

## About the API

The OpenAQ API provides open access to global air quality data, following REST principles with resource-oriented URLs, standard HTTP response codes, and JSON-formatted responses. OpenAQ focuses on criteria air pollutants, primarily aggregating PM2.5, PM10, SO2, NO2, CO, O3, BC, relative humidity and temperature measurement data. For a limited set of locations, we have data for PM1, PM4, CO2, NO, NOx, CH4 & UFP.

The data on OpenAQ do not represent all air quality monitoring data in the world; they are data that OpenAQ has discovered or been introduced to and that are publicly available. We welcome new data contributions.

All data accessed through the API is public. However, you are responsible for complying with applicable third-party terms. Please see our Terms of Use for very important information on intellectual property, allowable and considerate use, and more.

This documentation guide covers the OpenAQ API version 3. To get started, explore the available endpoints and consult the documentation for guidance.

## Retired versions

The OpenAQ version 1 and version 2 endpoints were retired on January 31, 2025 and are no longer accessible. Please migrate all existing code to version 3 as soon as possible. /v1 and /v2 endpoints will return HTTP 410 Gone responses.

This repository of global air quality data is brought to you by OpenAQ, a nonprofit organization founded in the U.S. OpenAQ was the first to aggregate ground-level ambient air quality data on an open-source platform. We provide unique value due to a combination of attributes that make these data easily accessible, interoperable, interpretable and actionable. We:

- harmonize the data into a single, uniform format so that they are easily comparable

- share the data in physical units rather than as an air quality index

- host near real-time and historical data

- share “metadata” (the data that provides the context for individual data points)

- make the data available programmatically via an API

- keep the underlying data-fetching software fully open

- provide tools for users of all abilities to access and use the data

If you value this service, please consider making a donation to sustain the OpenAQ platform far into the future.

## Terms of use

### Introduction

Welcome to OpenAQ, a data repository of open air quality data aggregated from sources across the globe. This page explains the terms for using data on the OpenAQ Platform (Terms of Use for Data Users) and the terms for sharing data on the OpenAQ Platform (Terms and Conditions for Data Publishers), together referred to as “Terms”.

Please read these Terms carefully. By using the OpenAQ Platform or by sharing data on the OpenAQ Platform, you acknowledge that, without limitation or qualification, you have read, understood, and agreed to the Terms and any future modifications to the Terms and to the collection and use of your information as set forth in our Privacy Policy.

OpenAQ reserves the right to modify these Terms without notice, in whole or in part, at our sole discretion. If we change these Terms, we will post the changes here and any changes shall become effective as of the time they are posted. Your continued use of the OpenAQ Platform or continued sharing of data on the OpenAQ Platform will signify your continued agreement to these Terms as they may be revised.

#### OpenAQ Terms of Use for Data Users

As a condition of using the OpenAQ Platform, you warrant to OpenAQ that you agree to these Terms of Use, including any future modifications by us. You also warrant that your use complies with all applicable laws where you are located. If you do not accept these Terms of Use, do not use the OpenAQ Platform.

#### Using the OpenAQ Platform

For programmatic access to data on the OpenAQ Platform or to use certain features on OpenAQ Explorer, users must register for an API key and account.

Users must provide a valid name and valid email address during the registration process. This email address must not be from a temporary or disposable email service.

An individual is permitted to register for only one API key. An entity may have more than one user, and thus more than one API key, but may not use multiple accounts to over-consume our services or otherwise abuse the system. We monitor usage patterns to detect and prevent such activities, ensuring a fair environment for all users.

As an account user, you acknowledge that you are responsible for all activity occurring under the use of your account. You are responsible for maintaining the security of your account and password. Unauthorized use of your API key, including transfer to another user, is prohibited.

To safeguard the integrity and security of our data platform, downloading data is strictly prohibited unless done through registered and authorized use. Users must utilize provided APIs, export functions, or other officially sanctioned methods for accessing data.

Unauthorized methods, including but not limited to scraping data from visualizations or other interface elements, are expressly forbidden. Violations of this policy will result in immediate suspension of access, revocation of user credentials, and potential legal action. This measure ensures that our data remains accurate, secure, and available for legitimate, registered users.

Users are permitted to run, modify, and self-host the open-source software components of our tooling, including the API codebase, in accordance with the applicable open-source license. However, utilizing our official, hosted API implementation to develop products or services that substantially duplicate or directly compete with OpenAQ's core offerings is

prohibited. This restriction applies specifically to the use of our hosted API service, not to self-hosted instances or modified versions of the open-source code. Users leveraging our hosted API agree to do so in a manner that complements rather than replicates our primary services. OpenAQ reserves the right to suspend or terminate access to our hosted API for users found to be in violation of this clause. This provision does not limit the rights granted under the open-source license for the software itself, but governs the use of our hosted API service.

#### Intellectual Property and Attribution

OpenAQ aggregates air quality data from government agencies and other sources. We only aggregate data that, to the best of our knowledge, has been made available for redistribution and use throughout the world. However, we provide no assurance that the data provided may be used free of any third-party claims. In some jurisdictions, copyright and/or laws and regulations may apply to OpenAQ source data.

We provide attribution to the source of data wherever required and whenever possible via our web interface and APIs. Users of the OpenAQ platform are solely responsible for their use of the data and for compliance with any applicable laws and third-party terms. OpenAQ users must therefore review and comply with any terms published by data providers.

A number of our sources provide their air quality data under terms that require source attribution—sometimes under Creative Commons licenses or open government licenses, and sometimes under bespoke terms. Regardless of such requirements, we believe in the importance of attributing data sources. For this reason, OpenAQ provides attribution, often in the form of a URL, to the underlying source of data ingested onto the platform. When required, OpenAQ users must acknowledge the original source(s) of any data they use, following the provider's terms, and OpenAQ encourages users to acknowledge all original sources, even if the source does not require it.

Attribution to OpenAQ as the source data is also required when using OpenAQ services for accessing the data.

#### Considerate Use

You may not use the OpenAQ Platform in any manner which could damage, disable, overburden, or impair the Platform or interfere with any other party's use and enjoyment of the Platform. For example, through your use of the API key you agree not to, at OpenAQ's sole discretion, use an unreasonable amount of bandwidth and not leave requests running in perpetuity if data is no longer needed.

#### Access to Higher Rate Limits

Anyone wishing to access OpenAQ Platform at rates higher than the documented rate limit must comply with the payment terms and API License and Services Agreement required by OpenAQ.

#### Disclaimer

**IMPORTANT:** Although OpenAQ strives to make the aggregated data true to its source and as useful as possible, we cannot guarantee its accuracy or suitability for any user's purpose, and

any use of the data is at the user's own risk. Data from the OpenAQ Platform is offered as-is, without warranty of any kind, whether express or implied, including all implied warranties of merchantability, fitness for a particular purpose, title, and non-infringement. OpenAQ makes no guarantee that the OpenAQ Platform or data will meet any user's requirements, operate without interruption, achieve any intended result, or be accurate, complete, or error-free.

