

Yahoo Fiance

```

File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[1] On
df = yf.download("AAPL",
                 start="2011-01-01",
                 end="2021-12-31",
                 progress=False)

[2] On
print(f"Downloaded {len(df)} rows of data.")
df

... Downloaded 2768 rows of data.
   Price    Close    High     Low   Open   Volume
   Ticker      AAPL      AAPL     AAPL     AAPL     AAPL
Date
2011-01-03  9.884146  9.904840  9.742288  9.766281  445138400
2011-01-04  9.935733  9.972022  9.841560  9.970222  309080800
2011-01-05  10.017007  10.027204  9.882047  9.883547  255519600
2011-01-06  10.008913  10.054499  9.984020  10.038604  300428800
2011-01-07  10.080569  10.087487  9.954027  10.016708  311931200
...
2021-12-23  172.658051  173.216346  171.668808  172.236893  68356600
2021-12-27  176.624817  176.712964  173.431804  173.451383  74919600
2021-12-28  175.606201  177.604295  174.861822  176.458336  79144300
2021-12-29  175.694382  176.916699  174.479854  175.645406  62348900
2021-12-30  174.538574  176.859899  174.430834  175.782484  59773000
2768 rows x 5 columns

```

Nasdaq Data Link. Gave me error message. After research this is No longer supported with free

```

File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Show command palette (Ctrl+Shift+P)
[1] On
import intrinio_sdk as intrinio
import pandas as pd

[2] On
intrinio.ApiClient().set_api_key("0jk5HDc4ZGE000dkNGjYmle7Jg22DE5GUWnQ50ME")
security_api = intrinio.SecurityApi()

[3] On
r = security_api.get_security_stock_prices(
    identifier="AAPL",
    start_date="2011-01-01",
    end_date="2021-12-31",
    frequency="daily",
    page_size=10000
)

... ApiException: (401) Unauthorized: HTTP response headers: HTTPHeadersDict([('Date', 'Sun, 25 Jan 2026 18:57:06 GMT'), ('Content-Type', 'application/json'), ('Content-Length', '104'), ('Connection', 'keep-alive'), ('Vary', 'Origin,Accept-Encoding')]) HTTP response body: {'human': 'No active subscription(s).', 'message': 'An active subscription is required to view this data.'}

[4] On
ApiException: (401)
Reason: Unauthorized
HTTP response headers: HTTPHeadersDict([('Date', 'Sun, 25 Jan 2026 18:57:06 GMT'), ('Content-Type', 'application/json'), ('Content-Length', '104'), ('Connection', 'keep-alive'), ('Vary', 'Origin,Accept-Encoding')])
HTTP response body: {'human': 'No active subscription(s).', 'message': 'An active subscription is required to view this data.'}

Next steps: ( Explain error )

```

Gemini

Please explain this error:

ApiException: (401) Reason: Unauthorized HTTP response headers: HTTPHeadersDict([('Date', 'Sun, 25 Jan 2026 18:57:06 GMT'), ('Content-Type', 'application/json'), ('Content-Length', '104'), ('Connection', 'keep-alive'), ('Vary', 'Origin,Accept-Encoding')]) HTTP response body: {'human': 'No active subscription(s).', 'message': 'An active subscription is required to view this data.'}

The error you're seeing, ApiException: (401) Unauthorized, means that your Intrinio API key doesn't have the necessary permissions or an active subscription to access the stock price data for Apple. You'll need to ensure your Intrinio account has an active subscription that covers the data you're trying to retrieve, or use a different data source if you don't have access.

Accept & Run Accept Cancel

What can I help you build?

+ Gemini 2.5 Flash ▶

Gemini can make mistakes so double-check it and use code with caution. [Learn more](#)

