

Developer Guide. Cbonds API

| 23 August, 2023

[Definitions](#)

[Getting data from a web service](#)

[User authorization](#)

[Getting the service scheme](#)

[Data request](#)

[Key response parameters](#)

[Data request using filters](#)

[Examples](#)

[Limits and errors](#)

[Request volume](#)

[Data update frequency](#)

[Queries frequency](#)

[Restrictions on the number of records](#)

[List of possible errors in the web service response](#)

[Demo version of the web service](#)

[Example of accessing a web service in Python 3.x](#)

[List of endpoints](#)

[Bonds](#)

[Issuers](#)

[Indices](#)

[Stocks](#)

[ETF and Mutual Funds](#)

Definitions

Web service is a software system identified by a URL string. A description of this software system can be found by other software systems that can interact with it according to this description through messages transmitted via Internet protocols. Cbonds API web service interfaces are defined in XML and JSON.

WSDL (Web Services Description Language) is a language for describing web services based on XML.

SOAP (Simple Object Access Protocol) is an XML-based messaging protocol.

XML is an extensible markup language for storing and transmitting structured data.

JSON (JavaScript Object Notation) is a text-based data exchange format based on JavaScript.

Web service operation/method/endpoint (hereinafter referred to as endpoint) is a web service function that returns data to the client in WSDL+SOAP or JSON format.

User is a client of the Cbonds API service, having a login and password for authorization and accessing a specific dataset.

Getting data from a web service

User authorization

In order to prevent unauthorized access to data for all endpoints described below, you must specify the username/password assigned to the service for the user.

To get acquainted with the technical aspect of the web service, you can use access of a preconfigured user (i.e. test user):

Login: Test

Password: Test

The test user has access to a limited dataset, which does not display the completeness of the database.

Getting the service scheme

An XSD/XML schema or JSON schema can be obtained by a GET request with a user's username and password specified in the URL:

Schema	Request URL
XSD/XML	https://ws.cbonds.info/services/wsd/?login=LOGIN&password=PASSWORD
JSON	https://ws.cbonds.info/services/json?login=LOGIN&password=PASSWORD

You can also get the schema using the POST request by specifying the username and password in the body of the POST request (not in the URL).

Two parameters in the schema that indicate the possibility of sorting and filtering:

sortable - indicates the possibility of sorting by field (1 - sorting is possible, 0 - not possible);

searchable – indicates the possibility of filtering by field (1 - filtration is possible, 0 - not possible).

Data request

Interaction with a web service for receiving data is carried out by sending POST requests of the HTTPS protocol. Depending on your preferred data, you should send queries to the following URLs:

Data format	Request URL
XSD/XML	https://ws.cbonds.info/services/wsd/(endpoint name)*/?lang=eng
JSON	https://ws.cbonds.info/services/json/(endpoint name)*/?lang=eng

* for example: https://ws.cbonds.info/services/json/get_emissions

For the operation list, see Endpoints reference

The request body contains the request parameters. Request body structure in JSON format:

```

{
  "auth":{"login":"LOGIN","password":"PASSWORD"},
  "filters":[{"field":"...", "operator":"...", "value":...}, {}],
  "quantity":{"limit":10,"offset":0},
  "sorting":[{"field":"...", "order":"..."}],
  "fields":[{"field": "..."}, {"field": "..."}]
}

```

Where the main request parameters are:

“**auth**” - contains the user's login and password;

“**filters**” - contains the specified selection conditions;

“**quantity**” - contains parameters {"limit":..., "offset":...}

 “**limit**” - number of records per page for a single request;

 “**offset**” - page number. It must be a multiple of the limit, e.g. with a limit of 1000 offset can be 0, 1000, 2000, etc.

“**sorting**” - contains a list of fields for sorting;

“**fields**” - contains a list of fields to display in response.

