

```
In [1]: import requests
import pandas as pd

class ScriptData:

    def __init__(self, api_key):
        self.api_key = api_key
        self.base_url = 'https://www.alphavantage.co/query'

    def fetch_intraday_data(self, script):
        params = {'function': 'TIME_SERIES_INTRADAY',
                  'symbol': script,
                  'interval': '1min',
                  'apikey': self.api_key}
        response = requests.get(self.base_url, params=params)
        self.data = response.json()

    def convert_intraday_data(self, script):
        if not hasattr(self, 'data'):
            self.fetch_intraday_data(script)
        data = self.data['Time Series (1min)']
        df = pd.DataFrame(data).T
        df.index = pd.to_datetime(df.index)
        df = df.astype(float)
        df.index.name = 'timestamp'
        df.columns = ['open', 'high', 'low', 'close', 'volume']
        return df

    def __getitem__(self, key):
        return self.data[key]

    def __setitem__(self, key, value):
        self.data[key] = value

    def __contains__(self, key):
        return key in self.data
```

```
In [3]: api_key = 'FA7S0G4Q79GCW530'
sd = ScriptData(api_key)
sd.fetch_intraday_data('GOOGL')
df = sd.convert_intraday_data('GOOGL')
print(df)
```

	open	high	low	close	volume
timestamp					
2023-02-17 20:00:00	94.26	94.27	94.26	94.27	643.0
2023-02-17 19:59:00	94.27	94.28	94.26	94.26	1835.0
2023-02-17 19:58:00	94.27	94.27	94.27	94.27	1134.0
2023-02-17 19:57:00	94.28	94.28	94.25	94.25	1730.0
2023-02-17 19:56:00	94.28	94.28	94.28	94.28	716.0
...
2023-02-17 16:49:00	94.22	94.22	94.22	94.22	272.0
2023-02-17 16:48:00	94.21	94.21	94.21	94.21	571.0
2023-02-17 16:46:00	94.21	94.21	94.21	94.21	457.0
2023-02-17 16:43:00	94.20	94.21	94.20	94.21	536.0
2023-02-17 16:42:00	94.20	94.20	94.20	94.20	523.0

[100 rows x 5 columns]

```
In [4]: api_key = 'FA7S0G4Q79GCW530'
sd = ScriptData(api_key)
sd.fetch_intraday_data('AAPL')
```

```
df = sd.convert_intraday_data('AAPL')  
print(df)
```

timestamp	open	high	low	close	volume
2023-02-17 20:00:00	152.5900	152.6400	152.5900	152.6400	1864.0
2023-02-17 19:59:00	152.5601	152.5601	152.5601	152.5601	221.0
2023-02-17 19:58:00	152.5700	152.5700	152.5600	152.5600	618.0
2023-02-17 19:57:00	152.5600	152.5600	152.5600	152.5600	1185.0
2023-02-17 19:56:00	152.5500	152.5500	152.5500	152.5500	481.0
...
2023-02-17 17:18:00	152.4500	152.4500	152.4500	152.4500	980.0
2023-02-17 17:17:00	152.4500	152.4600	152.4300	152.4400	3268.0
2023-02-17 17:16:00	152.4800	152.5000	152.4300	152.5000	6825.0
2023-02-17 17:15:00	152.4500	152.4600	152.4200	152.4600	5596.0
2023-02-17 17:14:00	152.4799	152.4799	152.4600	152.4600	1813.0

[100 rows x 5 columns]

In []: