

---

**WOS**

***Release 0.2.7***

**Dec 27, 2021**



---

## Contents:

---

<b>1</b>	<b>Description</b>	<b>3</b>
<b>2</b>	<b>Installation</b>	<b>5</b>
<b>3</b>	<b>Documentation</b>	<b>7</b>
<b>4</b>	<b>Usage</b>	<b>9</b>
<b>5</b>	<b>Example</b>	<b>11</b>
<b>6</b>	<b>APIs</b>	<b>13</b>
<b>7</b>	<b>[FAQ] I cannot connect ...</b>	<b>15</b>
<b>8</b>	<b>Disclaimer</b>	<b>17</b>
<b>9</b>	<b>Indices and tables</b>	<b>19</b>
9.1	wos.client.WosClient . . . . .	19
9.2	wos.utils . . . . .	22
	<b>Python Module Index</b>	<b>23</b>
	<b>Index</b>	<b>25</b>



*SOAP Client for querying the Web of Science database*



# CHAPTER 1

---

## Description

---

Web of Science (previously Web of Knowledge) is an online subscription-based scientific citation indexing service maintained by Clarivate.

`wos` is a python SOAP Client (both API and command-line tool) to query the WOS database in order to get XML data from a query using the WWS access.





## CHAPTER 2

---

### Installation

---

The package has been uploaded to [PyPI](#), so you can install the package using pip:

pip install wos



## CHAPTER 3

---

### Documentation

---

This README and the documentation for the classes and methods can be accessed on [ReadTheDocs](#).



## CHAPTER 4

---

### Usage

---

You can use the `wos` command to query the Web of Science API. If you want to access data that needs to be accessed using the premium API, you also have to authenticate using your username and password.

**usage:** `wos [-h] [--close] [-l] [-u USER] [-p PASSWORD] [-s SID] {query,doi,connect} ...`

Query the Web of Science.

**positional arguments:**

`{query,doi,connect}` **sub-command help** `query` query the Web of Science. `doi` get the WOS ID from the DOI. `connect` connect and get an SID.

**optional arguments:**

<b>-h, --help</b>	show this help message and exit
<b>--close</b>	Close session.
<b>--proxy PROXY</b>	HTTP proxy
<b>--timeout TIMEOUT</b>	API timeout
<b>-l, --lite</b>	Wos Lite
<b>-v, --verbose</b>	Verbose

**authentication:** API credentials for premium access.

`-u USER, --user USER -p PASSWORD, --password PASSWORD -s SID, --sid SID`

You can use the WOS Lite API using the `--lite` parameter (for each query).

You can also authenticate using the session id (SID). In fact the sessions are not closed by the command line utility. Example:

```
$ wos --user JohnDoe --password 12345 connect
Authenticated using SID: ABCDEFGHIJKLM

$ wos --sid ABCDEFGHIJKLM query 'AU=Knuth Donald' -cl
```

(continues on next page)

(continued from previous page)

```
Authenticated using SID: ABCDEFGHIJKLM
<?xml version="1.0" ?>
<records>
  <REC r_id_disclaimer="ResearcherID data provided by Clarivate Analytics">
    <UID>WOS:000287850200007</UID>
    <static_data>
      <summary>
        <EWUID>
          <WUID coll_id="WOS"/>
          <edition value="WOS.SCI"/>
        </EWUID>
        <pub_info coverdate="MAR 2011" has_abstract="N" issue="1"
          pubmonth="MAR" pubtype="Journal" pubyear="2011"
          sortdate="2011-03-01" vol="33">
          <page begin="33" end="45" page_count="13">33-45</page>
        </pub_info>
        <titles count="6">
          <title type="source">MATHEMATICAL INTELLIGENCER</title>
        </titles>
      </summary>
    </static_data>
  </REC>
  ....
$ wos --sid ABCDEFGHIJKLM doi '10.1007/s00283-010-9170-7'
10.1007/s00283-010-9170-7
```

Check the [user\\_query](#) documentation to understand how to create query strings.