Intrinio error message. This task No longer supported with free account

SewellM01Lab.ipynb

```
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Successfully installed nasdaq-data-link-1.0.4
[36] In [1]
import pandas as pd
import nasdaqdatalink

nasdaqdatalink.ApiConfig.api_key = "8yytVbpyxh6Zp0XhITs"

df = nasdaqdatalink.get(dataset="NJKI/AAPL",
                         start_date="2011-01-01",
                         end_date="2023-12-31")

[36] Out[1]
JSONDecodeError: Traceback (most recent call last)
/usr/local/lib/python3.12/dist-packages/requests/models.py in json(self, **kwargs)
    975     try:
--> 976         return complexjson.loads(self.text, **kwargs)
    977     except JSONDecodeError as e:
...
12 frames
JSONDecodeError: Expecting value: line 1 column 1 (char 0)

During handling of the above exception, another exception occurred:

JSONDecodeError: Traceback (most recent call last)
JSONDecodeError: Expecting value: line 1 column 1 (char 0)

During handling of the above exception, another exception occurred:

DataLinkError: Traceback (most recent call last)
/usr/local/lib/python3.12/dist-packages/nasdaqdatalink/connection.py in parse(self, response)
    98     return response.json()
    99     except ValueError:
--> 100         raise DataLinkError(http_status=response.status_code, http_body=response.text)
    101
    102     @classmethod
DataLinkError: (Status 403) Something went wrong. Please try again. If you continue to have problems, please contact us at connect@data.nasdaq.com.
```

Variables Terminal 1:28 PM Python 3

Alpha Vantage

SewellM01Lab.ipynb

```
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
ALPHA_VANTAGE_API_KEY = "D7U7G3TGNQ1WPH2S"
crypto_api = CryptoCurrencies(key=ALPHA_VANTAGE_API_KEY,
                               output_format="pandas")

data, meta_data = crypto_api.get_digital_currency_daily(
    symbol="BTC",
    market="EUR"
)

[65] In [2]
crypto_api.get_digital_currency_exchange_rate(
    from_currency="BTC",
    to_currency="USD"
)[0].transpose()

[65] Out[2]
Realtime Currency Exchange Rate
1. From_Currency Code BTC
2. From_Currency Name Bitcoin
3. To_Currency Code USD
4. To_Currency Name United States Dollar
5. Exchange Rate 67292.53000000
6. Last Refreshed 2026-01-25 19:03:32
7. Time Zone UTC
8. Bid Price 67290.14200000
9. Ask Price 67295.78100000
```

Gemini

The error you're seeing, `ApiException: (401) Unauthorized`, means that your Intrinio API key doesn't have the necessary permissions or an active subscription to access the stock price data for Apple. You'll need to ensure your Intrinio account has an active subscription that covers the data you're trying to retrieve, or use a different data source if you don't have access.

Accept & Run Accept Cancel

How to print? Gemini 2.5 Flash

Gemini can make mistakes so double-check it and use code with caution. [Learn more](#)

Variables Terminal 2:03 PM Python 3

Alpha Vantage Theres more error message here as well. This is now a premium feature to view intraday data

The screenshot shows a Jupyter Notebook interface with two code cells. The first cell contains Python code for fetching intraday data from Alpha Vantage's API. The second cell shows an error message from the API indicating it's a premium endpoint.

```
File Edit View Insert Runtime Tools Help
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
9. Ask Price 86414.80300000
[36] 0s
import requests
import pandas as pd
from io import BytesIO

AV_API_URL = "https://www.alphavantage.co/query"
parameters = {
    "function": "CRYPTO_INTRADAY",
    "symbol": "ETH",
    "market": "USD",
    "interval": "5min",
    "outputsize": "full",
    "apikey": "D00G0T5MOLWH25"
}
r = requests.get(AV_API_URL, params=parameters)
data = r.json()

# Check if the API returned an error message instead of data:
if "Information" in data:
    print("API Error: [data['Information']]")
    df = pd.DataFrame() # Create an empty DataFrame or handle the error as appropriate
elif "Time Series Crypto (5min)" in data:
    df = pd.DataFrame(data["Time Series Crypto (5min)"])
    df = df.T
else:
    print("Unexpected API response format.")
    df = pd.DataFrame()

df
```

... error: Thank you for using Alpha Vantage! This is a premium endpoint. You may subscribe to any of the premium plans at <https://www.alphavantage.co/premium>

Variables Terminal

Gemini

J How to fetch historical crypto data without premium access?
You can use yfinance to fetch historical cryptocurrency data without needing a premium API key. I'll show you how to get daily historical data for Bitcoin (BTC-USD).