OpenAQ reserves the right to modify or discontinue, temporarily or permanently, the OpenAQ API with or without notice at any time. You agree that OpenAQ shall not be liable to your or to any third party for any modification, suspension or discontinuance of the OpenAQ API.

#### Code of Conduct

By using the OpenAQ Platform you acknowledge and agree to abide by the principles in our Code of Conduct.

#### Violation of Terms of Use for Users

If a user is found to be in violation of the Terms of Use, OpenAQ reserves the right to suspend or terminate access to the platform.

#### OpenAQ Terms and Conditions for Data Publishers

OpenAQ aggregates third-party air quality data that is in the public domain or has been shared freely by data publishers who choose to share their data with OpenAQ to redistribute via the OpenAQ Platform.

In the latter case, the data publisher grants a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable license to reproduce, adapt, publicly display, publicly perform, sublicense, and distribute the data and any derivative works. The form, nature, and content of the data available through the Platform may change without any notice to the data publisher. OpenAQ may also remove access to the data at any time without any notice to the data publisher.

If you believe your data has been provided through the OpenAQ platform without your permission, please see our DMCA (Copyright Infringement) Policy.

#### Quick start

This guide shows how to fetch data from the OpenAQ API. As an example, we will retrieve location details for the "New Delhi" station in New Delhi, India, with OpenAQ location ID 8118.

This tutorial uses curl to make HTTP requests and assumes you are familiar with using the command line on your computer.

Run `curl --version` in the command line to see if curl exists on your machine. If version details show, curl exists. If you see `command not found: curl`, download and install it. For more details, visit the curl download page at <https://curl.se/download.html>.

Sign up for an OpenAQ API key at <https://explore.openaq.org>. After signing up, find your API key in your settings. Use this key to authenticate requests.

### Caution

Treat your API key as you would a password. Do not share your API key with others. If your API key leaks, you can request a new one in your OpenAQ Explorer account.

Run the following curl command, including your API key in the X-API-Key header. Replace YOUR-OPENAQ-API-KEY with your API key (placeholder value will not work).

Terminal window

```
curl --request GET \  
--url "https://api.openaq.org/v3/locations/8118" \  
--header "X-API-Key: YOUR-OPENAQ-API-KEY"
```

The JSON response will look something like the following:

```
{  
  "meta": {  
    "name": "openaq-api",  
    "website": "/",  
    "page": 1,  
    "limit": 100,  
    "found": 1  
  },  
  "results": [  
    {  
      "id": 8118,  
      "name": "New Delhi",  
      "locality": "India",  
      "timezone": "Asia/Kolkata",  
      "country": { "id": 9, "code": "IN", "name": "India" },  
      "owner": { "id": 4, "name": "Unknown Governmental Organization" },  
      "provider": { "id": 119, "name": "AirNow" },  
      "isMobile": false,  
      "isMonitor": true,  
      "instruments": [{ "id": 2, "name": "Government Monitor" }],  
      "sensors": [  
        {  
          "id": 23534,  
          "name": "pm25 µg/m³",  
          "parameter": {  
            "id": 2,  
            "name": "pm25",  
            "units": "µg/m³",  
            "displayName": "PM2.5"  
          }  
        }  
      ]  
    }  
  ]  
}
```

```

    }
  }
],
"coordinates": { "latitude": 28.63576, "longitude": 77.22445 },
"licenses": [
  {
    "id": 33,
    "name": "US Public Domain",
    "attribution": {
      "name": "Unknown Governmental Organization",
      "url": null
    },
    "dateFrom": "2016-01-30",
    "dateTo": null
  }
],
"bounds": [77.22445, 28.63576, 77.22445, 28.63576],
"distance": null,
"datetimeFirst": {
  "utc": "2016-11-09T19:00:00Z",
  "local": "2016-11-10T00:30:00+05:30"
},
"datetimeLast": {
  "utc": "2024-12-13T14:30:00Z",
  "local": "2024-12-13T20:00:00+05:30"
}
}
]
}

```

## API Key

The OpenAQ API uses API keys to authenticate requests and manage access. These keys help control usage and enforce rate limits, ensuring that the service remains available and responsive for all users.

## Managing Your API Key

Sign up at [explore.openaq.org/register](https://explore.openaq.org/register) to get your API key. You can access and rotate your key from the OpenAQ Explorer account settings page [explore.openaq.org/account](https://explore.openaq.org/account).

## Caution

Treat your API key as you would a password. Do not share your API key with others. If your API key leaks, you can request a new one in your OpenAQ Explorer account.

## Using Your API Key

Include your API key in the X-API-Key header of each request. For example:

Terminal window

```
curl --request GET \  
--url "https://api.openaq.org/v3/locations/2178" \  
--header "X-API-Key: YOUR-OPENAQ-API-KEY"
```

Replace YOUR-OPENAQ-API-KEY with your actual key. This key verifies your requests and ensures they are processed.

Rate limits

OpenAQ limits the number of API requests you can make in a set time to ensure fair access for all users and prevent overuse. API rate limits are scoped to the user API key.

OpenAQ provides a very generous rate limit. If your needs exceed this rate limit, we offer a custom pricing option for a higher rate limit.

General use	Custom use
Free	Contact us for pricing
Rate limits	
60 / minute	
2,000 / hour	

Rate limits

Higher limits available depending on needs

Sign up      Contact us  
Requirements & Terms:

Do not leave requests running in perpetuity if data are no longer needed.

Adherence to OpenAQ Terms of Use is required, including source attribution. We reserve the right to suspend or terminate access should misuse of the OpenAQ Platform occur.

Requirements & Terms:

Jointly agreed upon Custom Use Agreement.

Adherence to OpenAQ Terms of Use is required, including source attribution. We reserve the right to suspend or terminate access should misuse of the OpenAQ Platform occur.

If a rate limit is exceeded, the API returns a 429 “Too many requests” HTTP status code. To continue, wait until the next rate limit period starts as described in OpenAQ’s rate limit policy. Avoid making more requests during this time to prevent further errors.

Caution

Exceeding API rate limits repeatedly can lead to either a temporary or permanent ban from the OpenAQ API. To maintain uninterrupted access, plan your API requests carefully.

## Rate limit headers

The OpenAQ API includes response headers to help you manage request rates and adhere to limits. These headers give information about your current request usage, remaining capacity, and reset times, allowing you to plan your API calls effectively.

### x-ratelimit-used

Shows the number of requests that have been made during the current rate limit period. This header helps users track their usage and avoid exceeding limits.

### x-ratelimit-reset

Provides a timestamp indicating when the rate limit period will reset. This header helps users plan when to start making requests again.

### x-ratelimit-limit

Specifies the maximum number of requests allowed within the current rate limit period. This header helps users understand the upper bound for the number of requests they can make.

### x-ratelimit-remaining

Indicates the number of requests remaining before reaching the rate limit. This header helps users avoid hitting the limit and encountering rate limit errors.

