

JODI REST API Specification v1.2

Last updated on 07 October 2026



Contents

Introduction.....	3
Authentication	4
Obtaining a Token	4
Response Status Codes	4
Response Payloads.....	5
Summary of Endpoints.....	6
Data Queries.....	6
Modified Data	6
Navigating Datasets	6
Calls to Endpoints.....	7
Data Query Calls	7
Modified Data Calls	8
Responses	10
Response Status Codes	10
Response Payloads.....	10
Appendix A – Code Examples	13
Obtaining a token.....	13
Querying data	15
Appendix B – Field Codes	16
Countries	16
Flows.....	21
Products.....	22
Units	22
Appendix C – Rate Limits	24

Introduction

This document describes the JODI REST API which can be used to programmatically access the data held in the JODI World database.

The REST API is read only, it does not allow modification of the data.

The API uses a series of methods that are callable over https and utilises query string parameters to query the database for data. In order to perform a request you are required to send a HTTP GET request to a URL (the endpoint) passing any query parameters you wish to specify to filter the data.

Whenever a call is made to an endpoint, a response will be returned in JSON format. The endpoints and instructions for querying the data are outlined in this document.

All calls to data endpoints are required to be authenticated, so in order to use the JODI REST API you will need a user account. User accounts can be created by signing up as a member of the JODI website (<https://www.jodidata.org/sign-up.aspx>).

All calls to the API are subject to rate limits to ensure fair usage. See Appendix C for more information.

Authentication

All calls to the JODI REST API are authenticated via JSON Web Tokens (JWT). Before accessing a data endpoint you will first need to obtain a JWT and include this in your call to the data endpoint.

Once you have obtained a token you need to store it and include it as a Bearer Token when sending requests to data endpoints. Tokens are valid for one hour, after which you will need to request a new one. We therefore recommend that you also store the date and time that your token expires.

Obtaining a Token

To obtain a JWT, send a POST request including your username and password.

Endpoint URL	https://www.jodidata.org/api/token
--------------	---

The table below provides a list of the parameters to be included in the POST body.

Parameter name	Data type	Description	Required
username	String	Your username is the email address you used to sign up on the JODI website.	Yes
password	String	The password that you created when signing up to the JODI website	Yes

Response Status Codes

The response type will be signified by the HTTP Status Code. The following codes are applicable:

200 – OK

The request was successful.

400 – Bad Request

The request was not valid or could not be served. The most likely reason will be missing parameters.

401 – Unauthorised

User authentication failed.

405 – Method Not Allowed

The request method was not allowed. This will usually be caused by sending a GET request instead of a POST request.

Response Payloads

Success Responses

If authentication is successful, the HTTP response body will contain the returned data in JSON format. The JSON will contain three top level keys - username, token and expires.

```
{
  "username": "user@example.com",
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
  "expires": "2026-10-07T15:18:32.3078577+01:00"
}
```

Key	Data type	Description
username	String	The username that was included in your call
token	String	The JSON Web Token
expires	DateTime	The date and time that the token expires

Summary of Endpoints

The JODI REST API has two endpoints for oil data and two endpoints for gas data, one to query the data and one to retrieve any data modified since your last request

Data Queries

Use the data URLs to query the full database for the data you require. Pass any query parameters you wish in your call to the endpoint and any matching data will be returned.

Modified Data

Use the modified data URLs to retrieve any data that has been added or updated since your last request. You should store the date of your last request and supply this in your call. The endpoint will return any data that has been added or updated since that date.

Navigating Datasets

Any data matching a request sent to an endpoint will be returned as paged datasets. You can specify the number of records you would like to receive in your results from a minimum of 100 up to a maximum of 100,000.

The data returned to you will contain metadata that will help you to navigate the pages of data. The metadata will include 2 keys to signify whether there are further pages of data – hasNext and hasPrevious. A value of true indicates if there are pages of results before and/or after the current page and matching next and/or previous keys will provide you with the url to call to retrieve the next or previous page of results.

You can use the next and previous meta parameters to easily navigate your results without having to manually build a query for each page.

Calls to Endpoints

There are two types of endpoint calls – a data query call (OilData and GasData) or a modified data call (OilModified and GasModified).

Data Query Calls

A data query call is used to query the database for oil or gas data. To make a call, you can send a HTTP GET request to the following endpoint URLs.

Endpoint URL – oil	https://www.jodidata.org/api/oildata
Endpoint URL – gas	https://www.jodidata.org/api/gasdata

The table below provides a list of the query string parameters that can be included in the URL to filter the data.

