Wikipedia

Wikipedia is free crowd-sourced encyclopedia. It is constantly updated and has millions of articles. Wikipedia also has a simple API that's been wrapped for Python in the **wikipedia** package. Let's install it and see how we might use it!

```
pip install wikipedia
```

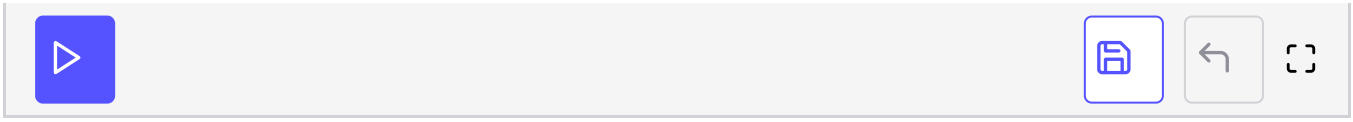
The wikipedia package depends on BeautifulSoup and the requests package. Fortunately, it will install them or upgrade them itself if they are missing or out-of-date. Now that we have wikipedia installed, let's try asking it what it knows about the Python:

```
import wikipedia

print(wikipedia.search('Python'))
#['Python', 'Python (programming language)',
#'Monty Python', 'Burmese python',
#'Python molurus', 'Python (missile)',
#'Ball python', 'Python curtus',
#'African rock python', 'History of Python']
```

Well, that was slightly unexpected. If we just search for the word "Python" on Wikipedia, it is going to return a list of items. So we need to be more specific. That second item in the list looks like the one we want, so let's try grabbing its **summary** information:

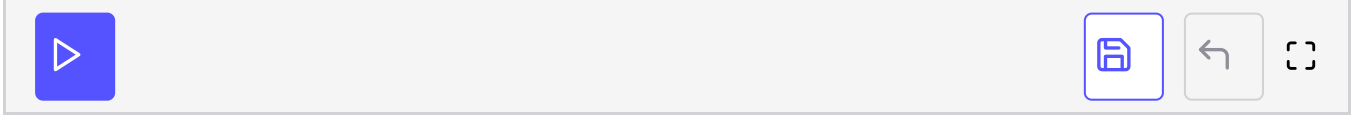
```
import wikipedia

print(wikipedia.summary('Python (programming language)'))
#('Python is a widely used high-level, general-purpose, interpreted, dynamic '
# 'programming language. Its design philosophy emphasizes code readability, and '
# 'its syntax allows programmers to express concepts in fewer lines of code '
```

```
# 'than possible in languages such as C++ or Java. The language provides '
# 'constructs intended to enable clear programs on both a small and large '
# 'scale.\n'
# 'Python supports multiple programming paradigms, including object-oriented, '
# 'imperative and functional programming or procedural styles. It features a '
# 'dynamic type system and automatic memory management and has a large and '
# 'comprehensive standard library.\n'
# 'Python interpreters are available for many operating systems, allowing '
# 'Python code to run on a wide variety of systems. Using third-party tools, '
# 'such as Py2exe or Pyinstaller, Python code can be packaged into stand-alone '
# 'executable programs for some of the most popular operating systems, so '
# 'Python-based software can be distributed to, and used on, those environments '
# 'with no need to install a Python interpreter.\n'
# 'CPython, the reference implementation of Python, is free and open-source '
# 'software and has a community-based development model, as do nearly all of '
# 'its variant implementations. CPython is managed by the non-profit Python '
# 'Software Foundation.')
```



That looks better. Now we know how to grab summary information about something that we searched, so let's use that knowledge to make a function we can use to query Wikipedia:

```
import wikipedia


def print_wikipedia_results(word):
    """
    Searches for pages that match the specified word
    """
    results = wikipedia.search(word)

    for result in results:
        try:
            page = wikipedia.page(result)
        except wikipedia.exceptions.DisambiguationError:
            print('DisambiguationError')
            continue
        except wikipedia.exceptions.PageError:
            print('PageError for result: ' + result)
            continue

        print(page.summary.encode('utf-8'))

if __name__ == '__main__':
    print_wikipedia_results('wombat')
```