## Example headers

Terminal window

```
HTTP/2 200
```

```
content-type: application/json
```

```
content-length: 1681
```

```
date: Sat, 24 Aug 2024 21:20:25 GMT
```

```
x-ratelimit-used: 1
```

```
x-ratelimit-reset: 60
```

```
x-ratelimit-limit: 10
```

```
x-ratelimit-remaining: 9
```

```
...
```

In the example HTTP response above, the rate limit headers reveal:

You can make a total of 10 requests in this period.

You have used 1 request so far.

You have 9 requests remaining before reaching the limit.

The limit will reset in 60 seconds, at which point the request count will start over.

## Pagination

Air quality monitoring produces a significant volume of data. Therefore, the OpenAQ API uses pagination to return a subset of the results. By default, a request will return a maximum of 100 results in a single page. This limit can be adjusted using the limit query parameter, up



to a maximum of 1,000 results per page, i.e., limit=1000. Using this limit in conjunction with the page query parameter, you can access a large result set across multiple requests.

#### Tip

For resources with a high volume of results, such as /measurements and /hours using the datetimeFrom and datetimeTo to limit to a single year or less is highly recommended. Querying the entire dataset and paging using the page query parameter alone can be slow. Adding a smaller time range allows the query to leverage database indexes and will generally result in more performant queries.

The total number of records in a result set is available in the JSON response body in the meta object under the found key. The meta object in the response JSON also provides the limit under the limit key:

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 16492
  },
  "results": [...]
}
```

#### Dates, times and timezones

##### Datetimes

Dates and datetimes in the OpenAQ API follow the ISO-8601 datetime format, e.g., YYYY-MM-DDTHH:MM:SS.SSSSZ.

Data on OpenAQ are harmonized to follow an exclusive time-ending standard. This means a time stamp for an hourly measurement 03:00 represents the data from 02:00 until 02:59.

##### Dates

Dates represent the start of a time period. For example, a date like 2024-01-01 represents January 1, 2024, and a year like 2024-01-01 represents the calendar year, 2024. This contrasts (see above) with how OpenAQ handles datetimes, where they represent the end of a time period. In other words, while a date marks the start of a specific day or year, a datetime indicates when that period has concluded. This distinction is crucial for accurately interpreting the data retrieved from the OpenAQ API, as it ensures clarity in understanding when data collection periods begin and end.

##### Timezones

Handling time zones correctly is crucial due to the global nature of the dataset, which includes air quality data from monitoring stations across different time zones. When requesting data from the OpenAQ API, timestamps should ideally include a timezone in the

ISO-8601 format (e.g., 2024-08-25T15:00:00Z for UTC time). This ensures that the data retrieved accurately reflects the time it was recorded, regardless of the station's location. If a timezone is not provided in the timestamp, the API automatically assumes the local time for the station where the data was collected. This default behavior is important to understand as it can affect data comparisons across different regions and time zones.

To ensure timezones are clearly represented in results, the API returns a datetime object which includes both UTC time and local time.

```
{  
  "utc": "2019-07-11T20:00:00Z",  
  "local": "2019-07-11T14:00:00-06:00"  
}
```

The OpenAQ API provides the timezone ID string for a given monitoring location under the timezone key. This follows the tzid in the Time Zone Database

### Geospatial queries

The OpenAQ API provides two methods for making geospatial queries: bounding box and point and radius. Each provides a method for selecting coordinate points that intersect with a geographic area. Choose only one of these two methods at a time. If both queries appear in a URL, the API returns an HTTP 422 “Unprocessable Content” error.

### Bounding box

A bounding box query identifies geographic features within a rectangular area defined by specific minimum and maximum longitude and latitude values. This can be helpful for finding features within arbitrary geographic areas.

The bounding box query parameter for the OpenAQ API is bbox. This parameter takes a comma delimited list of 4 WGS84 (EPSG:4326) coordinates in the form minimum X, minimum Y, maximum X, maximum Y, where X is the longitude and Y is latitude.

### Conditions

If coordinate values used with the bbox query parameter are outside the bounds of WGS84 (-180,-90,180,90), the API returns an HTTP 422 “Unprocessable Content” error.

If bbox appears in the query with the coordinates or radius query parameters, the API returns an HTTP 422 “Unprocessable Content” error.

### Example

To query a bounding box around Accra, Ghana, the query parameter for the URL looks something like this.

```
?bbox=5.488869,-0.396881,5.732144,-0.021973
```

This bounding box query can be represented as the following polygon:

### Libraries

Libraries can simplify and enhance the way your application connects with the OpenAQ API by providing pre-built code tailored to specific programming environments. Once integrated into your project, these libraries allow you to interact with the OpenAQ API more efficiently and consistently.

#### Official OpenAQ libraries

OpenAQ maintains official client libraries for interacting with the OpenAQ API with different programming language including the Python and R. These libraries are the recommended tool for most developers when interacting with the OpenAQ API. These tools provides a standardized interface that simplifies integration and ensures that developers are following best practices. By using these libraries developers can more easily access the full range of OpenAQ's features and data, and benefit from a consistent, well-documented approach that aligns with the latest updates and functionalities of the API.

#### Examples

This page provides examples of common queries and how to use the OpenAQ API to address them. For more complex solutions we recommend using one of the official API clients or SDKs, see Libraries for more information.

#### Note

Examples on this page will not work as-is. All requests use a placeholder API Key value, YOUR-OPENAQ-API-KEY. This key will not work and will return an HTTP 401 error, replace with your own API key.

#### Filtering locations by parameter

cURL

Python

R

Terminal window

```
curl --request GET \  
  --url "https://api.openaq.org/v3/locations?parameters_id=2&limit=1000" \  
  --header "X-API-Key: YOUR-OPENAQ-API-KEY"
```

Where parameters\_id 2 is the ID for PM2.5 and limits the number results to 1,000 per page.

#### Finding locations near a point

cURL

Python

R

Terminal window