## CHAPTER 5

---

### Example

---

Obviously you can also use the python client programmatically:

```
from wos import WosClient
import wos.utils

with WosClient('JohnDoe', '12345') as client:
    print(wos.utils.query(client, 'AU=Knuth Donald'))
```





In `wos 0.1.11+`, the `WosClient` class can access the following APIs.

- `retrieve` [[lite](#) / [premium](#)]
- `retrieveById` [[lite](#) / [premium](#)]
- `search` [[lite](#) / [premium](#)]
- `citedReferences` [[premium](#)]
- `citedReferencesRetrieve` [[premium](#)]
- `citingArticles` [[premium](#)]
- `relatedRecords` [[premium](#)]



---

### [FAQ] I cannot connect . . .

---

I am not affiliated with Clarivate. The library leverages the Web of Science [WWS](#) API (Web Services Premium or Lite), which is a paid service offered by Clarivate. This means that your institution has to pay for the Web of Science Core Collection access. The simple registration to Web of Knowledge / Web of Science does not entitle you to access the WWS API service.

So if you receive errors like `No matches returned for Username` or `No matches returned for IP`, these errors are thrown directly by the WWS API server. This means that the library is correctly communicating with the server, but you do not have access to the Web Services API. I do understand that you can access the WOS website from your network, but the website access and the API access (used in this project) are two separated products, and the website access does not imply the API access, since Clarivate bills them separately. This project does not scrape the website (which would violate the terms of usage) but invokes the WWS APIs offered by Clarivate. Thus there is nothing this project can do to help you.

**If you think this is an error and you should be entitled to access the services, please contact Clarivate support first and verify if you have the WWS access. Please open an issue ONLY when you have (1) verified with Clarivate support that you have WWS access; (2) verified that you are connected from the correct network.**



## CHAPTER 8

---

### Disclaimer

---

All product names, trademarks, and registered trademarks are the property of their respective owners. All company, product, and service names used in this document are for identification purposes only. The use of these names, trademarks, and brands do not constitute an endorsement or recommendation by the companies.

---



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

## 9.1 `wos.client.WosClient`

Here is the documentation for the methods in the `wos.client.WosClient` class.

---

```
class wos.client.WosClient (user=None, password=None, SID=None, close_on_exit=True,  
                             lite=False, proxy=None, timeout=600, throttle=(2, 1))
```

Query the Web of Science. You must provide user and password only to user premium WWS service.

**with** `WosClient()` **as** `wos`: `results = wos.search(...)`

```
citedReferences (uid, count=100, offset=1, retrieveParameters=None)
```

The `citedReferences` operation returns references cited by an article identified by a unique identifier. You may specify only one identifier per request.

**Uid** Web of Science unique record identifier

**Count** Number of records to display in the result. Cannot be less than 0 and cannot be greater than 100. If count is 0 then only the summary information will be returned.

**Offset** First record in results to return. Must be greater than zero

**RetrieveParameters** Retrieve parameters. If omitted the result of `make_retrieveParameters(offset, count, 'RS', 'D')` is used.

**citedReferencesRetrieve** (*queryId, count=100, offset=1, retrieveParameters=None*)

The citedReferencesRetrieve operation submits a query returned by a previous citedReferences operation.

This operation is useful for overcoming the retrieval limit of 100 records per query. For example, a citedReferences operation may find 106 cited references, as revealed by the content of the recordsFound element, but it returns only records 1-100. You could perform a subsequent citedReferencesretrieve operation to obtain records 101-106.

**QueryId** The query ID from a previous citedReferences operation

**Count** Number of records to display in the result. Cannot be less than 0 and cannot be greater than 100. If count is 0 then only the summary information will be returned.

**Offset** First record in results to return. Must be greater than zero

**RetrieveParameters** Retrieve parameters. If omitted the result of make\_retrieveParameters(offset, count, 'RS', 'D') is used.

**citingArticles** (*uid, count=100, offset=1, editions=None, timeSpan=None, retrieveParameters=None*)

The citingArticles operation finds citing articles for the article specified by unique identifier. You may specify only one identifier per request. Web of Science Core Collection (WOS) is the only valid database for this operation.