Request body structure in WSDL+SOAP format:

```

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Body
    <m:RequestMessage_emissions xmlns:m="https://ws.cbonds.info/">

      <m:AuthData>
        <m:Login><![CDATA[LOGIN]]></m:Login>
        <m>Password><![CDATA[PASSWORD]]></m>Password>
      </m:AuthData>

      <m:Filters>
        <m:NumericFilter>
          <m:FilterField>...</m:FilterField>
          <m:FilterOperator>...</m:FilterOperator>
          <m:FilterValue><![CDATA[...]]></m:FilterValue>
        </m:NumericFilter>
      </m:Filters>

      <m:QuantityData>
        <m:Limit>10</m:Limit>
        <m:Offset>0</m:Offset>
      </m:QuantityData>

      <m:SortRules>
        <m:SortRule>
          <m:SortField></m:SortField>
          <m:SortOrder>...</m:SortOrder>
        </m:SortRule>
      </m:SortRules>

      <m:ResponseItemFields>

```

```

    <m:ResponseItemField>
      <m:ItemField>...</m:ItemField>
    </m:ResponseItemField>
  </m:ResponseItemFields>

</m:RequestMessage_emissions>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Key response parameters

Response structure in JSON format:

```

{
  "count":4,
  "total":4,
  "limit":10,
  "offset":0,
  "exec_time":0.4443,
  "items":[...],
  "meta":{"cms_full_gen_time":0.4444, "user_id":33 }
}

```

Where response parameters are:

"count" - the number of records found for the current page (for the last page, it may not be equal to limit);

"total" - the number of records found. To build pagination (a sequential request plan for selecting all the data in parts), we focus on the given field in the response (divide it by the limit and get the last page number in the whole part, if we count from zero). For example, if the limit is 1000 and the value total=2500, you will need to perform three queries with offsets $0*1000=0$, $1*1000=1000$, $2*1000=2000$;

"limit" - takes the same value as at the request;

"offset" - the offset value that is not a multiple of the limit is reduced to a multiple to the lower side;

"exec_time" - this parameter is required for optimization. If there are problems with the performance of the service (and there are opportunities), you can organize logging of this parameter for each operation, in the ideal case - statistics with aggregated values for the operation, filters, offset, etc.

"items" - sends all records for the corresponding page;

"cms_full_gen_time"- is used for optimization;

"user_id" – your user identifier in our system, for the test user user_id = 33.

Response structure in WSDL+SOAP format:

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope

```

```

xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns1="https://ws.cbonds.info/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <ns1:ResponseMessage_emitents>
      <ns1:Count>4</ns1:Count>
      <ns1:Total>4</ns1:Total>
      <ns1:Limit>10</ns1:Limit>
      <ns1:Offset>0</ns1:Offset>
      <ns1:ExecTime>0.4139</ns1:ExecTime>
      <ns1:Items><ns1:Item>
        ...
      </ns1:Item></ns1:Items>
    </ns1:ResponseMessage_emitents>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

<!-- user_id: 33 -->
<!-- cms_full_gen_time: 0.4145 -->
<!-- lang: eng -->
<!-- strict_mode: 1 -->
<!-- performLogging: -1 -->

```

Data request using filters

When requesting data, you can create filters selecting only certain information that meets the specified criteria from all the data. **The list of fields for which a filter can be created is limited.**

List of possible operators for data filtering:

Operator	Decoding the operator	Operator value
'eq'	'equal'	Equal
'ne'	'not equal'	Not equal
'lt'	'less'	Strictly less
'le'	'less or equal'	Less or equal
'gt'	'greater'	Strictly greater
'ge'	'greater or equal'	Greater or equal
'in'	'is in'	Included in the busload
'ni'	'is not in'	Not included in the busload
'bw'	'begins with'	Begins with (characters)
'bn'	'does not begin with'	Does not begin with (characters)
'ew'	'ends with'	Ends with (characters)
'en'	'does not end with'	Does not end with (characters)
'cn'	'contains'	Contains part of the string
'nc'	'does not contain'	Does not contain part of the string
'nu'	'is null',	Has a NULL value
'nn'	'is not null'	Not NULL

Examples

Examples of requests that can be applied to the "get_emissions" endpoint (issues static data).