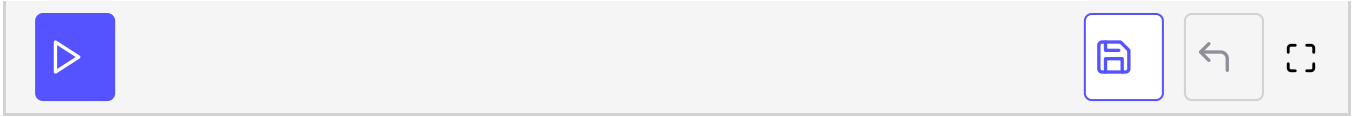
Our function, **print_wikipedia_results**, takes in the word or phrase that we'd like to look up and returns a list of results. We then loop over the results and

attempt to create a **WikipediaPage** instance. However if you pass an

ambiguous word or phrase to the **page** method, then it will raise a **DisambiguationError**. So we catch that and ignore it. We also catch **PageErrors**, although you might want to just raise an error on those as they seem a bit more serious. When we do get a valid WikipediaPage object back though, we go ahead and print out its summary.

Let's look briefly at some of the properties we can access on our WikipediaPage object:

```
import wikipedia

page = wikipedia.page('Python (programming language)')
print(page)
#<WikipediaPage 'Python (programming language)'>

print(page.title)
#'Python (programming language)'

print(page.url)
#'https://en.wikipedia.org/wiki/Python_(programming_language)'

print(page.content.encode('utf-8'))
#('Python is a widely used high-level, general-purpose, interpreted, dynamic ' ...
```

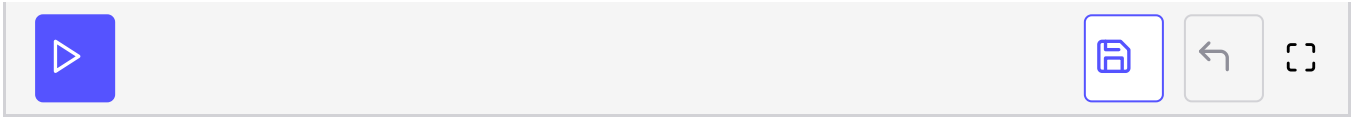


Here we show how you can get the title, url and content of the article. You can also get the links, images, html and more too.

Another fun feature of the wikipedia package is that it allows you to set what language the results should be in. Let's take a look:

```
import wikipedia

print(wikipedia.set_lang("fr"))
None

page = wikipedia.page('Python (programming language)')
print(page.summary.encode('utf-8'))
#('Python est un langage de programmation objet, multi-paradigme et '
# 'multiplateformes. Il favorise la programmation impérative structurée, '
# "fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, "
# "d'une gestion automatique de la mémoire par ramasse-miettes et d'un système "
# "de gestion d'exceptions ; il est ainsi similaire à Perl, Ruby, Scheme, "
# 'Smalltalk et Tcl.\n'
# 'Le langage Python est placé sous une licence libre proche de la licence BSD '
```

```
# 'et fonctionne sur la plupart des plates-formes informatiques, des '
# 'supercalculateurs aux ordinateurs centraux, de Windows à Unix en passant par '
# 'GNU/Linux, Mac OS, ou encore Android, iOS, et aussi avec Java ou encore '

# '.NET. Il est conçu pour optimiser la productivité des programmeurs en '
# 'offrant des outils de haut niveau et une syntaxe simple à utiliser.\n'
# 'Il est également apprécié par les pédagogues qui y trouvent un langage où la '
# 'syntaxe, clairement séparée des mécanismes de bas niveau, permet une '
# 'initiation aisée aux concepts de base de la programmation.')
```



Here we told the wikipedia package that we want the language to be French, then we grabbed the summary of our Python page again. As you can see, it's now in French.