Parameter name	Field name	Data type	Description	Required
dateFrom	Date	String (yyyy-MM)	The start of the date range for your query. This will query the Date field in the database and return data matching this date or later. Default: A year before the dateTo date	No
dateTo	Date	String (yyyy-MM)	The end of the date range for your query. This will query the Date field in the database and return data matching this date or earlier. Default: A year after the dateFrom date if provided, otherwise the month before today's date	No
countries	Country	Comma separated string	A comma separated list of two character country codes to include in your query. If not provided all countries will be included in the returned data.	No
flows	Flow	Comma separated string	A comma separated list of flow codes to include in your query.	No

products			If not provided all flows will be included in the returned data.	
	Product	Comma separated string	A comma separated list of product codes to include in your query. If not provided all products will be included in the returned data.	No
units	Unit	Comma separated string	A comma separated list of unit codes to include in your query. If not provided all units will be included in the returned data.	No
pageSize	-	Integer	The number of rows to include in the returned data. Default: 100 Minimum: 100 Maximum: 100,000	No
pageNumber	-	Integer	The page number to return. When paging you should be careful to always keep the same page size to avoid missing rows. Default: 1	No

Example Call

```
GET https://www.jodidata.org/api/oildata?dateFrom=2026-07&dateTo=2026-08&units=KBD&pageSize=1000
```

Modified Data Calls

A modified data call is used to request data that has been added or updated since your last request for oil or gas data. To make a call, you can send a HTTP GET request to the following endpoint URLs.

Endpoint URL – oil	https://www.jodidata.org/api/oilmodified
Endpoint URL – gas	https://www.jodidata.org/api/gasmodified

The table below provides a list of the query string parameters that can be included in the URL to filter the data.

Parameter name	Field name	Data type	Description	Required
date	Date	Date (yyyy-MM-ddTHH:mm:ss)	The date of your last request. The response will include only the data that has been added or updated after this date. Minimum: 1 year prior to today's date	Yes
pageSize	-	Integer	The number of rows to include in the returned data. Default: 100 Minimum: 100 Maximum: 100,000	No
pageNumber	-	Integer	The page number to return. When paging you should be careful to always keep the same page size to avoid missing rows. Default: 1	No

Example Call

```
GET https://www.jodidata.org/api/oilmodified?date=2026-07-01T12:30:00
```

Responses

There are two types of response which you could receive in response to a call to an endpoint:

1. Success – the call was successful
2. Error – the call failed due to an error

Response Status Codes

The response type will be signified by the HTTP Status Code. The following codes are applicable:

200 – OK

The request was successful.

400 – Bad Request

The request was not valid or could not be served. The response payload will contain further details.

401 – Unauthorised

User authentication failed.

404 – Not Found

An invalid endpoint was requested.

405 – Method Not Allowed

The request method was not allowed. This will usually be caused by sending a POST request instead of a GET request.

429 – Too Many Requests

You have reached your rate limit. See Appendix C.

Response Payloads

If your call to an endpoint is successful, the HTTP response body will contain the returned data in JSON format. The JSON will contain two top level keys, data and meta.

```
{
  "data": [ ... ],
  "meta": { ... }
}
```

Data Key Values

The data key will contain an array of data points in the following format:

```

"data": [
  {
    "country": "AM",
    "product": "GASDIES",
    "flow": "TOTIMPSB",
    "unit": "KBD",
    "date": "2026-07",
    "quantity": 3.1032,
    "quality": 3
  },
  ...
]

```

Key	Field name	Data type	Description
country	Country	String	The two character country value of the data point
product	Product	String	The product code value of the data point
flow	Flow	String	The flow code value of the data point
unit	Unit	String	The unit code value of the data point
date	Date	Date String (yyyy-MM)	The date of the data point in the form Year-Month, e.g. 2020-01
quantity	Quantity	Decimal	The quantity value of the data point
quality	Quality	Integer	The quality value of the data point

Meta Key Values

The meta key contains a JSON object in the following format:

```

"meta": {
  "pageNumber": 1,
  "pageSize": 1000,
  "totalRecords": 6879,
  "totalPages": 7,
  "hasPrevious": false,
  "hasNext": true,
  "next":
  "https://www.jodidata.org/api/oildata?pageNumber=2&pageSize=1000&dateFrom=2026-07&dateTo=2026-08&units=KBD",
  "previous": null,
  "filters": {
    "countries": null,
    "dateFrom": "2026-07-01",
    "dateTo": "2026-08-31",