**Uid** A unique item identifier. It cannot be None or empty string.

**Count** Number of records to display in the result. Cannot be less than 0 and cannot be greater than 100. If count is 0 then only the summary information will be returned.

**Offset** First record in results to return. Must be greater than zero

**Editions** List of editions to be searched. If None, user permissions will be substituted.

Fields: collection - Name of the collection edition - Name of the edition

**TimeSpan** This element defines specifies a range of publication dates. If timeSpan is null, then the maximum time span will be inferred from the editions data.

Fields: begin - Beginning date for this search. Format: YYYY-MM-DD end - Ending date for this search. Format: YYYY-MM-DD

**RetrieveParameters** Retrieve parameters. If omitted the result of make\_retrieveParameters(offset, count, 'RS', 'D') is used.

**close** ()

The close operation loads the session if it is valid and then closes it and releases the session seat. All the session data are deleted and become invalid after the request is processed. The session ID can no longer be used in subsequent requests.

**connect** ()

Authenticate to WOS and set the SID cookie.

**is\_lite** ()

Returns True if the client is for WOS lite

**static make\_retrieveParameters** (*offset=1, count=100, name='RS', sort='D'*)

Create retrieve parameters dictionary to be used with APIs.

**Count** Number of records to display in the result. Cannot be less than 0 and cannot be greater than 100. If count is 0 then only the summary information will be returned.

**Offset** First record in results to return. Must be greater than zero



**Name** Name of the field to order by. Use a two-character abbreviation to specify the field ('AU': Author, 'CF': Conference Title, 'CG': Page, 'CW': Source, 'CV': Volume, 'LC': Local Times Cited, 'LD': Load Date, 'PG': Page, 'PY': Publication Year, 'RS': Relevance, 'SO': Source, 'TC': Times Cited, 'VL': Volume)

**Sort** Must be A (ascending) or D (descending). The sort parameter can only be D for Relevance and TimesCited.

**relatedRecords** (*uid, count=100, offset=1, editions=None, timeSpan=None, retrieveParameters=None*)

The relatedRecords operation finds Related Records for the article specified by unique identifier. Related Records share cited references with the specified record. The operation returns the parent record along with the Related Records. The total number of Related Records for the parent record is shown at the end of the response. Use the retrieve parameter count to limit the number of Related Records returned.

**Uid** A unique item identifier. It cannot be None or empty string.

**Count** Number of records to display in the result. Cannot be less than 0 and cannot be greater than 100. If count is 0 then only the summary information will be returned.

**Offset** First record in results to return. Must be greater than zero

**Editions** List of editions to be searched. If None, user permissions will be substituted.

Fields: collection - Name of the collection edition - Name of the edition

**TimeSpan** This element defines specifies a range of publication dates. If timeSpan is null, then the maximum time span will be inferred from the editions data.

Fields: begin - Beginning date for this search. Format: YYYY-MM-DD end - Ending date for this search. Format: YYYY-MM-DD

**RetrieveParameters** Retrieve parameters. If omitted the result of make\_retrieveParameters(offset, count, 'RS', 'D') is used.

**retrieve** (*queryId, count=100, offset=1, retrieveParameters=None*)

The retrieve operation submits a query returned by a previous search, citingArticles, relatedRecords, or retrieveById operation. However, different retrieve parameters may be used to modify the output. For example, if a search operation returns five records sorted by times cited, a subsequent retrieve operation could run the same search against the same database and edition but return 10 records sorted by relevance.

This operation is also useful for overcoming the retrieval limit of 100 records per query. For example, a search operation may find 220 records, as revealed by the content of the recordsFound element, but it returns only records 1-100. A subsequent retrieve operation could return records 101-200 and a third retrieve operation the remaining 20.

**QueryId** The query ID from a previous search

**Count** Number of records to display in the result. Cannot be less than 0 and cannot be greater than 100. If count is 0 then only the summary information will be returned.