The request will be sent to the address https://ws.cbonds.info/services/json/get_emissions/?lang=eng

1. Select an issue with specific ISIN - US037833EL06

Request in JSON format:

```
{
  "auth": {"login": "LOGIN", "password": "PASSWORD"},
  "filters": [{"field": "isin_code", "operator": "eq", "value": "US037833EL06"}],
  "quantity": {"limit": 10, "offset": 0},
  "sorting": [{"field": "id", "order": "asc"}]
  "fields": [{"field": "id"}, {"field": "maturity_date"}]
}
```

Request in WSDL+SOAP format:

```
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Body>
    <m:RequestMessage_emissions xmlns:m="https://ws.cbonds.info/">
      <m:AuthData>
        <m:Login>LOGIN</m:Login>
        <m>Password>PASSWORD</m>Password>
      </m:AuthData>
      <m:Filters>
        <m:NumericFilter>
          <m:FilterField>isin_code</m:FilterField>
          <m:FilterOperator>eq</m:FilterOperator>
          <m:FilterValue><![CDATA[US037833EL06]]></m:FilterValue>
        </m:NumericFilter>
      </m:Filters>
      <m:QuantityData>
        <m:Limit>10</m:Limit>
        <m:Offset>0</m:Offset>
      </m:QuantityData>
      <m:SortRules>
        <m:SortRule>
          <m:SortField>id</m:SortField>
          <m:SortOrder>asc</m:SortOrder>
        </m:SortRule>
      </m:SortRules>
      <m:ResponseItemFields>
        <m:ResponseItemField>
          <m:ItemField>id</m:ItemField>
        </m:ResponseItemField>
        <m:ResponseItemField>
          <m:ItemField>maturity_date</m:ItemField>
        </m:ResponseItemField>
      </m:ResponseItemFields>
    </m:RequestMessage_emissions>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Further, for brevity, all examples will be given only in JSON format.

2. Select multiple issues with specific ISINs, e.g. US037833EL06 and USU5876JAM72.

Request in JSON format:

```
{
  "auth":{"login":"LOGIN","password":"PASSWORD"},
  "filters":[{"field":"isin_code","operator":"in","value":"US037833EL06;USU5876JAM72"}],
  "quantity":{"limit":10,"offset":0},
  "sorting":[{"field":"","order":"asc"}],
  "fields":[{"field":"id"}, {"fields":"maturity_date"}]
}
```

3. Select issues information on which were updated after August 21, 2023

Request in JSON format:

```
{
  "auth":{"login":"LOGIN","password":"PASSWORD"},
  "filters":[{"field":"updating_date","operator":"ge","value":"2023-08-21"}],
  "quantity":{"limit":10,"offset":0},
  "sorting":[{"field":"","order":"asc"}]
  "fields":[]
}
```

4. Select issues information on which were updated between August 1 and August 21, 2023

Request in JSON format:

