

---

**Avanza**

**Mar 26, 2020**



---

## Contents

---

<b>1</b>	<b>Security Notice</b>	<b>3</b>
1.1	API Reference . . . . .	3
1.1.1	Config . . . . .	3
1.1.2	Ticker . . . . .	4
1.1.3	Search . . . . .	5
1.1.4	News . . . . .	6
1.1.5	Collection . . . . .	6
1.1.6	Chartdata . . . . .	6
1.1.7	Version . . . . .	7
<b>2</b>	<b>Indices and tables</b>	<b>9</b>
	<b>Python Module Index</b>	<b>11</b>
	<b>Index</b>	<b>13</b>



tests	
package	

Based on <https://github.com/fhqvst/avanza>

Python wrapper for Unofficial Avanza API

Code is work in progress and is only meant as “proof of concept”

Authentication is done with selenium, which will save cookies in current working directory.

Short example of getting the current buyprice for msft stock:

```
import avanza

msft = avanza.Ticker(3873)
price = msft.buy_price
print(price)
> 166.48
```

Short example of displaying graph price of msft stock, using matplotlib:

```
import avanza
import matplotlib.pyplot as plt

df = avanza.ChartData().get_ticker_chartdata(3873)
df.plot(kind='line', x='timestamp', y='value')
plt.show()
```

More examples can be found [here](#)

[Documentation](#)



This python wrapper voids following bandit codes: B101 B108 B301

## 1.1 API Reference

### 1.1.1 Config

**class** `avanza.Config`

Bases: `object`

Config class for setting and retrieving settings

**settings**

**cookie\_path** path to file which should be used when loading/storing cookies

**classmethod** `get (name)`

Retrieves the value of setting

**Parameters** **name** (*str*) – The key which value should be returned

**classmethod** `set (name)`

Change settings using dict

#### Examples

```
>>> avanza.Config.set({'cookie_path': '/path/to/file'})
```

**Parameters** **name** (*dict*) – The new values to set

### 1.1.2 Ticker

**class** `avanza.Ticker` (*orderbook\_id*, *\*\*kwargs*)

Bases: `avanza.base.Base`

Request information about stock/certificate/fund/etc

**Parameters**

- **orderbook\_id** (*int*) – id of ticker
- **instrument** (*str*) – Type of instrument, Defaults to ‘stock’
- **auth** (*bool*) – Set true for additional information, Defaults to False