Accept & Run Accept Cancel

What can I help you build? Gemini 2.5 Flash

Gemini can make mistakes so double-check it and use code with caution. Learn more

6:50 PM Python 3

Able to use YFinance to view Bitcoin intraday Data

The screenshot shows a Jupyter Notebook interface with two code cells. The first cell uses the yfinance library to download Bitcoin price data. The second cell displays the resulting DataFrame.

```
File Edit View Insert Runtime Tools Help
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
[Gemini]
... Gemini
import yfinance as yf
btc_data = yf.download("BTC-USD", start="2020-01-01", end="2023-12-31")
display(btc_data.head())
... /tmp/ipython-input-2599484688.py:3: FutureWarning: YF.download() has changed argument auto_adjust default to True
btc_data = yf.download("BTC-USD", start="2020-01-01", end="2023-12-31")
[*****100%*****] 1 of 1 completed
Price Close High Low Open Volume
Ticker BTC-USD BTC-USD BTC-USD BTC-USD BTC-USD BTC-USD
Date
2020-01-01 7200.174316 7254.330566 7174.944336 7194.892090 18565664997
2020-01-02 6985.470215 7212.155273 6935.270020 7202.551270 20802083465
2020-01-03 7344.884277 7413.715332 6914.996094 6984.428711 26111481032
2020-01-04 7410.656738 7427.385742 7309.514160 7345.375488 18444271275
2020-01-05 7411.317383 7544.497070 7400.535645 7410.451660 19725074095
```

Variables Terminal

Gemini

J How to fetch historical crypto data without premium access?
You can use yfinance to fetch historical cryptocurrency data without needing a premium API key. I'll show you how to get daily historical data for Bitcoin (BTC-USD).

Accept & Run Accept Cancel

What can I help you build? Gemini 2.5 Flash

Gemini can make mistakes so double-check it and use code with caution. Learn more

6:50 PM Python 3

Earning Information using Alpha Vantage

SewellM01Lab.ipynb

```

File Edit Run all Tools Help
Commands + Code + Text Run all

[38] AV_API_URL = "https://www.alphavantage.co/query"
parameters = {
    "function": "EARNINGS_CALENDAR",
    "horizon": "3month",
    "apikey": "D3U7G3TSNOLW0H25"
}
r = requests.get(AV_API_URL, params=parameters)
pd.read_csv(BytesIO(r.content))

```

	symbol	name	reportDate	fiscalDateEnding	estimate	currency	timeOfDay
0	BKR	BAKER HUGHES CO - CLASS A	2026-01-25	2025-12-31	0.6700	USD	post-market
1	AGNC	AGNC INVESTMENT CORP	2026-01-26	2025-12-31	0.3700	USD	post-market
2	AGYS	AGILYSYS INC	2026-01-26	2025-12-31	0.3200	USD	post-market
3	AKO-A	EMBOTTELLADORA ANDINA S.A.	2026-01-26	2025-12-31	0.5627	USD	NaN
4	ALPIB	ALPIB	2026-01-26	2025-12-31	NaN	USD	NaN
...
9675	SRRRF	SRRRF	2026-04-24	2026-03-31	NaN	USD	NaN
9676	TBXXF	TBXXF	2026-04-24	2026-03-31	0.0900	USD	NaN
9677	WAMFF	WAMFF	2026-04-24	2026-03-31	-0.0100	USD	NaN
9678	WDFCF	WDFCF	2026-04-24	2026-03-31	NaN	USD	NaN
9679	YGTFF	YGTFF	2026-04-24	2026-03-31	NaN	USD	NaN

9680 rows x 7 columns

Bitcoin OHLC prices CoinGecko

SewellM01Lab.ipynb

```

File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all

[33] from pycoingecko import CoinGeckoAPI
from datetime import datetime
import pandas as pd

[34] cg = CoinGeckoAPI()

[35] ohlc = cg.get_coin_ohlc_by_id(
        id='bitcoin', vs_currency='usd', days='14'
    )
    ohlc_df = pd.DataFrame(ohlc)
    ohlc_df.columns = ["date", "open", "high", "low", "close"]
    ohlc_df["date"] = pd.to_datetime(ohlc_df["date"], unit="ms")
    ohlc_df