```

```

    "flows": null,
    "products": null,
    "units": "KBD",
    "url": "https://www.jodidata.org/api/oildata",
    "pageNumber": 1,
    "pageSize": 1000,
    "query": "dateFrom=2026-07&dateTo=2026-08&units=KBD"
  }
}

```

Key	Data type	Description
pageNumber	Integer	The page number of the current result set
pageSize	Integer	The number of records in the current result set
totalRecords	Integer	The total number of records matching the query
totalPages	Integer	The total number of pages in the query results
hasNext	Boolean	Is there a next page in the query results
hasPrevious	Boolean	Is there a previous page in the query results
next	String	The URL to call to retrieve the next page of results
previous	String	The URL to call to retrieve the previous page of results
filters	JSON	A JSON object containing the call query

Appendix A – Code Examples

Obtaining a token

cUrl

```
curl -X POST --fail-with-body --silent --show-error \  
  --data-urlencode "username=user@example.com" \  
  --data-urlencode "password=your-password" \  
  "https://www.jodidata.org/api/token"
```

.NET

```
using var httpClient = new HttpClient();  
  
using var content = new FormUrlEncodedContent(  
[  
    new KeyValuePair<string, string>("username", "user@example.com"),  
    new KeyValuePair<string, string>("password", "your-password")  
]);  
  
using var response = await httpClient.PostAsync(  
    "https://www.jodidata.org/api/token",  
    content);  
  
response.EnsureSuccessStatusCode();  
  
using var tokenJson = System.Text.Json.JsonDocument.Parse(  
    await response.Content.ReadAsStringAsync());  
  
string jwt = tokenJson.RootElement  
    .GetProperty("token")  
    .GetString()  
    ?? throw new InvalidOperationException("No token was returned.");
```

Java

```
import java.net.URI;
import java.net.URLEncoder;
import java.net.http.HttpClient;
import java.net.http.HttpRequest;
import java.net.http.HttpResponse;
import java.nio.charset.StandardCharsets;

HttpClient client = HttpClient.newHttpClient();

String form =
    "username=" + URLEncoder.encode(
        "user@example.com",
        StandardCharsets.UTF_8)
    + "&password=" + URLEncoder.encode(
        "your-password",
        StandardCharsets.UTF_8);

HttpRequest request = HttpRequest.newBuilder()
    .uri(URI.create("https://www.jodidata.org/api/token"))
    .header(
        "Content-Type",
        "application/x-www-form-urlencoded")
    .POST(HttpRequest.BodyPublishers.ofString(form))
    .build();

HttpResponse<String> response = client.send(
    request,
    HttpResponse.BodyHandlers.ofString());
```

Querying data

cUrl

```
curl --fail-with-body --silent --show-error \  
  "https://www.jodidata.org/api/oildata?countries=SA&products=CRUDEOIL" \  
  -H "Authorization: Bearer YOUR_TOKEN"
```

.NET

```
using System.Net.Http.Headers;  
  
using var httpClient = new HttpClient();  
  
httpClient.DefaultRequestHeaders.Authorization =  
    new AuthenticationHeaderValue("Bearer", "YOUR_TOKEN");  
  
using var response = await httpClient.GetAsync(  
    "https://www.jodidata.org/api/oildata?countries=SA&products=CRUDEOIL");
```

Java

```
import java.net.URI;  
import java.net.http.HttpClient;  
import java.net.http.HttpRequest;  
import java.net.http.HttpResponse;  
  
HttpClient httpClient = HttpClient.newHttpClient();  
  
HttpRequest request = HttpRequest.newBuilder()  
    .uri(URI.create(  
        "https://www.jodidata.org/api/oildata"  
        + "?countries=SA&products=CRUDEOIL"))  
    .header("Authorization", "Bearer YOUR_TOKEN")  
    .GET()  
    .build();  
  
HttpResponse<String> response = httpClient.send(  
    request,  
    HttpResponse.BodyHandlers.ofString());
```