```
curl --request GET \  
  --url \  
"https://api.openaq.org/v3/locations?coordinates=136.90610,35.14942&radius=12000&limit=1000" \  
  --header "X-API-Key: YOUR-OPENAQ-API-KEY"
```

Where 136.90610,35.14942 is the central point in latitude,longitude and the radius is 12,000 meters (12km) and limits the number results to 1,000 per page. See the Geospatial queries guide for more information on geospatial querying. This radius query represents an area like:

## Countries

The Countries resource lists countries in the OpenAQ data set and details the air quality parameters available for each. The list only includes countries that have monitoring locations from which OpenAQ ingests data, and the Country value is derived from the coordinates provided by the upstream provider. OpenAQ uses the Natural Earth 10m dataset for country boundaries, see this disclaimer. OpenAQ uses the ISO 3166-1 alpha 2 country codes, which may differ from that reported by the upstream provider.

## Purpose and Use

The Countries resource helps users discover which countries are in the OpenAQ data set. It also shows the air quality metrics for each country. This makes it possible to filter and analyze air quality data by country.

MapLibre | © OpenStreetMap contributors

## Key fields

id - the unique identifier used within the OpenAQ system for a country

code - the ISO 3166-1 alpha 2 code for the country

name - the English name of the country

## Example response payload

<https://api.openaq.org/v3/countries/42>

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 1
  },
  "results": [
    {
      "id": 42,
      "code": "KZ",
      "name": "Kazakhstan",
      "datetimeFirst": "2018-07-27T17:00:00Z",
      "datetimeLast": "2024-08-31T21:07:06.731Z",
      "parameters": [
        {
          "id": 1,
          "name": "pm10",
          "units": "µg/m³",
          "displayName": null
        }
      ]
    }
  ]
}
```

```

{
  "id": 2,
  "name": "pm25",
  "units": "µg/m³",
  "displayName": null
},
{
  "id": 19,
  "name": "pm1",
  "units": "µg/m³",
  "displayName": null
},
{
  "id": 98,
  "name": "relativehumidity",
  "units": "%",
  "displayName": null
},
{
  "id": 100,
  "name": "temperature",
  "units": "c",
  "displayName": null
},
{
  "id": 125,
  "name": "um003",
  "units": "particles/cm³",
  "displayName": null
}
]
}
]
}

```

Table of current countries values

id	name	ISO A2
130	Afghanistan	AF
122	Algeria	DZ
129	Andorra	AD
176	Antarctica	AQ
6	Argentina	AR
70	Armenia	AM
177	Australia	AU
89	Austria	AT
64	Azerbaijan	AZ
250	Bahrain	BH

128	Bangladesh	BD	
60	Belgium	BE	
158	Belize	BZ	
132	Bosnia and Herzegovina		BA
45	Brazil	BR	
110	Bulgaria	BG	
150	Burkina Faso	BF	
57	Cambodia	KH	
147	Cameroon	CM	
156	Canada	CA	
115	Chad	TD	
3	Chile	CL	
10	China	CN	
138	Colombia	CO	
29	Costa Rica	CR	
103	Croatia	HR	
185	Curaçao	CW	
8	Cyprus	CY	
49	Czech Republic		CZ
96	Côte d'Ivoire	CI	
32	Democratic Republic of the Congo		CD
71	Denmark	DK	
7	Dhekelia	-99	
137	Ecuador	EC	
162	Egypt	EG	
51	Estonia	EE	
14	Ethiopia	ET	
55	Finland	FI	
22	France	FR	
50	Germany	DE	
152	Ghana	GH	
154	Gibraltar	GI	
80	Greece	GR	
118	Guatemala	GT	
83	Guinea	GN	
24	Guyana	GY	
136	Honduras	HN	
167	Hong Kong	HK	
75	Hungary	HU	
192	Iceland	IS	
9	India	IN	
1	Indonesia	ID	
90	Iraq	IQ	
78	Ireland	IE	
11	Israel	IL	
91	Italy	IT	
190	Japan	JP	

224	Jersey	JE	
144	Jordan	JO	
42	Kazakhstan	KZ	
17	Kenya	KE	
65	Kosovo	XK	
116	Kuwait	KW	
69	Kyrgyzstan	KG	
68	Lao PDR	LA	
52	Latvia	LV	
44	Lithuania	LT	
58	Luxembourg	LU	
182	Madagascar	MG	
18	Malawi	MW	
2	Malaysia	MY	
239	Maldives	MV	
98	Mali	ML	
223	Malta	MT	
219	Mauritius	MU	
157	Mexico	MX	
142	Moldova	MD	
121	Monaco	MC	
47	Mongolia	MN	
131	Montenegro	ME	
27	Morocco	MA	
123	Mozambique	MZ	
127	Myanmar	MM	
145	Nepal	NP	
94	Netherlands	NL	
180	New Zealand	NZ	
100	Nigeria	NG	
62	North Macedonia		MK
53	Norway	NO	
40	Oman	OM	
109	Pakistan	PK	
12	Palestine	PS	
139	Paraguay	PY	
5	Peru	PE	
183	Philippines	PH	
77	Poland	PL	
141	Portugal	PT	
211	Puerto Rico	PR	
105	Qatar	QA	
222	Republic of Cabo Verde		CV
25	Republic of Korea		KR
74	Romania	RO	
48	Russian Federation		RU
126	Rwanda	RW	

38	Saint-Martin	MF
112	San Marino	SM
106	Saudi Arabia	SA
99	Senegal	SN
97	Serbia	RS
231	Singapore	SG
76	Slovakia	SK
104	Slovenia	SI
37	South Africa	ZA
15	South Sudan	SS
67	Spain	ES
184	Sri Lanka	LK
86	Sudan	SD
54	Sweden	SE
92	Switzerland	CH
189	Taiwan	TW
43	Tajikistan	TJ
111	Thailand	TH
166	The Gambia	GM
199	Trinidad and Tobago	TT
73	Tunisia	TN
66	Turkey	TR
143	Turkmenistan	TM
133	Uganda	UG
34	Ukraine	UA
59	United Arab Emirates	AE
79	United Kingdom	GB
155	United States	US
46	Uruguay	UY
41	Uzbekistan	UZ
56	Vietnam	VN
81	Zambia	ZM
108	Zimbabwe	ZW

## Instruments

The Instruments resource provides information about the air quality measuring devices that gather the data. Each instrument has details such as its name, type, and manufacturer.

## Purpose and Use

The Instruments resource helps users learn about the instruments gathering air quality data. This resource also highlights whether an instrument is used for official monitoring. By using the information, users can understand what kinds of devices gather air quality data, which is useful when analyzing the reliability and scope of the data.

## Key fields

id: A unique identifier for each instrument in the system.

name: The name of the device (e.g., “Clarity Node-S” or “BAM 1020”).



isMonitor: A field showing whether the device is used for official monitoring.

manufacturer: Details about the company or organization that made the device, including its ID and name. See the Manufacturers resource page for more information.