**Offset** First record in results to return. Must be greater than zero

**RetrieveParameters** Retrieve parameters. If omitted the result of make\_retrieveParameters(offset, count, 'RS', 'D') is used.

**retrieveById** (*uid, count=100, offset=1, retrieveParameters=None*)

The retrieveById operation returns records identified by unique identifiers. The identifiers are specific to each database.

**Uid** Web of Science unique record identifier

**Count** Number of records to display in the result. Cannot be less than 0 and cannot be greater than 100. If count is 0 then only the summary information will be returned.

**Offset** First record in results to return. Must be greater than zero

**RetrieveParameters** Retrieve parameters. If omitted the result of `make_retrieveParameters(offset, count, 'RS', 'D')` is used.

**search** (*query*, *count=5*, *offset=1*, *editions=None*, *symbolicTimeSpan=None*, *timeSpan=None*, *retrieveParameters=None*)

The search operation submits a search query to the specified database edition and retrieves data. This operation returns a query ID that can be used in subsequent operations to retrieve more records.

**Query** User query for requesting data. The query parser will return errors for invalid queries

**Count** Number of records to display in the result. Cannot be less than 0 and cannot be greater than 100. If count is 0 then only the summary information will be returned.

**Offset** First record in results to return. Must be greater than zero

**Editions** List of editions to be searched. If None, user permissions will be substituted.

Fields: collection - Name of the collection edition - Name of the edition

**SymbolicTimeSpan** This element defines a range of load dates. The load date is the date when a record was added to a database. If `symbolicTimeSpan` is specified, the `timeSpan` parameter must be omitted. If `timeSpan` and `symbolicTimeSpan` are both omitted, then the maximum publication date time span will be inferred from the editions data.

Valid values: '1week' - Specifies to use the end date as today and the begin date as 1 week prior to today. '2week' - Specifies to use the end date as today and the begin date as 2 week prior to today. '4week' - Specifies to use the end date as today and the begin date as 4 week prior to today.

**TimeSpan** This element defines specifies a range of publication dates. If `timeSpan` is used, the `symbolicTimeSpan` parameter must be omitted. If `timeSpan` and `symbolicTimeSpan` are both omitted, then the maximum time span will be inferred from the editions data.

Fields: begin - Beginning date for this search. Format: YYYY-MM-DD end - Ending date for this search. Format: YYYY-MM-DD

**RetrieveParameters** Retrieve parameters. If omitted the result of `make_retrieveParameters(offset, count, 'RS', 'D')` is used.

## 9.2 wos.utils

Here is the documentation for the methods in the *wos.utils* package.

---

`wos.utils.doi_to_wos(wosclient, doi)`

Convert DOI to WOS identifier.

`wos.utils.query(wosclient, wos_query, xml_query=None, count=5, offset=1, limit=100)`

Query Web of Science and XML query results with multiple requests.

`wos.utils.single(wosclient, wos_query, xml_query=None, count=5, offset=1)`

Perform a single Web of Science query and then XML query the results.

### W

`wos.client`, [19](#)  
`wos.utils`, [22](#)



### C

`citedReferences()` (*wos.client.WosClient method*),  
19  
`citedReferencesRetrieve()`  
(*wos.client.WosClient method*), 19  
`citingArticles()` (*wos.client.WosClient method*),  
20  
`close()` (*wos.client.WosClient method*), 20  
`connect()` (*wos.client.WosClient method*), 20

### D

`doi_to_wos()` (*in module wos.utils*), 22

### I

`is_lite()` (*wos.client.WosClient method*), 20

### M

`make_retrieveParameters()`  
(*wos.client.WosClient static method*), 20

### Q

`query()` (*in module wos.utils*), 22

### R

`relatedRecords()` (*wos.client.WosClient method*),  
21  
`retrieve()` (*wos.client.WosClient method*), 21  
`retrieveById()` (*wos.client.WosClient method*), 21

### S

`search()` (*wos.client.WosClient method*), 22  
`single()` (*in module wos.utils*), 22

### W

`wos.client` (*module*), 19  
`wos.utils` (*module*), 22  
`WosClient` (*class in wos.client*), 19