Appendix B – Field Codes

Countries

Code	Name	Code	Name
AF	Afghanistan	BQ	Bonaire, Sint Eustatius and Saba
AX	Åland Islands	BA	Bosnia And Herzegovina
AL	Albania	BW	Botswana
DZ	Algeria	BV	Bouvet Island
AS	American Samoa	BR	Brazil
AD	Andorra	IO	British Indian Ocean Territory
AO	Angola	BN	Brunei Darussalam
AI	Anguilla	BG	Bulgaria
AQ	Antarctica	BF	Burkina Faso
AG	Antigua And Barbuda	BI	Burundi
AR	Argentina	KH	Cambodia
AM	Armenia	CM	Cameroon
AW	Aruba	CA	Canada
AU	Australia	CV	Cape Verde
AT	Austria	KY	Cayman Islands
AZ	Azerbaijan	CF	Central African Republic
BS	Bahamas	TD	Chad
BH	Bahrain	CL	Chile
BD	Bangladesh	CN	China
BB	Barbados	CX	Christmas Island
BY	Belarus	CC	Cocos (Keeling) Islands
BE	Belgium	CO	Colombia
BZ	Belize	KM	Comoros

BJ	Benin	CG	Congo
BM	Bermuda	CD	Congo, The Democratic Republic Of The
BT	Bhutan	CK	Cook Islands
BO	Bolivia, Plurinational State of	CR	Costa Rica
CI	Côte D'ivoire	GR	Greece
HR	Croatia	GL	Greenland
CW	Curaçao	GD	Grenada
CU	Cuba	GP	Guadeloupe
CY	Cyprus	GU	Guam
CZ	Czech Republic	GT	Guatemala
DK	Denmark	GG	Guernsey
DJ	Djibouti	GN	Guinea
DM	Dominica	GW	Guinea-Bissau
DO	Dominican Republic	GY	Guyana
EC	Ecuador	HT	Haiti
EG	Egypt	HM	Heard Island And Mcdonald Islands
SV	El Salvador	VA	Holy See (Vatican City State)
GQ	Equatorial Guinea	HN	Honduras
ER	Eritrea	HK	Hong Kong
EE	Estonia	HU	Hungary
ET	Ethiopia	IS	Iceland
FK	Falkland Islands (Malvinas)	IN	India
FO	Faroe Islands	ID	Indonesia
FJ	Fiji	IR	Iran, Islamic Republic of
FI	Finland	IQ	Iraq
FR	France	IE	Ireland

GF	French Guiana	IM	Isle Of Man
PF	French Polynesia	IL	Israel
TF	French Southern Territories	IT	Italy
GA	Gabon	JM	Jamaica
GM	Gambia	JP	Japan
GE	Georgia	JE	Jersey
DE	Germany	JO	Jordan
GH	Ghana	KZ	Kazakhstan
GI	Gibraltar	KE	Kenya
KI	Kiribati	MC	Monaco
KP	Korea, Democratic People's Republic Of	MN	Mongolia
KR	Korea, Republic of	ME	Montenegro
KW	Kuwait	MS	Montserrat
KG	Kyrgyzstan	MA	Morocco
LA	Lao People's Democratic Republic	MZ	Mozambique
LV	Latvia	MM	Myanmar
LB	Lebanon	NA	Namibia
LS	Lesotho	NR	Nauru
LR	Liberia	NP	Nepal
LY	Libya	NL	Netherlands
LI	Liechtenstein	NC	New Caledonia
LT	Lithuania	NZ	New Zealand
LU	Luxembourg	NI	Nicaragua
MO	Macao	NE	Niger
MK	North Macedonia	NG	Nigeria
MG	Madagascar	NU	Niue

MW	Malawi	NF	Norfolk Island
MY	Malaysia	MP	Northern Mariana Islands
MV	Maldives	NO	Norway
ML	Mali	OM	Oman
MT	Malta	PK	Pakistan
MH	Marshall Islands	PW	Palau
MQ	Martinique	PS	Palestine, State of
MR	Mauritania	PA	Panama
MU	Mauritius	PG	Papua New Guinea
YT	Mayotte	PY	Paraguay
MX	Mexico	PE	Peru
FM	Micronesia, Federated States Of	PH	Philippines
MD	Moldova, Republic Of	PN	Pitcairn
PL	Poland	GS	South Georgia And The South Sandwich Islands
PT	Portugal	ES	Spain
PR	Puerto Rico	LK	Sri Lanka
QA	Qatar	SD	Sudan
RE	Réunion	SR	Suriname
RO	Romania	SJ	Svalbard And Jan Mayen
RU	Russian Federation	SZ	Eswatini
RW	Rwanda	SE	Sweden
BL	Saint Barthélemy	CH	Switzerland
SH	Saint Helena, Ascension And Tristan Da Cunha	SY	Syrian Arab Republic
KN	Saint Kitts And Nevis	TW	Taiwan, Province of China
LC	Saint Lucia	TJ	Tajikistan