```
{
  "auth":{"login":"LOGIN","password":"PASSWORD"},
  "filters":[{"field":"updating_date","operator":"le","value":"2023-08-01"},
    {"field":"updating_date","operator":"ge","value":"2023-08-21"}],
  "quantity":{"limit":10,"offset":0},
  "sorting":[{"field":"","order":"asc"}]
  "fields":[]
}
```

Limits and errors

Request volume

Maximum number of records in the response (items in the items field) 1000

To get all the data, you must run multiple queries by changing the **offset**. Each response contains information about the total number of records (the “total” field) that are available with the filters transferred in the request.

Data update frequency

Data in all endpoints, except for the *get_tradings_stocks_full_realtime_new* and *get_tradings_realtime* endpoints (intraday quotes), is updated 1 time per hour.

Queries frequency

Period	Limit
Allowed number of requests per endpoint per minute	No more than 30 times
Allowed number of requests per endpoint per day	No more than 10,000 times

Restrictions on the number of records

To ensure stable and uninterrupted work of the web service, a limit is introduced on the number of records in the result set.

The default maximum number of records is 1,000,000. If this value is exceeded, an error message with status 500100 will be broadcast.

List of possible errors in the web service response

- Errors that you can get when accessing a web service:
 - HTTP/1.1 301 Moved Permanently** — access via http instead of https;
 - HTTP/1.1 403 Forbidden** — incorrect login; invalid password; no login, no password, no authorization block; the limit of records in the response has been exceeded (more details in paragraph 4. Number of records in the result set exceeded);
 - HTTP/1.1 500 Internal Service Error** — config with syntax error; a request for an endpoint to which there is no access;
 - HTTP/1.1 504** – response waiting time out.
- Software errors:
 - https protocol required** — access via http instead of https;
 - Max requests per minute limit max_per_minute exceeded. Try to request the next minute.** — limit on the number of requests per minute;
 - Max requests per day limit max_per_day exceeded. Try to request next day.** — limit on the number of requests per day.
- Messages about intercepted errors:
 - Undefined index: xxxx
 - Invalid field (xxxx) for filtering

- c. The request JSON string is not valid. The collection "%services_XXXXX%" is not available for user id "y"
4. Number of records in the result set exceeded, message text:
"You exceeded the maximum number of records in the result set total size. Please use filtering in your request or refine existing filters to reduce the volume of results. The number of records in the result set is n, the maximum possible number is 1000000."

Demo version of the web service

The demo version serves as an assistant for developing client applications. The tool contains the same functionality as the full version and differs only in the presence of a visual interface.

To start working with the demo version of the web service, click one of the links:

1. <https://ws.cbonds.info/services/json/demo/>
2. <https://ws.cbonds.info/services/wSDL/demo/>

Below is a description of working only with the demo version for JSON, working with the demo version for WSDL is similar.

1. Enter username and password for access (or test user credentials) and click the "Request service schema"

The screenshot displays a web interface titled "Cbonds.Database WebServices >> Json Demo". It features a "Credentials" section with a "Login" field containing the text "Test" and a "Password" field with masked characters "****". Below the password field is a link "Request service schema" followed by the text "with <document> tags localized for:" and a dropdown menu currently set to "eng". Underneath the credentials section are two large, empty rectangular panels labeled "Request" and "Response".

2. After requesting the service schema, the "Request" and "Response" windows will be filled in.

The "Response" window contains the service schema, in this example, a JSON schema.



3. Select one of the available endpoints from the "Available operation" drop-down list. For example, let's select the "get_emissions" endpoint.



4. After selecting the endpoint, the following request parameter settings will be available:

- select the number of records to send;
- sort by a field in received responses;
- filters on the data to be sent.

These options can be applied to all endpoints except reference endpoints. To work with filters, select a field, click the "append" button, select an operator and a value.

Cbonds.Database WebServices » Json Demo

Credentials
Login Password
[Request service schema](#) with <document> tags localized for:

Table operations: Dictionary: No

Records Limit (page size) Records Offset (first record position)

Sorting:
Field
Order

Response fields:
Field
id issue
formal issuer (id)
Country of the formal issuer(id)
Branch of the formal issuer(id)
Registration form of the formal issuer (id)
Type formal issuer
formal issuer (eng)

Restrictions:
Append field for restriction

Perform operation
Request
/ services/ json/
login=Test&password=test&lang=eng

Response
success
{
 "service": {"get auction types":
 {"uri": "https://ws.cbonds.info/services/
 \\/json/get_auction_types\/", "arguments":
 {"auth":
 {"login": "", "password": ""}, "filters":
 [{"field": "id", "type": "int", "available_ope
 rators":
 ["eq", "ne", "nu", "nn", "lt", "le", "gt", "ge", "
 in", "ni"], "title": "ID", "sortable": 0, "search
 able": 0},
 {"field": "name_eng", "type": "string", "avail
 able_operators":
 ["eq", "ne", "nu", "nn", "bw", "bn", "sw", "en", "
 cn", "nc", "in", "ni"], "title": "Auction type
 /name", "searchable": 0, "searchable": 0} }
 }
 }
 }
 }
 }
 }
 }
}

5. After the request is completed, the body of the POST request is automatically generated in the "Request" window. The "Response" window will display the response in the appropriate format.

