

## Trending Python Modules

```
In [1]: from bs4 import BeautifulSoup
import requests
import pandas as pd

In [2]: source = requests.get('https://www.simplilearn.com/top-python-libraries-for-data-science-article')

soup = BeautifulSoup(source.text, 'html.parser')
```

## Top 20 Python Modules

```
In [3]: movies = soup.find('article').find('ul').find_all('li')
movies

Out[3]: [<li aria-level="1">TensorFlow</li>,
<li aria-level="1">NumPy</li>,
<li aria-level="1">SciPyÂ </li>,
<li aria-level="1">Pandas</li>,
<li aria-level="1">MatplotlibÂ </li>,
<li aria-level="1">Keras</li>,
<li aria-level="1">SciKit-Learn</li>,
<li aria-level="1">PyTorch</li>,
<li aria-level="1">Scrapy</li>,
<li aria-level="1">BeautifulSoup</li>,
<li aria-level="1">LightGBM</li>,
<li aria-level="1">ELIS</li>,
<li aria-level="1">Theano</li>,
<li aria-level="1">NuPIC</li>,
<li aria-level="1">Ramp</li>,
<li aria-level="1">Pipenv</li>,
<li aria-level="1">Bob</li>,
<li aria-level="1">PyBrain</li>,
<li aria-level="1">Caffe2</li>,
<li aria-level="1">Chainer</li>]

In [4]: counter = 1
top20_modules = {}
key = []
value = []
for movie in movies:
    key.append(counter)
    value.append(movie.get_text())
    counter += 1

top20_modules =dict(zip(value, key))
top20_modules

Out[4]: {'TensorFlow': 1,
'NumPy': 2,
'SciPyÂ\xa0': 3,
'Pandas': 4,
'MatplotlibÂ\xa0': 5,
'Keras': 6,
'SciKit-Learn': 7,
'PyTorch': 8,
'Scrapy': 9,
'BeautifulSoup': 10,
'LightGBM': 11,
'ELIS': 12,
'Theano': 13,
'NuPIC': 14,
'Ramp': 15,
'Pipenv': 16,
'Bob': 17,
'PyBrain': 18,
'Caffe2': 19,
'Chainer': 20}

In [5]: #dataframe = pd.DataFrame({'Module_Name': value, 'Rank':key})
dataframe = pd.DataFrame(list(top20_modules.items()), columns = ['Python Module', 'Rank'])
dataframe

Out[5]:
```

	Python Module	Rank
0	TensorFlow	1
1	NumPy	2
2	SciPyÂ	3
3	Pandas	4
4	MatplotlibÂ	5
5	Keras	6
6	SciKit-Learn	7
7	PyTorch	8
8	Scrapy	9
9	BeautifulSoup	10
10	LightGBM	11
11	ELIS	12
12	Theano	13
13	NuPIC	14
14	Ramp	15
15	Pipenv	16
16	Bob	17
17	PyBrain	18
18	Caffe2	19
19	Chainer	20

```
In [6]: dataframe.dtypes

Out[6]: Python Module      object
Rank          int64
dtype: object
```