Example response payload

<https://api.openaq.org/v3/instruments/2>

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 1
  },
  "results": [
    {
      "id": 2,
      "name": "Government Monitor",
      "isMonitor": true,
      "manufacturer": {
        "id": 4,
        "name": "Unknown Governmental Organization"
      }
    }
  ]
}
```

## Latest

The Latest resource in the OpenAQ API provides the most recent air quality measurement from a sensor. Each record provides the measurement time, value, location coordinates, and sensor information.

## Note

The latest value does not represent the most recent measurement ingested by OpenAQ for a given sensor, but instead represents the last measurement value in the series. Due to the nature of how some upstream providers report data, measurements are not necessarily reported in order. Polling latest resources for measurement values does not guarantee complete data coverage as measurements may be ingested out of their time series order and not show up as a “latest” value.

## Purpose and Use

The Latest resource provides access to the newest air quality data from sensors. It shows recent measurements and their geographic location, allowing users to track air quality trends quickly and make timely decisions.

The Latest resource is accessible through either a single Location, showing latest values for all sensors available at that Location, or by parameter, showing the latest value across all sensors that measure that parameter, e.g.:

<https://api.openaq.org/v3/locations/2178/latest>

or

<https://api.openaq.org/v3/parameters/2/latest>

#### Key Fields

**datetime:** the time of the measurement, provided in UTC and local time. See Dates, times and timezones for more information.

**value:** the sensor measurement reading.

**coordinates:** the location in WGS84 latitude and longitude of the reading.

**sensorsId:** the unique ID for the sensor that took the reading. See the Sensors resource page for more information.

**locationsId:** the unique ID for the Location where the sensor is situated. See the Locations resource page for more information.

Example response payload

<https://api.openaq.org/v3/locations/2178/latest>

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 8
  },
  "results": [
    {
      "datetime": {
        "utc": "2024-09-25T22:00:00Z",
        "local": "2024-09-25T16:00:00-06:00"
      },
      "value": 0.0023,
      "coordinates": {
        "latitude": 35.1353,
        "longitude": -106.584702
      },
      "sensorsId": 3916,
      "locationsId": 2178
    },
    {
      "datetime": {
        "utc": "2024-09-25T22:00:00Z",
```

```
    "local": "2024-09-25T16:00:00-06:00"
  },
  "value": 0.0005,
  "coordinates": {
    "latitude": 35.1353,
    "longitude": -106.584702
  },
  "sensorId": 3918,
  "locationId": 2178
},
{
  "datetime": {
    "utc": "2024-09-25T22:00:00Z",
    "local": "2024-09-25T16:00:00-06:00"
  },
  "value": 4.4,
  "coordinates": {
    "latitude": 35.1353,
    "longitude": -106.584702
  },
  "sensorId": 3920,
  "locationId": 2178
},
{
  "datetime": {
    "utc": "2024-09-25T22:00:00Z",
    "local": "2024-09-25T16:00:00-06:00"
  },
  "value": 0.067,
  "coordinates": {
    "latitude": 35.1353,
    "longitude": -106.584702
  },
  "sensorId": 3917,
  "locationId": 2178
},
{
  "datetime": {
    "utc": "2024-09-25T22:00:00Z",
    "local": "2024-09-25T16:00:00-06:00"
  },
  "value": 29.0,
  "coordinates": {
    "latitude": 35.1353,
    "longitude": -106.584702
  },
  "sensorId": 3919,
```

```

    "locationsId": 2178
  },
  {
    "datetime": {
      "utc": "2024-09-25T22:00:00Z",
      "local": "2024-09-25T16:00:00-06:00"
    },
    "value": 0.2,
    "coordinates": {
      "latitude": 35.1353,
      "longitude": -106.584702
    },
    "sensorsId": 25227,
    "locationsId": 2178
  },
  {
    "datetime": {
      "utc": "2024-09-25T22:00:00Z",
      "local": "2024-09-25T16:00:00-06:00"
    },
    "value": 0.002,
    "coordinates": {
      "latitude": 35.1353,
      "longitude": -106.584702
    },
    "sensorsId": 4272103,
    "locationsId": 2178
  },
  {
    "datetime": {
      "utc": "2024-09-25T22:00:00Z",
      "local": "2024-09-25T16:00:00-06:00"
    },
    "value": 0.0,
    "coordinates": {
      "latitude": 35.1353,
      "longitude": -106.584702
    },
    "sensorsId": 4272226,
    "locationsId": 2178
  }
]
}

```

## Licenses

The Licenses resource in the OpenAQ API provides information about the licenses governing the use of data within the API.

## Purpose and Use

The Licenses resource lists the licenses that govern use of the data in OpenAQ. Each license details permissions and restrictions, such as whether attribution is required and whether commercial use, modifications, or redistribution are allowed. This resource helps users comply with any licensing requirements. As per OpenAQ's Terms of Use, compliance is required.

## Key fields

id: A unique identifier for each license.

name: The name of the license.

commercialUseAllowed: Indicates whether the license permits the use of data for commercial purposes.

attributionRequired: Specifies whether users must give credit to the original source when using the data.

shareAlikeRequired: States whether derivative works must be licensed under the same terms as the original.

modificationAllowed: Shows whether users can alter the data.

redistributionAllowed: Indicates whether users can distribute the data.

sourceUrl: Provides a link to the official license terms for further details.

Example response payload

<https://api.openaq.org/v3/licenses/41>

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 1
  },
  "results": [
    {
      "id": 41,
      "name": "CC BY 4.0 DEED",
      "commercialUseAllowed": true,
      "attributionRequired": true,
      "shareAlikeRequired": true,
      "modificationAllowed": true,
      "redistributionAllowed": true,
      "sourceUrl": "https://creativecommons.org/licenses/by/4.0/"
    }
  ]
}
```

## Locations

The Locations resource in the OpenAQ API provides details about air quality monitoring stations, including their name, geographic coordinates, and time zone information. It also identifies the responsible organization (when known) and data provider. Each location measures various parameters and provides information on the sensors, including whether the station is mobile or stationary, and whether the station is actively monitoring air quality. Licensing and attribution details clarify data usage rights.

### Purpose and Use

The Locations resource helps users find air quality stations in specific areas and understand details about the station, from what instrumentation it uses and which pollutants are measured to who runs the station and owns the data. This allows for analysis and appropriate use of the data.

### Key fields

id: a unique identifier for each location.

name: the name of the station.

country: the country where the station operates, including its ISO 3166-1 alpha 2 code and name. See the Countries resource page for more information.

provider: the organization or individual that facilitates the sharing of data. See the Providers resource for more information.

owner: the organization or individual responsible for the station. See the Owners resource page for more information.

coordinates: the geographic position, WGS84 latitude and longitude, of the station. For mobile locations, this is the first geographic point measured.

instruments: a list of devices used at the location. See the Instruments resource for more information.

sensors: a list of sensors that measure pollutants, including the parameter measured. See the Sensors resource for more information.

licenses: a list of licenses, with attribution information and a time period that each license covers. See the License resource for more information.

timezone:

Example response payload

<https://api.openaq.org/v3/locations/2178>

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 1
  },
  "results": [
    {
      "id": 2178,
      "name": "Del Norte",
      "locality": "Albuquerque",
```

```
"timezone": "America/Denver",
"country": {
  "id": 155,
  "code": "US",
  "name": "United States"
},
"owner": {
  "id": 4,
  "name": "Unknown Governmental Organization"
},
"provider": {
  "id": 119,
  "name": "AirNow"
},
"isMobile": false,
"isMonitor": true,
"instruments": [
  {
    "id": 2,
    "name": "Government Monitor"
  }
],
"sensors": [
  {
    "id": 25227,
    "name": "co ppm",
    "parameter": {
      "id": 8,
      "name": "co",
      "units": "ppm",
      "displayName": "CO"
    }
  },
  {
    "id": 4272226,
    "name": "no ppm",
    "parameter": {
      "id": 35,
      "name": "no",
      "units": "ppm",
      "displayName": "NO"
    }
  },
  {
    "id": 3916,
    "name": "no2 ppm",
    "parameter": {
```

```
"id": 7,
"name": "no2",
"units": "ppm",
"displayName": "NO2"
},
{
  "id": 4272103,
  "name": "nox ppm",
  "parameter": {
    "id": 19840,
    "name": "nox",
    "units": "ppm",
    "displayName": "NOx"
  }
},
{
  "id": 3917,
  "name": "o3 ppm",
  "parameter": {
    "id": 10,
    "name": "o3",
    "units": "ppm",
    "displayName": "O3"
  }
},
{
  "id": 3919,
  "name": "pm10 µg/m3",
  "parameter": {
    "id": 1,
    "name": "pm10",
    "units": "µg/m3",
    "displayName": "PM10"
  }
},
{
  "id": 3920,
  "name": "pm25 µg/m3",
  "parameter": {
    "id": 2,
    "name": "pm25",
    "units": "µg/m3",
    "displayName": "PM2.5"
  }
},
{
  }
```



```

      "id": 3918,
      "name": "so2 ppm",
      "parameter": {
        "id": 9,
        "name": "so2",
        "units": "ppm",
        "displayName": "SO2"
      }
    },
    "coordinates": {
      "latitude": 35.1353,
      "longitude": -106.584702
    },
    "licenses": [
      {
        "id": 33,
        "name": "US Public Domain",
        "attribution": {
          "name": "Unknown Governmental Organization",
          "url": null
        },
        "dateFrom": "2016-01-30",
        "dateTo": null
      }
    ],
    "bounds": [-106.584702, 35.1353, -106.584702, 35.1353],
    "distance": null,
    "datetimeFirst": {
      "utc": "2016-03-06T20:00:00Z",
      "local": "2016-03-06T13:00:00-07:00"
    },
    "datetimeLast": {
      "utc": "2024-09-25T22:00:00Z",
      "local": "2024-09-25T16:00:00-06:00"
    }
  }
]
}