```
Request
https:// ws.cbonds.info/ services/ json/
get_emissions/ ?lang=eng

{"auth":
{"login":"Test","password":"test"},"filters":
[{"field":"id","operator":"eq","value":"US037
833EL06"}],"quantity":
{"limit":10,"offset":0},"sorting":
[{"field":"","order":"asc"},"fields":[]}
```

Thus, when developing a product, you can consult the data returned by the web service using demo version.

Example of accessing a web service in Python 3.x

Below there is an example of Python 3.x code performing:

1. Data request with filtering;
2. Writing data to pandas DataFrame;
3. Saving a table to csv.

```
import requests
import pandas as pd

url_emissions = 'https://ws.cbonds.info/services/json/get_emissions/?lang=eng'

json_data_emissions = {"auth":{"LOGIN":"Test","password":"PASSWORD"},
"filters":[{"field":"isin_code","operator":"in","value":"US037833EL06;USU5876JAM72"}],
"quantity":{"limit":10,"offset":0},"sorting":[{"field":"","order":"asc"}]}

response_emissions = requests.get(url_emissions, json=json_data_emissions).json()
print(response_emissions)

initial_dict = response_emissions['items']

df = pd.DataFrame.from_dict(initial_dict)
df

df.to_csv(index=False)
```

List of endpoints

For each web service endpoint, the list of available, filtered, and sorted fields is limited. All available endpoint fields are divided into basic and additional fields. The basic operation fields are available to all users when connecting to the web service. If necessary, a number of additional fields can be added to the list of available fields for web service endpoints. The list of fields that can be searched or sorted can also be expanded if necessary.

List of all endpoints available via Cbonds web-site - <https://cbonds.com/api/catalog/folders/#catalog>

Bonds

Issues static data - `get_emissions`

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_emissions/

https://ws.cbonds.info/services/wsd/get_emissions/?lang=eng

https://ws.cbonds.info/services/json/get_emissions/?lang=eng

Issues default data - `get_emission_default`

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_emission_default/

https://ws.cbonds.info/services/wsd/get_emission_default/?lang=eng

https://ws.cbonds.info/services/json/get_emission_default/?lang=eng

Issues listing data - `get_emission_quote_listings`

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_emission_quote_listings/

https://ws.cbonds.info/services/wsd/get_emission_quote_listings/?lang=eng

https://ws.cbonds.info/services/json/get_emission_quote_listings/?lang=eng

Tap issues data - `get_emission_tap_issues`

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_emission_tap_issues/

https://ws.cbonds.info/services/wsd/get_emission_tap_issues/?lang=eng

https://ws.cbonds.info/services/json/get_emission_tap_issues/?lang=eng

Coupon payment schedule - `get_flow_new`

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_flow_new/

https://ws.cbonds.info/services/wsd/get_flow_new/?lang=eng

https://ws.cbonds.info/services/json/get_flow_new/?lang=eng

Guarantors data - `get_emission_guarantors`

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_emission_guarantors/

https://ws.cbonds.info/services/wsd/get_emission_guarantors/?lang=eng

https://ws.cbonds.info/services/json/get_emission_guarantors/?lang=eng

Data on out/call options - get_offert

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_offert/

https://ws.cbonds.info/services/wsd/get_offert/?lang=eng

https://ws.cbonds.info/services/json/get_offert/?lang=eng

Participation of investment banks in the placement - get_participation