## Brief Details for Top 5 Modules

```
In [7]: content_tag = soup.find_all('article')[1].find_all('ul')[1:11]
content_tag

Out[7]: [<ul>
<li aria-level="1">Better computational graph visualizations</li>
<li aria-level="1">Reduces error by 50 to 60 percent in neural machine learning</li>
<li aria-level="1">Parallel computing to execute complex models</li>
<li aria-level="1">Seamless library management backed by Google</li>
<li aria-level="1">Quicker updates and frequent new releases to provide you with the latest featuresÂ </li>
</ul>,
<ul>
<li aria-level="1">Speech and image recognitionÂ </li>
<li aria-level="1">Text-based applicationsÂ </li>
<li aria-level="1"><a href="https://www.simplilearn.com/tutorials/python-tutorial/time-series-analysis-in-python" rel="noopener" target="_blank" title="Time-series analysis">Time-series ana
lysis</a></li>
<li aria-level="1">Video detection</li>
</ul>,
<ul>
<li aria-level="1">Collection of algorithms and functions built on the NumPy extension of Python</li>
<li aria-level="1">High-level commands for <a href="https://www.simplilearn.com/data-visualization-article" rel="noopener" target="_blank" title="data manipulation and visualization">data m
anipulation and visualization</a>
</li>
<li aria-level="1">Multidimensional image processing with the SciPy ndimage submodule</li>
<li aria-level="1">Includes built-in functions for solving differential equations</li>
</ul>,
<ul>
<li aria-level="1">Multidimensional image operations</li>
<li aria-level="1">Solving differential equations and the Fourier transform</li>
<li aria-level="1">Optimization algorithms</li>
<li aria-level="1">Linear algebra</li>
</ul>,
<ul>
<li aria-level="1">Provides fast, precompiled functions for numerical routines</li>
<li aria-level="1">Array-oriented computing for better efficiency</li>
<li aria-level="1">Supports an object-oriented approach</li>
<li aria-level="1">Compact and faster computations with vectorization</li>
</ul>,
<ul>
<li aria-level="1">Extensively used in data analysisÂ </li>
<li aria-level="1">Creates powerful N-dimensional array</li>
<li aria-level="1">Forms the base of other libraries, such as SciPy and <a href="https://www.simplilearn.com/tutorials/scikit-learn-tutorial/what-is-scikit-learn-and-how-to-install-it" rel
="noopener" target="_blank" title="scikit-learn">scikit-learn</a>
</li>
<li aria-level="1">Replacement of MATLAB when used with SciPy and matplotlib</li>
</ul>,
<ul>
<li aria-level="1">Eloquent syntax and rich functionalities that gives you the freedom to deal with missing data</li>
<li aria-level="1">Enables you to create your own function and run it across a series of data</li>
<li aria-level="1">High-level abstraction</li>
<li aria-level="1">Contains high-level data structures and manipulation tools</li>
</ul>,
<ul>
<li aria-level="1">General <a href="https://www.simplilearn.com/data-wrangling-article" rel="noopener" target="_blank" title="data wrangling">data wrangling</a> and <a href="https://www.sim
plilearn.com/data-cleaning-why-and-how-to-get-started-article" rel="noopener" target="_blank" title="data cleaning">data cleaning</a>
</li>
<li aria-level="1">ETL (extract, transform, load) jobs for data transformation and data storage, as it has excellent support for loading CSV files into its data frame format</li>
<li aria-level="1">Used in a variety of academic and commercial areas, including statistics, finance and neuroscienceÂ </li>
<li aria-level="1">Time-series-specific functionality, such as date range generation, moving window, linear regression and date shifting.</li>
</ul>,
<ul>
<li aria-level="1">Usable as a MATLAB replacement, with the advantage of being free and open sourceÂ </li>
<li aria-level="1">Supports dozens of backends and output types, which means you can use it regardless of which operating system youâre using or which output format you wish to use</li>
<li aria-level="1">Pandas itself can be used as wrappers around MATLAB API to drive MATLAB like a cleaner</li>
<li aria-level="1">Low memory consumption and better runtime behavior</li>
</ul>,
<ul>
<li aria-level="1">Correlation analysis of variables</li>
<li aria-level="1">Visualize 95 percent confidence intervals of the models</li>
<li aria-level="1">Outlier detection using a scatter plot etc.</li>
<li aria-level="1">Visualize the distribution of data to gain instant insights</li>
</ul>]

In [8]: features = []
applications = []
for i in range(0, len(content_tag)-1, 2):
    features.append(content_tag[i].text.rstrip().replace('\n', '. '))
for i in range(1, len(content_tag), 2):
    applications.append(content_tag[i].text.rstrip().replace('\n', '. '))

In [9]: features[0]

Out[9]: '. Better computational graph visualizations. Reduces error by 50 to 60 percent in neural machine learning. Parallel computing to execute complex models. Seamless library management backed b
y Google. Quicker updates and frequent new releases to provide you with the latest features'
```

```
In [10]: applications[0]

Out[10]: '. Speech and image recognitionÂ\xa0. Text-based applicationsÂ\xa0. Time-series analysis. Video detection'
```

```
In [11]: features[-1]

Out[11]: '. Usable as a MATLAB replacement, with the advantage of being free and open sourceÂ\xa0. Supports dozens of backends and output types, which means you can use it regardless of which operati
ng system youâre using or which output format you wish to use. Pandas itself can be used as wrappers around MATLAB API to drive MATLAB like a cleaner. Low memory consumption and bett
er runtime behavior'
```

```
In [12]: applications[-1]

Out[12]: '. Correlation analysis of variables. Visualize 95 percent confidence intervals of the models. Outlier detection using a scatter plot etc.. Visualize the distribution of data to gain instant
insights'
```

```
In [13]: key = key[0:5]
value = value[0:5]

key, value

Out[13]: ([1, 2, 3, 4, 5],
['TensorFlow', 'NumPy', 'SciPyÂ\xa0', 'Pandas', 'MatplotlibÂ\xa0'])

In [14]: df = pd.DataFrame({'Module_Name':value, 'Rank':key, 'Features':features, 'Applications':applications})
df

Out[14]:
```

	Module_Name	Rank	Features	Applications
0	TensorFlow	1	. Better computational graph visualizations. R...	. Speech and image recognitionÂ . Text-based a...
1	NumPy	2	. Collection of algorithms and functions built...	. Multidimensional image operations. Solving d...
2	SciPyÂ	3	. Provides fast, precompiled functions for num...	. Extensively used in data analysisÂ . Creates...
3	Pandas	4	. Eloquent syntax and rich functionalities tha...	. General data wrangling and data cleaning. ...
4	MatplotlibÂ	5	. Usable as a MATLAB replacement, with the adv...	. Correlation analysis of variables. Visualize...

```
In [15]: df.dtypes

Out[15]: Module_Name      object
Rank          int64
Features      object
```

Applications object  
dtype: object