---

**Note:** Additional information if authenticated

---

**buy\_price**

Grabs buy price of ticker

**Returns** *float*

**change**

Grabs change price of ticker

**Returns** *float*

**change\_percent**

Grabs change price of ticker in percent

**Returns** *float*

**country**

Grabs the country of ticker

**Returns** *str*

**currency**

Grabs currency of ticker

**Returns** *str*

**flag\_code**

Grabs flag code of ticker

**Returns** *str*

**highest\_price**

Grabs highest price of ticker

**Returns** *float*

**id**

Grabs the id of ticker

**Returns** *int*

**info**

Grabs full json of ticker call

**Returns** *dict*

**isin**

Grabs ISIN of ticker



**Returns** *str*

**last\_price**  
Grabs last price of ticker

**Returns** *float*

**last\_price\_updated**  
Grabs last time price was updated

**Returns** *str* – ISO 8601

**lowest\_price**  
Grabs lowest price of ticker

**Returns** *float*

**marketplace**  
Grabs marketplace of ticker

**Returns** *str*

**name**  
Grabs full name of ticker

**Returns** *str*

**quote\_updated**  
Grabs last time quote was updated

**Returns** *str* – ISO 8601

**sell\_price**  
Grabs buy sell of ticker

**Returns** *float*

**symbol**  
Grabs symbol of ticker

**Returns** *str*

### 1.1.3 Search

**class** `avanza.Search` (*search\_query*)  
Bases: `avanza.base.Base`

**by\_instrument** (*instrument*)  
Grabs the results filtered by instrument type

**Parameters** **instrument** (*str*) – instrument type

**Returns** *list*

**count**  
Grabs total number of hits

**Returns** *int*

**first**  
Grabs the first result

**Returns** *Dict*

---

**Note:** Results are ordered by instrument, which means stock is the most likely result

---

**info**  
Grabs full json of ticker call

**Returns** *dict*

**results**  
Grabs the list of results

**Returns** *list*

### 1.1.4 News

**class** `avanza.News(index)`  
Bases: `avanza.base.Base`

Returns Avanza news

**Parameters** `index(int)` – The amount of news to be returned

**info**  
Grabs full json of returned news

**Returns** *dict*

**pretty**  
Prints the news in a prettier format

**Returns** *str*

**results**  
Grabs the list of results

**Returns** *list*

### 1.1.5 Collection

#### 1.1.6 Chartdata

**class** `avanza.chartdata.ChartData`  
Bases: `object`

Grab json chartdata and output as pandas DataFrame

**static** `get_distribution_chartdata()`  
Returns values from account distribution pie chart

**Returns** *pandas.core.frame.DataFrame* – Fallback to dict if pandas is not imported

---

**Note:** Authentication necessary

Will “unpack” original drilldown

---

**static** `get_overview_chartdata(time_period='one_month')`  
Returns chartdata from overview page

**Parameters** `time_period` (*str*) – time period

**Returns** *pandas.core.frame.DataFrame* – Fallback to dict if pandas is not imported

---

**Note:** Authentication necessary

---

**static** `get_ticker_chartdata` (*orderbook\_id*, *\*\*kwargs*)

Returns chartdata from overview page

**Parameters**

- `orderbook_id` (*int*) – Id of ticker
- `time_period` (*str*) – time period
- `chart_type` (*str*) – The kind of chartdata to retrieve
  - area: Data for typical line chart (default)
  - candlestick: Data for candlestick/ohlc chart
  - ohlc: Produces same result as candlestick
- `chart_resolution` (*str*) – resolution of chart

**Returns** *pandas.core.frame.DataFrame* – Fallback to dict if pandas is not imported

---

**Note:** Authentication necessary

---

### 1.1.7 Version

`avanza.__version__`

Get the current version in string format



## CHAPTER 2

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



### **a**

`avanza.chartdata`, 6  
`avanza.collection`, 6





## A

avanza.\_\_version\_\_ (in *module* *avanza.\_\_version\_\_*), 7  
 avanza.chartdata (*module*), 6  
 avanza.collection (*module*), 6

## B

buy\_price (*avanza.Ticker attribute*), 4  
 by\_instrument() (*avanza.Search method*), 5

## C

change (*avanza.Ticker attribute*), 4  
 change\_percent (*avanza.Ticker attribute*), 4  
 ChartData (*class in avanza.chartdata*), 6  
 Config (*class in avanza*), 3  
 count (*avanza.Search attribute*), 5  
 country (*avanza.Ticker attribute*), 4  
 currency (*avanza.Ticker attribute*), 4

## F

first (*avanza.Search attribute*), 5  
 flag\_code (*avanza.Ticker attribute*), 4

## G

get() (*avanza.Config class method*), 3  
 get\_distribution\_chartdata()  
     (*avanza.chartdata.ChartData static method*), 6  
 get\_overview\_chartdata()  
     (*avanza.chartdata.ChartData static method*), 6  
 get\_ticker\_chartdata()  
     (*avanza.chartdata.ChartData static method*), 7

## H

highest\_price (*avanza.Ticker attribute*), 4

## I

id (*avanza.Ticker attribute*), 4  
 info (*avanza.News attribute*), 6  
 info (*avanza.Search attribute*), 6

info (*avanza.Ticker attribute*), 4  
 isin (*avanza.Ticker attribute*), 4

## L

last\_price (*avanza.Ticker attribute*), 5  
 last\_price\_updated (*avanza.Ticker attribute*), 5  
 lowest\_price (*avanza.Ticker attribute*), 5

## M

marketplace (*avanza.Ticker attribute*), 5

## N

name (*avanza.Ticker attribute*), 5  
 News (*class in avanza*), 6

## P

pretty (*avanza.News attribute*), 6

## Q

quote\_updated (*avanza.Ticker attribute*), 5

## R

results (*avanza.News attribute*), 6  
 results (*avanza.Search attribute*), 6

## S

Search (*class in avanza*), 5  
 sell\_price (*avanza.Ticker attribute*), 5  
 set() (*avanza.Config class method*), 3  
 settings (*avanza.Config attribute*), 3  
 symbol (*avanza.Ticker attribute*), 5

## T

Ticker (*class in avanza*), 4