```

## Manufacturers

The Manufacturers resource in the OpenAQ API provides information on the companies that produce the air quality monitoring instruments.

## Purpose and Use

The Manufacturers resource allows users to identify who manufactured the instrument producing the air quality measurement. Understanding the origins of the equipment is useful for assessing the reliability of the instruments.

Note: This metadata is not available in all cases, and will be replaced with placeholder information.

#### Key fields

id: a unique identifier for each manufacturer.

name: the name of the manufacturer.

instruments: a list of instruments produced by the manufacturer, including their ID and name. See the Instruments resource page for more information.

Example response payload

<https://api.openaq.org/v3/manufacturers/4>

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 1
  },
  "results": [
    {
      "id": 4,
      "name": "Unknown Governmental Organization",
      "instruments": [
        {
          "id": 2,
          "name": "Government Monitor"
        }
      ]
    }
  ]
}
```

#### Measurements

The measurements resources in the OpenAQ API provide measurement values from sensors. Measurement values are available from a number of different resources as their original values and different levels of aggregation.

#### Purpose and Use

In addition to providing raw measurement values as reported by providers, the API provides options for rolling up data into different aggregation periods.

#### Resources

## Measurements

[https://api.openaq.org/v3/sensors/{sensors\\_id}/measurements](https://api.openaq.org/v3/sensors/{sensors_id}/measurements)

The measurements resource is the original measurement value as reported by the upstream source.

## Hours

[https://api.openaq.org/v3/sensors/{sensors\\_id}/hours](https://api.openaq.org/v3/sensors/{sensors_id}/hours)

The hours resource is the hourly average (mean) value of measurements.

## Note

Many sensors on the OpenAQ platform report hourly values. In the case where the upstream reporting period is hourly, the hours resource and measurements resource are equivalent.

## Days

[https://api.openaq.org/v3/sensors/{sensors\\_id}/days](https://api.openaq.org/v3/sensors/{sensors_id}/days)

The days resource is the daily average (mean), computed from the hourly average values from 01:00 to 0:00 in local time.

## Years

[https://api.openaq.org/v3/sensors/{sensors\\_id}/years](https://api.openaq.org/v3/sensors/{sensors_id}/years)

The years resource is the yearly average (mean), computed from the hourly average values from January 01 at 01:00 to December 31 0:00 in local time.

## Aggregation Periods

Additional aggregation is available and can be based on the above data groups

## Caution

Aggregating from measurements, e.g. [/v3/sensors/{sensors\\_id}/measurements/yearly](#), can be computationally slow and may not resolve, resulting in a 408 Request Timeout error. For most reliable results, limit queries to smaller time periods using the `date_from` and `date_to` query parameters.

## Hourly

[https://api.openaq.org/v3/sensors/{sensors\\_id}/measurements/hourly](https://api.openaq.org/v3/sensors/{sensors_id}/measurements/hourly)

The hourly aggregator returns the average (mean) value for each hour across the selected time range.

## Note

A call to the `/v3/sensors/{sensors_id}/measurements/hourly` is functionally the same as `/v3/sensors/{sensors_id}/hours`. The `/hours` resource is the preferred option.

#### Daily

[https://api.openaq.org/v3/sensors/{sensors\\_id}/measurements/daily](https://api.openaq.org/v3/sensors/{sensors_id}/measurements/daily)

[https://api.openaq.org/v3/sensors/{sensors\\_id}/hours/daily](https://api.openaq.org/v3/sensors/{sensors_id}/hours/daily)

The daily aggregator returns the average (mean) value for each day across the selected time range.

#### Note

A call to the `/v3/sensors/{sensors_id}/hours/daily` is functionally the same as `/v3/sensors/{sensors_id}/days`. The `/days` resource is the preferred option.

#### Monthly

[https://api.openaq.org/v3/sensors/{sensors\\_id}/measurements/monthly](https://api.openaq.org/v3/sensors/{sensors_id}/measurements/monthly)

[https://api.openaq.org/v3/sensors/{sensors\\_id}/hours/monthly](https://api.openaq.org/v3/sensors/{sensors_id}/hours/monthly)

[https://api.openaq.org/v3/sensors/{sensors\\_id}/days/monthly](https://api.openaq.org/v3/sensors/{sensors_id}/days/monthly)

The monthly aggregator returns the average (mean) value for each month across the selected time range.

#### Yearly

[https://api.openaq.org/v3/sensors/{sensors\\_id}/measurements/yearly](https://api.openaq.org/v3/sensors/{sensors_id}/measurements/yearly)

[https://api.openaq.org/v3/sensors/{sensors\\_id}/hours/yearly](https://api.openaq.org/v3/sensors/{sensors_id}/hours/yearly)

[https://api.openaq.org/v3/sensors/{sensors\\_id}/days/yearly](https://api.openaq.org/v3/sensors/{sensors_id}/days/yearly)

The yearly aggregator returns the average (mean) value for each year across the selected time range.

#### Note

A call to the `/v3/sensors/{sensors_id}/hours/yearly` is functionally the same as `/v3/sensors/{sensors_id}/years`. The `/years` resource is the preferred option.

#### Hour of day

[https://api.openaq.org/v3/sensors/{sensors\\_id}/hours/hourofday](https://api.openaq.org/v3/sensors/{sensors_id}/hours/hourofday)

The hourofday aggregator returns the average (mean) value for each hour in a 24 hour day, across the selected time range. This allows analyzing air quality trends on an hourly basis, offering insights into daily patterns and peak pollution times, such as diurnal patterns.

#### Day of week

[https://api.openaq.org/v3/sensors/{sensors\\_id}/measurements/dayofweek](https://api.openaq.org/v3/sensors/{sensors_id}/measurements/dayofweek)

[https://api.openaq.org/v3/sensors/{sensors\\_id}/hours/dayofweek](https://api.openaq.org/v3/sensors/{sensors_id}/hours/dayofweek)

The dayofweek aggregator returns the average (mean) value for each day in a week (Monday-Sunday), across the selected time range. This allows analyzing air quality trends on an daily basis, offering insights into daily patterns and peak pollution days.

#### Month of year

[https://api.openaq.org/v3/sensors/{sensors\\_id}/measurements/monthofyear](https://api.openaq.org/v3/sensors/{sensors_id}/measurements/monthofyear)

[https://api.openaq.org/v3/sensors/{sensors\\_id}/hours/monthofyear](https://api.openaq.org/v3/sensors/{sensors_id}/hours/monthofyear)

[https://api.openaq.org/v3/sensors/{sensors\\_id}/days/monthofyear](https://api.openaq.org/v3/sensors/{sensors_id}/days/monthofyear)

The monthofyear aggregator returns the average (mean) value for each month in a year (January-December), across the selected time range. This allows analyzing air quality trends on an monthly basis, offering insights into seasonal patterns.

#### Response details

##### Summary

The summary object provides summary statistics about a aggregated measurement value. The object includes the minimum and maximum values, quartile values (25th, 50th and 75th), average (mean), high percentile (98th) and low percentile (2nd) and standard deviation. When requesting measurements this field is null.