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_participation/

https://ws.cbonds.info/services/wsd/get_participation/?lang=eng

https://ws.cbonds.info/services/json/get_participation/?lang=eng

Bonds quotes from stock exchanges - get_tradings_new (40 days snapshot)

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_tradings_new/

https://ws.cbonds.info/services/wsd/get_tradings_new/?lang=eng

https://ws.cbonds.info/services/json/get_tradings_new/?lang=eng

Intraday quotes on bonds - get_tradings_realtime

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_tradings_realtime/

https://ws.cbonds.info/services/wsd/get_tradings_realtime/?lang=eng

https://ws.cbonds.info/services/json/get_tradings_realtime/?lang=eng

OTC quotes on bonds - get_mpquotes

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_mpquotes/

https://ws.cbonds.info/services/wsd/get_mpquotes/?lang=eng

https://ws.cbonds.info/services/json/get_mpquotes/?lang=eng

ACI data as of the date (without quotes) - get_aci

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_aci/

https://ws.cbonds.info/services/wsd/get_aci/?lang=eng

https://ws.cbonds.info/services/json/get_aci/?lang=eng

Data on issues covenants - get_covenants_emissions

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_covenants_emissions/

https://ws.cbonds.info/services/wsd/get_covenants_emissions/?lang=eng

https://ws.cbonds.info/services/json/get_covenants_emissions/?lang=eng

Government bonds auctions - get_auctions

Fields description - https://cbonds.com/api/catalog/API_Emissions/get_auctions/

https://ws.cbonds.info/services/wsd/get_auctions/?lang=eng

https://ws.cbonds.info/services/json/get_auctions/?lang=eng

Information on basic assets for structured products - get_emission_underlying_assets

Fields description -

https://cbonds.com/api/catalog/API_Emissions/get_emission_underlying_assets/

https://ws.cbonds.info/services/wsd/get_emission_underlying_assets/?lang=eng

https://ws.cbonds.info/services/json/get_emission_underlying_assets/?lang=eng

Use of proceeds for bonds - get_use_of_proceeds

Fields description - https://cbonds.com/api/catalog/dictionaries/get_use_of_proceeds/

https://ws.cbonds.info/services/wsd/get_use_of_proceeds/?lang=eng

https://ws.cbonds.info/services/json/get_use_of_proceeds/?lang=eng

Issuers

Issuers static data - get_emitents

Fields description - https://cbonds.com/api/catalog/API_FinancialReports/get_emitents/

https://ws.cbonds.info/services/wsd/get_emitents/?lang=eng

https://ws.cbonds.info/services/json/get_emitents/?lang=eng

IFRS indicators (financial sector, provided by Cbonds) - get_report_msfo_finance

Fields description - https://cbonds.com/api/catalog/API_FinancialReports/get_report_msfo_finance/

https://ws.cbonds.info/services/wsd/get_report_msfo_finance/?lang=eng

https://ws.cbonds.info/services/json/get_report_msfo_finance/?lang=eng

IFRS indicators (real sector, provided by Cbonds) - get_report_msfo_real

Fields description - https://cbonds.com/api/catalog/API_FinancialReports/get_report_msfo_real/

https://ws.cbonds.info/services/wsd/get_report_msfo_real/?lang=eng

https://ws.cbonds.info/services/json/get_report_msfo_real/?lang=eng

Financial statements globally (annual) - get_report_msfo_global_annual

Fields description -

https://cbonds.com/api/catalog/API_FinancialReports/get_report_msfo_global_annual/

https://ws.cbonds.info/services/wSDL/get_report_msfo_global_annual/?lang=eng

https://ws.cbonds.info/services/json/get_report_msfo_global_annual/?lang=eng