MF	Saint Martin (French Part)	TZ	Tanzania, United Republic Of
PM	Saint Pierre And Miquelon	TH	Thailand
VC	Saint Vincent And The Grenadines	TL	Timor-Leste
WS	Samoa	TG	Togo
SM	San Marino	TK	Tokelau
ST	Sao Tome And Principe	TO	Tonga
SA	Saudi Arabia	TT	Trinidad and Tobago
SN	Senegal	TN	Tunisia
RS	Serbia	TR	Turkey
SC	Seychelles	TM	Turkmenistan
SL	Sierra Leone	TC	Turks And Caicos Islands
SG	Singapore	TV	Tuvalu
SX	Sint Maarten (Dutch Part)	UG	Uganda
SK	Slovakia	UA	Ukraine
SI	Slovenia	AE	United Arab Emirates
SB	Solomon Islands	GB	United Kingdom
SO	Somalia	US	United States
ZA	South Africa	UM	United States Minor Outlying Islands
UY	Uruguay	EH	Western Sahara
UZ	Uzbekistan	YE	Yemen
VU	Vanuatu	ZM	Zambia
VE	Venezuela, Bolivarian Republic of	ZW	Zimbabwe
VN	Viet Nam	X1	Kosovo
VG	Virgin Islands, British	X2	Kaliningrad
VI	Virgin Islands, U.S.		
WF	Wallis And Futuna		

Flows

Code	Name
INDPROD	Production
OSOURCES	From Other Sources
TOTIMPSB	Imports
TOTEXPSB	Exports
REFGROUT	Refinery Output
RECEIPTS	Receipts
PTRANSF	Products Transferred
TRANSBAK	Products Transferred / Backflows (Primary oil)
DIRECUSE	Direct Use
IPTRANSF	Interproduct Transfers
STOCKCH	Stock Change
STATDIFF	Statistical Difference
REFINOBS	Refinery Intake
TOTDEMO	Demand (oil) / Gross Inland Deliveries (Observed) (gas)
CLOSTLV	Closing Stocks
IMPLNG	Imports LNG
IMPPIP	Imports Pipeline
EXPLNG	Exports LNG
EXPPIP	Exports Pipeline
TOTDEMC	Gross Inland Deliveries (Calculated)
MAINTOT	Gross Inland Deliveries of which: Electricity and Heat Generation
CONVER	Conversion Factor

Products

Code	Name
CRUDEOIL	Crude Oil
NGL	NGL
OTHERCRUDE	Other (Primary oil)
TOTCRUDE	Total (Primary oil)
LPG	LPG
NAPHTHA	Naphtha
GASOLINE	Gasoline
KEROSENE	Kerosene
JETKERO	Of Which: Kerosene Type Jet Fuel
GASDIES	Gas/Diesel Oil
RESFUEL	Fuel Oil
ONONSPEC	Other Oil Products
TOTPRODS	Total Oil Products
NATGAS	Natural Gas
LNG	Liquefied Natural Gas

Units

Code	Name
KBD	Thousand Barrels per day (kb/d)
KBBL	Thousand Barrels (kbbl)
KL	Thousand Kilolitres (kl)
KTONS	Thousand Metric Tons (kmt)
CONVBBL	Conversion factor barrels/ktons
M3	Million Metres Cubed

TJ	Terajoules
M3_T	Metres Cubed per Tonne

Appendix C – Rate Limits

To prevent abuse and ensure fair access to the API we employ a per user rate limit for requests to our endpoints of 100 requests per minute. This means that you can make a maximum of 100 requests in each rolling 60 second period.

Request periods start from the time of your first request so if you make your first request at a time of 13:45:20 you can then make a maximum of a further 99 requests until 13:46:20. At 13:46:20 the rate will reset and you can make another 100 requests from the time of your next request.

Once you exceed the rate limit you will receive 429 (Too Many Requests) response codes to any request sent until the rate limit resets.

All responses to requests to our endpoints will contain custom headers explaining the rate limit:

Header Key	Data Type	Description
X-Rate-Limit-Limit	String	Details of the Rate Limit
X-Rate-Limit-Remaining	Int	The number of requests you have remaining in the current request period
X-Rate-Limit-Reset	DateTime	The date and time that the current request period ends