```
{
  "min": 0.001,
  "q02": 0.002,
  "q25": 0.025,
  "median": 0.033,
  "q75": 0.043,
  "q98": 0.05669999999999999,
  "max": 0.063,
  "avg": 0.03316917293233082,
  "sd": 0.013343557164562366
}
```

##### Coverage

The coverage object provides information on the coverage of measurements when aggregating values over time. When requesting measurements this field is will show expectedCount and observedCount values of 1, and percentComplete of 100.

```
{
  "expectedCount": 273,
```

```

"expectedInterval": "273:00:00",
"observedCount": 266,
"observedInterval": "266:00:00",
"percentComplete": 97.0,
"percentCoverage": 97.0,
"datetimeFrom": {
  "utc": "2024-01-01T10:00:00Z",
  "local": "2024-01-01T03:00:00-07:00"
},
"datetimeTo": {
  "utc": "2024-10-01T09:00:00Z",
  "local": "2024-10-01T03:00:00-06:00"
}
}

```

#### Key fields

value - The average (mean) value in aggregate, except in the measurements resource value is the original reported measurement value.

parameter - An object describing the parameter measured by the sensor.

period - An object that describes the period aggregate includes a label of the interval, the interval in hours, and a date range of the period aggregated.

coordinates - The coordinates of the measurement value for mobile monitoring. null for stationary monitoring.

summary - An object that provides summary statistics of the aggregated data. Includes minimum, maximum, median and quartile values. This object will be null for measurements

coverage - An object that provides coverage information of the aggregated data. Includes expected counts, observed count, percent complete and a date range for the values aggregated.

#### Example response payload

```

{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 1000,
    "found": ">1000"
  },
  "results": [
    {
      "value": 0.043,
      "parameter": {
        "id": 10,
        "name": "o3",
        "units": "ppm",
        "displayName": null
      },

```

```

"period": {
  "label": "raw",
  "interval": "01:00:00",
  "datetimeFrom": {
    "utc": "2016-03-06T19:00:00Z",
    "local": "2016-03-06T12:00:00-07:00"
  },
  "datetimeTo": {
    "utc": "2016-03-06T20:00:00Z",
    "local": "2016-03-06T13:00:00-07:00"
  }
},
"coordinates": null,
"summary": null,
"coverage": {
  "expectedCount": 1,
  "expectedInterval": "01:00:00",
  "observedCount": 1,
  "observedInterval": "01:00:00",
  "percentComplete": 100.0,
  "percentCoverage": 100.0,
  "datetimeFrom": {
    "utc": "2016-03-06T19:00:00Z",
    "local": "2016-03-06T12:00:00-07:00"
  },
  "datetimeTo": {
    "utc": "2016-03-06T20:00:00Z",
    "local": "2016-03-06T13:00:00-07:00"
  }
}
}
]
}

```

## Owners

The Owners resource in the OpenAQ API provides information about the people or organizations responsible for managing air quality monitoring stations and instruments. Owners may include governmental agencies, research institutions, other organizations that oversee air quality monitoring, and even individuals who have installed an air quality monitoring device on their property. The data “owner” may be the same as the data “provider” (see Providers).

## Purpose and Use

The Owners resource helps users identify who manages specific air quality monitoring stations. Knowing the owner offers context about the individual or organization behind the data collection, and to whom to attribute the data.

### Key fields

id: a unique identifier for each owner.

name: the name of the organization or individual responsible for managing the air monitoring location.

Example response payload

<https://api.openaq.org/v3/owners/3549>

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 1
  },
  "results": [
    {
      "id": 3549,
      "name": "City of Zabok"
    }
  ]
}
```

### Parameters

The Parameters resource in the OpenAQ API provides detailed information about the air quality parameters included in the dataset. This resource is essential for selecting and understanding the different air quality metrics available for analysis.

### Purpose and Use

The Parameters resource helps users identify and understand the various air quality parameters provided by the OpenAQ API. It details each parameter's name, units, and description, which supports users in choosing the relevant metrics for their air quality analysis and monitoring.

### Key fields

id: A unique identifier for each parameter.

name: The internal name used to reference the parameter in the API.

units: The unit of measurement for the parameter (e.g.,  $\mu\text{g}/\text{m}^3$  for mass concentration).

displayName: A user-friendly label for the parameter.

description: A brief explanation of what the parameter measures.

Example response payload

<https://api.openaq.org/v3/parameters/2>

```
{
  "meta": {
    "name": "openaq-api",
```



```

    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 1
  },
  "results": [
    {
      "id": 2,
      "name": "pm25",
      "units": "µg/m³",
      "displayName": "PM2.5",
      "description": "Particulate matter less than 2.5 micrometers in diameter mass concentration"
    }
  ]
}

```

## Providers

The Providers resource in the OpenAQ API provides details about the entities that supply air quality data to the platform. These providers may use APIs, websites, or other means to share their data with OpenAQ, ensuring access to air quality information from various sources.