```

	date	open	high	low	close
0	2026-01-12 00:00:00	90569.0	90819.0	90245.0	90819.0
1	2026-01-12 04:00:00	90680.0	92356.0	90694.0	91818.0
2	2026-01-12 08:00:00	91684.0	92128.0	91283.0	91283.0
3	2026-01-12 12:00:00	91287.0	91425.0	90311.0	90562.0
4	2026-01-12 16:00:00	90521.0	91446.0	90129.0	91446.0
...
79	2026-01-25 04:00:00	89097.0	89168.0	88740.0	88746.0
80	2026-01-25 08:00:00	88825.0	88940.0	88545.0	88845.0
81	2026-01-25 12:00:00	88543.0	88601.0	88120.0	88413.0
82	2026-01-25 16:00:00	88593.0	88801.0	88514.0	88514.0
83	2026-01-25 20:00:00	88317.0	88317.0	86386.0	86395.0

84 rows x 5 columns

Next steps: Generate code with ohlc_df | New interactive sheet

Top 7 trending coins Coin Gecko

The screenshot shows a Jupyter Notebook interface with the title "SewellM01Lab.ipynb". The code cell contains Python code to fetch trending coins from Coin Gecko and display them in a DataFrame:

```
trending_coins = cg.get_search_trending()
pd.DataFrame([coin["item"] for coin in trending_coins["coins"]])
.drop(columns=["thumb", "small", "large"])
```

The resulting DataFrame is displayed in the notebook, showing the following data:

	id	coin_id	name	symbol	market_cap_rank	slug	price_btc	score	data
0	caldera	54475	Caldera	ERA	808	caldera	2.438777e-06	0	{'price': 0.21113695924895368, 'price_btc': 0.0...}
1	river	69318	River	RIVER	76	river	8.158096e-04	1	{'price': 70.26865471545308, 'price_btc': 0.0...}
2	bitcoin	1	Bitcoin	BTC	1	bitcoin	1.000000e+00	2	{'price': 86533.13093428001, 'price_btc': 1.0...}
3	pax-gold	9519	PAX Gold	PAXG	56	pax-gold	5.839623e-02	3	{'price': 5055.649178356551, 'price_btc': 0.0...}
4	seeker	70974	Seeker	SKR	367	seeker	2.588329e-07	4	{'price': 0.022404845790309205, 'price_btc': 0.0...}
5	nietzschean-penguin	71738	Nietzschean Penguin	PENGUIN	474	nietzschean-penguin	9.814154e-07	5	{'price': 0.08496596372369872, 'price_btc': 0.0...}
6	kinesis-silver	29789	Kinesis Silver	KAG	175	kinesis-silver	1.213620e-03	6	{'price': 105.0690605948046, 'price_btc': 0.0...}
7	tether-gold	10481	Tether Gold	XAUT	50	tether-gold	5.812757e-02	7	{'price': 5032.3895788357695, 'price_btc': 0.0...}
8	solana	4128	Solana	SOL	7	solana	1.373569e-03	8	{'price': 118.91659005691399, 'price_btc': 0.0...}
9	pudgy-penguins	52622	Pudgy Penguins	PENGU	139	pudgy-penguins	1.059442e-07	9	{'price': 0.00917211069657325, 'price_btc': 0.0...}
10	monero	69	Monero	XMR	21	monero	5.206182e-03	10	{'price': 450.72483047815023, 'price_btc': 0.0...}
11	lighter	71121	Lighter	LIT	173	lighter	1.812684e-05	11	{'price': 1.5693300500822736, 'price_btc': 0.0...}
12	nomina	69687	Nomina	NOM	694	nomina	1.822991e-07	12	{'price': 0.0157825568450395, 'price_btc': 0.0...}
13	dusk-network	5217	Dusk	DUSK	475	dusk	1.944501e-06	13	{'price': 0.16834503443111482, 'price_btc': 0.0...}
14	ripple	44	XRP	XRP	5	xrp	2.113536e-05	14	{'price': 1.8297921258046201, 'price_btc': 0.0...}

Bitcoin current price coingecko

The screenshot shows a Jupyter Notebook interface with the title "SewellM01Lab.ipynb". The code cell contains Python code to get the price of Bitcoin in USD:

```
cg.get_price(ids="bitcoin", vs_currencies="usd")
{'bitcoin': {'usd': 86562}}
```

The resulting output is displayed in the notebook, showing the current price of Bitcoin in USD:

```
{'price': 86562}
```