a51ee0eb4e7bbaa9acff223
<
...
```

 Updated about 2 months ago

[← About Market Data API](#)

[Historical API →](#)

Did this page help you?  Yes  No

Developers

[Docs](#)[Slack](#)[Forum](#)[Support](#)[Learn](#)[SDKs & Libraries](#)[GitHub](#)[API Status](#)[Security](#)

About Us

[Story](#)[Blog](#)[Contact](#)[We're hiring](#)

Disclosures

[Disclosure Library](#)[Form CRS](#)[Form CRS Responses to Conversation Starters](#)[Privacy Policy](#)[Cookie Policy](#)[Terms & Conditions](#)[Submit Feedback](#)

© 2025 Alpaca Securities LLC All rights reserved.

© 2025 Alpaca Crypto LLC All rights reserved.

© 2025 AlpacaDB, Inc. All rights reserved.

Securities brokerage services are provided by Alpaca Securities LLC ("Alpaca Securities"), member [FINRA/SIPC](#), a wholly-owned subsidiary of AlpacaDB, Inc. Technology and services are offered by AlpacaDB, Inc.

Cryptocurrency services are made available by Alpaca Crypto LLC ("Alpaca Crypto"), a FinCEN registered money services business (NMLS # 2160858), and a wholly-owned subsidiary of AlpacaDB, Inc. Alpaca Crypto is not a member of SIPC or FINRA. Cryptocurrencies are not stocks and your cryptocurrency investments are not protected by either FDIC or SIPC. Please see the [Disclosure Library](#) for more information.

This is not an offer, solicitation of an offer, or advice to buy or sell securities or cryptocurrencies, or open a brokerage account or cryptocurrency account in any jurisdiction where Alpaca Securities or Alpaca Crypto respectively, are not registered or licensed, as applicable.

! [View Important Disclosures](#)