## Purpose and Use

The Providers resource helps users identify the organizations contributing air quality data to the platform. By understanding who supplies the data to OpenAQ, users can assess the origins of the data, which is useful for evaluating their reliability and scope. Providers may be governmental agencies, research organizations, or private companies. The data “provider” may be the same as the data “owner” (see Owners). The data “provider” may also be the same as the instrument “manufacturer” (see Manufacturers).

## Key fields

**id:** A unique identifier for each provider.

**name:** The name of the provider.

**sourceName:** The internal name used for the data source.

**exportPrefix:** A label used when exporting data to AWS Open Data bucket.

**datetimeAdded:** The date the provider was added to the OpenAQ platform, provided in UTC and local time. See Dates, times and timezones for more information.

**datetimeFirst:** The datetime of the first measurement available in OpenAQ from the provider, provided in UTC and local time. See Dates, times and timezones for more information.

**datetimeLast:** The datetime of the last measurement available in OpenAQ from the provider. See Dates, times and timezones for more information.

**parameters:** The list of pollutants and environmental parameters available from the provider. See the Parameters resource page for more information.

**bbox:** The geographic bounds (bounding box) for which the provider’s data covers in form X minimum, Y minimum, X maximum, Y maximum.

Example response payload

<https://api.openaq.org/v3/providers/119>

```
{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,
    "limit": 100,
    "found": 1
  },
  "results": [
    {
      "id": 119,
      "name": "AirNow",
      "sourceName": "AirNow",
      "exportPrefix": "airnow",
      "datetimeAdded": "2023-03-29T20:23:57.054584Z",
      "datetimeFirst": "2016-01-30T01:00:00Z",
      "datetimeLast": "2024-09-25T23:00:00Z",
      "entitiesId": 1,
      "parameters": [
        {
          "id": 1,
          "name": "pm10",
          "units": "µg/m³",
          "displayName": null
        },
        {
          "id": 2,
          "name": "pm25",
          "units": "µg/m³",
          "displayName": null
        },
        {
          "id": 7,
          "name": "no2",
          "units": "ppm",
          "displayName": null
        },
        {
          "id": 8,
          "name": "co",
          "units": "ppm",
          "displayName": null
        },
        {
          "id": 9,
```

```
    "name": "so2",
    "units": "ppm",
    "displayName": null
  },
  {
    "id": 10,
    "name": "o3",
    "units": "ppm",
    "displayName": null
  },
  {
    "id": 11,
    "name": "bc",
    "units": "µg/m³",
    "displayName": null
  },
  {
    "id": 35,
    "name": "no",
    "units": "ppm",
    "displayName": null
  },
  {
    "id": 19840,
    "name": "nox",
    "units": "ppm",
    "displayName": null
  }
],
"bbox": {
  "type": "Polygon",
  "coordinates": [
    [
      [
        -161.767,
        -34.5766
      ],
      [
        -161.767,
        70.1319
      ],
      [
        123.424434,
        70.1319
      ],
      [
        123.424434,
```

```

        -34.5766
      ],
      [
        -161.767,
        -34.5766
      ]
    ]
  }
}
]
}

```

## Sensors

The Sensors resource in the OpenAQ API provides details about individual air quality sensors that collect environmental data. One or more sensors belong to a Location. Each sensor tracks a single parameter, such as air pollutant measurements or related measurements (e.g., temperature and humidity) and reports on their concentration over time.

## Purpose and Use

The Sensors resource allows users to gain insights into the specific sensors used for collecting air quality data. By viewing sensor details, users can analyze time coverage and better understand the data's reliability.

## Key Fields

id: a unique identifier for each sensor.

name: The name of the sensor.

parameter: Information about the parameter measured, including the pollutant name , units , and display name. See the Parameters resource page for more information.

datetimeFirst: The datetime of the first measurement available in OpenAQ from the sensor, provided in UTC and local time See Dates, times and timezones for more information.

datetimeLast: The datetime of the last measurement available in OpenAQ from the sensor. See Dates, times and timezones for more information.

coverage: Includes details about data completeness, such as the expected number of measurements, actual counts, and the time period the data spans.

latest: The most recent recorded value and corresponding geographical coordinates (latitude and longitude). See the Latest resource page for more information.

summary: Statistical summary about the values across the lifetime of the sensor on OpenAQ, between datetimeFirst and datetimeLast, including minimum, maximum, average values.

Example response payload

<https://api.openaq.org/v3/sensors/3917>

```

{
  "meta": {
    "name": "openaq-api",
    "website": "/",
    "page": 1,

```

```
"limit": 100,
"found": 1
},
"results": [
{
  "id": 3917,
  "name": "o3 ppm",
  "parameter": {
    "id": 10,
    "name": "o3",
    "units": "ppm",
    "displayName": "O3"
  },
  "datetimeFirst": {
    "utc": "2016-03-06T20:00:00Z",
    "local": "2016-03-06T13:00:00-07:00"
  },
  "datetimeLast": {
    "utc": "2024-09-18T23:00:00Z",
    "local": "2024-09-18T17:00:00-06:00"
  },
  "coverage": {
    "expectedCount": 1,
    "expectedInterval": "01:00:00",
    "observedCount": 58447,
    "observedInterval": "58447:00:00",
    "percentComplete": 5844700.0,
    "percentCoverage": 5844700.0,
    "datetimeFrom": {
      "utc": "2016-03-06T20:00:00Z",
      "local": "2016-03-06T13:00:00-07:00"
    },
    "datetimeTo": {
      "utc": "2024-09-18T23:00:00Z",
      "local": "2024-09-18T17:00:00-06:00"
    }
  },
  "latest": {
    "datetime": {
      "utc": "2024-09-18T23:00:00Z",
      "local": "2024-09-18T17:00:00-06:00"
    },
    "value": 0.055,
    "coordinates": {
      "latitude": 35.1353,
      "longitude": -106.584702
    }
  }
}
```

```

    },
    "summary": {
      "min": 0.0,
      "max": 0.109,
      "avg": 0.03580498661195457
    }
  }
]
}

```

## Errors

The OpenAQ API uses standard HTTP status codes to indicate the outcome of API requests, whether successful or erroneous. A successful request is accompanied by a 200 “OK” status code that confirms the operation was completed as expected. In cases of errors, the API responds with specific status codes that signify different types of issues. These codes enable developers to quickly understand and address any problems that arise when using the API.

HTTP Code	Status text	Description
200	OK	Successful request.
401	Unauthorized	Valid API key is missing.
403	Forbidden	The requested resource may exist but the user is not granted access. This may be a sign that the user account has been blocked for non-compliance of the terms of use.
404	Not Found	The requested resource does not exist.
405	Method Not Allowed	The HTTP method is not supported. The OpenAQ API currently only supports GET requests.
408	Request Timeout	The request timed out, the query may be too complex causing it to run too long.
410	Gone	.
422	Unprocessable Content	The query provided is incorrect and does not follow the standards set by the API specification.
429	Too Many Requests	The number of requests exceeded the rate limit for the given time period.
500, 502, 503, 504	Server errors	Something has failed on the side of OpenAQ services. Contact us on Slack.