

## getTicker: Documentation

getTicker allows you to rapidly get the ticker symbol directly related to any organization.

To do this, getTicker is composed of two web-scraping bots: call them Wiki and Crunch. Wiki attempts to fetch the organization's ticker using the data available on [www.wikipedia.org](http://www.wikipedia.org) , and Crunch does the same but on [www.crunchbase.com](http://www.crunchbase.com) .

getTicker takes two arguments: merchant and first.

For merchant, the user may enter any organization name as a string (such as a company, a brand, or a merchant) for which he wishes to know the ticker symbol that is most closely related to it.

For first, the user must enter either **w** or **c** which are simply variables that contain strings ('wiki' and 'crunch'). This will tell getTicker which is the priority bot and which is the backup bot. The priority bot is always launched first, and if it is unsuccessful, the backup bot is launched.

getTicker has only one output (see signature below.)

### **Signature:**

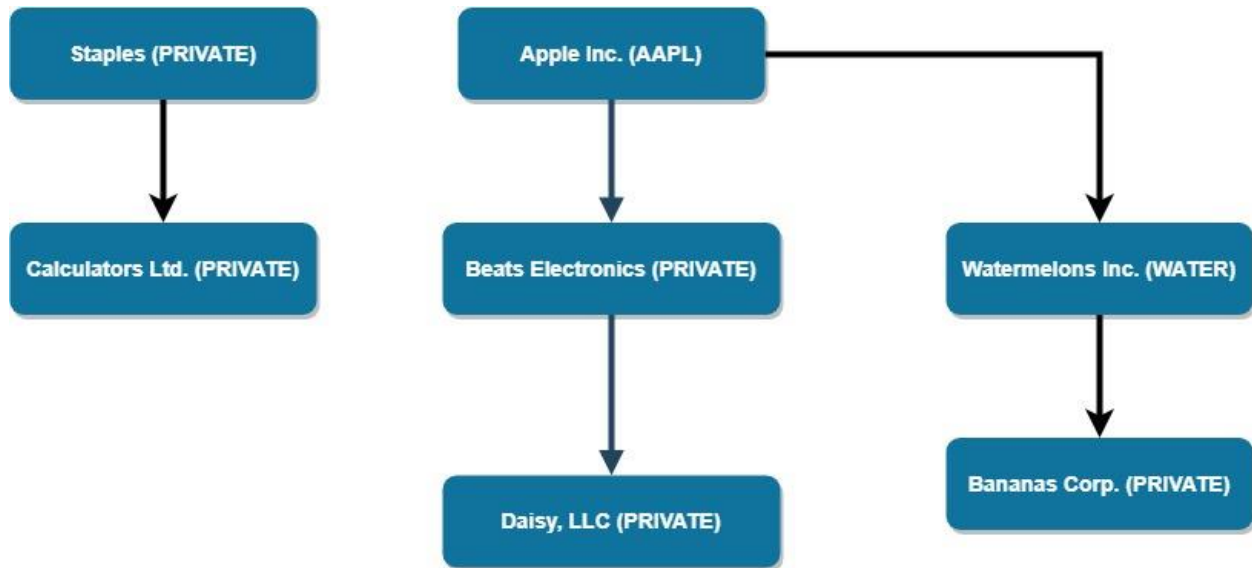
string (e.g: Apple, ticketmaster.com), variable (w or c) → string (ticker symbol, 'PRIVATE', or 'ERR')

getTicker will return 'PRIVATE' if the organization is private and if it has any parent organizations, they are all private as well.

getTicker will return the ticker symbol of the organization if itself is publicly traded. If that organization is private but at least one of its parent organizations is public, getTicker will return the ticker symbol of the first parent organization (in hierarchal order starting from the organization that is the input of the user) that is publicly traded.

getTicker will return 'ERR' (for ERROR) whenever either bots were able to determine a ticker symbol directly related to the organization inputted by the user.

### Examples:



```
In [6]: getTicker('Apple Inc.', w)
```

```
Out[6]: 'AAPL'
```

```
In [7]: getTicker('Beats Electronics', w)
```

```
Out[7]: 'AAPL'
```

```
In [8]: getTicker('Daisy, LLC', w)
```

```
Out[8]: 'AAPL'
```

```
In [9]: getTicker('Watermelons Inc.', w)
```

```
Out[9]: 'WATER'
```

```
In [10]: getTicker('Bananas Corp.', w)
```

```
Out[10]: 'WATER'
```

```
In [11]: getTicker('Staples', c)
```

```
Out[11]: 'PRIVATE'
```

```
In [12]: getTicker('Calculators Ltd.', c)
```

```
Out[12]: 'PRIVATE'
```

You do not need to be extremely precise when entering the organization name. For example, `getTicker(apple)` would return `AAPL`, no need to write `getTicker(Apple Inc.)`. However, the more precise the better, especially when the organizations are not as popular as Apple.

**Important Note:** getTicker uses data from Wikipedia and Crunchbase to determine the ticker of the organization. While these are both great resources, there are a few important issues:

- 1- Be careful with common or homophonic organization names such as 'hair.com' or 'philosophy'. The bot simply selects the most popular result of the user's input the respective database, regardless of whether the organization actually exists in the database.
- 2- Wikipedia has inconsistent structure. This means that when getTicker fetches tickers, parent companies, and html webpages, it looks for usual patterns that are on many other Wikipedia pages. However, some pages may be unique or different from most pages, causing getTicker to be confused.
- 3- Wikipedia is community led (for the most part). Wikipedia contains information which is not supervised by any moderator, and therefore may be misleading.

Problem 1 is very complex to counter but could be optimized using fuzzy string matching.

Problem 2 and 3 can be dealt with by making the Crunch bot the backup of the Wiki bot (meaning setting first = w). But this order can be changed in order to optimize results depending on the situation.

#### **Other useful functions:**

recentAC(merchant): string → pandas.DataFrame

returns a dataframe of some of the recent acquisitions made by merchant (beta version).

getTickerW(merchant): string → string

Exactly like getTicker but only attempts to fetch the ticker on Wikipedia (meaning no backup bot).

getTickerC(merchant): string → string

Exactly like getTicker but only attempts to fetch the ticker on Crunchbase (meaning no backup bot).

#### **Lower level functions:**

getHTMLW(merchant): string → bs4.BeautifulSoup

returns the HTML of the merchant's Wikipedia page.

getHTMLC(merchant): string → bs4.BeautifulSoup

returns the HTML of the merchant's Crunchbase page.

findBoxW(html, ret): bs4.BeautifulSoup, bool or string → bs4.BeautifulSoup or bool or string

returns the HTML of the information box on the Wikipedia page, returns whatever ret is if it can't find it.

isPrivateW(html): bs4.BeautifulSoup → bool

isPrivateC(html): bs4.BeautifulSoup → bool

returns True if the organization is private, False otherwise.

isTradedW(html): bs4.BeautifulSoup → string or bool

isTradedC(html): bs4.BeautifulSoup → string or bool

returns the ticker symbol of the organization if public, False otherwise.

getParentW(html) bs4.BeautifulSoup → bs4.BeautifulSoup or string

getParentC(html) bs4.BeautifulSoup → bs4.BeautifulSoup or string

returns the HTML of the Wikipedia or Crunchbase page of the organization's parent company. Returns 'ERR' if it does not exist.