Financial statements globally (quarter) - get_report_msfo_global_quarter

Fields description -

https://cbonds.com/api/catalog/API_FinancialReports/get_report_msfo_global_quarter/

https://ws.cbonds.info/services/wSDL/get_report_msfo_global_quarter/?lang=eng

https://ws.cbonds.info/services/json/get_report_msfo_global_quarter/?lang=eng

Indices

Index portfolio structure - get_index_content

Fields description - https://cbonds.com/api/catalog/API_Indices/get_index_content/

https://ws.cbonds.info/services/wSDL/get_index_content/?lang=eng

https://ws.cbonds.info/services/json/get_index_content/?lang=eng

Index values - get_index_value_new

Fields description - https://cbonds.com/api/catalog/API_Indices/get_index_value_new/

https://ws.cbonds.info/services/wSDL/get_index_value_new/?lang=eng

https://ws.cbonds.info/services/json/get_index_value_new/?lang=eng

Stocks

Stock parameters - get_stocks_full

Fields description - https://cbonds.com/api/catalog/API_Stocks/get_stocks_full/

https://ws.cbonds.info/services/wSDL/get_stocks_full/?lang=eng

https://ws.cbonds.info/services/json/get_stocks_full/?lang=eng

Stocks dividends - get_stocks_dividends_v2

Fields description - https://cbonds.com/api/catalog/API_Stocks/get_stocks_dividends_v2/

https://ws.cbonds.info/services/wSDL/get_stocks_dividends_v2/?lang=eng

https://ws.cbonds.info/services/json/get_stocks_dividends_v2/?lang=eng

Stocks quotes from stock exchanges - get_tradings_stocks_full_new (40 days snapshot)

Fields description - https://cbonds.com/api/catalog/API_Stocks/get_tradings_stocks_full_new/

https://ws.cbonds.info/services/wsdll/get_tradings_stocks_full_new/?lang=eng

https://ws.cbonds.info/services/json/get_tradings_stocks_full_new/?lang=eng

Stocks listing data - get_stocks_trading_grounds

Fields description - https://cbonds.com/api/catalog/API_Stocks/get_stocks_trading_grounds/

https://ws.cbonds.info/services/wsdll/get_stocks_trading_grounds/?lang=eng

https://ws.cbonds.info/services/json/get_stocks_trading_grounds/?lang=eng

Capitalization of stocks - get_stocks_capitalization

Fields description - https://cbonds.com/api/catalog/API_Stocks/get_stocks_capitalization/

https://ws.cbonds.info/services/wsdll/get_stocks_capitalization/?lang=eng

https://ws.cbonds.info/services/json/get_stocks_capitalization/?lang=eng

ETF and Mutual Funds

ETF share classes and Mutual Funds quotes from stock exchanges -

get_etf_share_classes_quotes

Fields description - https://cbonds.com/api/catalog/API_Funds/get_etf_share_classes_quotes/

https://ws.cbonds.info/services/wsdll/get_etf_share_classes_quotes/?lang=eng

https://ws.cbonds.info/services/json/get_etf_share_classes_quotes/?lang=eng

ETF funds and Mutual Funds static data - get_etf_funds

Fields description - https://cbonds.com/api/catalog/API_Funds/get_etf_funds/

https://ws.cbonds.info/services/wsdll/get_etf_funds/?lang=eng

https://ws.cbonds.info/services/json/get_etf_funds/?lang=eng

ETF share classes and Mutual Funds listing data - get_etf_share_classes_trading_grounds

Fields description -

https://cbonds.com/api/catalog/dictionaries/get_etf_share_classes_trading_grounds/

https://ws.cbonds.info/services/wsdll/get_etf_share_classes_trading_grounds/?lang=eng

https://ws.cbonds.info/services/json/get_etf_share_classes_trading_grounds/?lang=eng

ETF share classes NAV data - get_etf_share_classes_nav

Fields description - https://cbonds.com/api/catalog/dictionaries/get_etf_share_classes_nav/

https://ws.cbonds.info/services/wsdll/get_etf_share_classes_nav/?lang=eng

https://ws.cbonds.info/services/json/get_etf_share_classes_nav/?lang=eng