

# Candid.

## Issue Lab API guide

Date last modified: 7/15/2021

### Contents

- Introduction
- Request an API Key
- Using the API
- Parameters at a Glance
- Parameters in depth
  - key
  - format
  - issues
  - organization
  - pub\_date and pub\_date\_end
  - add\_date and add\_date\_end
  - modified\_date and modified\_date\_end
  - number and page\_num
  - sort
  - keyword (including multiple keywords)
  - coverage and coverage\_strict
  - pubtype and not\_pubtype
  - ein\_any, ein\_gm, and ein\_org
  - pcs\_subject **NEW**
  - pcs\_population **NEW**
  - issuelab\_id **NEW**

## Introduction

Issue Lab is Candid's knowledge management platform. We provide free access to thousands of case studies, evaluations, white papers, and issue briefs addressing some of the world's most pressing social problems. We also are committed to increasing access to this knowledge by making it broadly available via our distribution network which includes our Application Programming Interface (API) service available at [data.issuelab.org](https://data.issuelab.org).

Issue Lab's API gives you access to our entire collection. You can grab our resources in XML or JSON - your choice, and can use this data for mash-ups and/or as content alongside your own site offerings.

This guide provides descriptions, instructions, and plenty of examples to illustrate how to construct API requests. If while using our API you get stuck or have a question, or need assistance getting started, don't hesitate to get in touch with us at [issuelab@candid.org](mailto:issuelab@candid.org).

## Request an API Key

To get started, you will need your own API key. We tie API keys to domains, so you will need one for every application you want to bring our data into. Don't have a key yet? Visit <https://data.issuelab.org> to request one. Once you have a key, you are ready to go!

## Using the API

The parameters described below let you constrain your request by issue area(s), publishing organization, publication date, date resources were added to our database, number of resources you'd like to receive, and format (XML or JSON). Used together, these parameters enable you to construct intricate data requests. For example, you can run a weekly request for all resources added in the last seven days that were indexed under the issue area "Children and Youth". On the simpler side, you can request everything available with just two parameters -- your API key and your desired output format. Read on to learn more.

## Parameters at a Glance

parameter	required?	notes
key	yes	Need an API key? Visit <a href="https://data.issuelab.org">https://data.issuelab.org</a> to request one.
format	yes	Choose JSON or XML
issues	optional	38 issue areas are available to query
organization	optional	Enables access to all resources available from a publishing organization
pub_date and pub_date_end	optional	Choose works published on a particular date or in a particular date range
add_date add_date_end	optional	Choose works added to Issue Lab on a particular date or in a particular date range
modified_date and modified_date_end	optional	Choose works modified on a particular date or in a particular date range
number and page_num	optional	Request a specific number of resources
sort	optional	Sort returned records in ascending or descending order
keyword	optional	Request records by keyword or phrase
coverage/coverage_strict	optional	Request records by geographic area covered
pubtype/not_pubtype	optional	19 publication types are available to query
ein_any/ein_gm/ein_org	optional	Request records by the EIN of a publishing organization
pcs_population	optional	Requests records based on population group as defined in Candid's Philanthropy Classification System (PCS) (see <a href="https://taxonomy.candid.org/populations">https://taxonomy.candid.org/populations</a> )
pcs_subject	optional	Requests records based on subject as defined in Candid's Philanthropy Classification System (PCS) (see <a href="https://taxonomy.candid.org/subjects">https://taxonomy.candid.org/subjects</a> )

## Parameters in Depth

Here is everything you need to know about the parameters available to include in your requests.

### Parameter: key

#### **Required**

You need an API key to access our resources. Visit <https://data.issuelab.org> to request one.

Example request:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml](https://data.issuelab.org/fetch/key/your_key/format/xml)

### Parameter: format

#### **Required**

You can retrieve data in either XML or JSON format. Note that either of the example requests below, which include only the “key” parameter and “format” parameter, are all you need to make a request for data. In other words, if you copy/paste one of these two examples into your browser, and replace your key with your actual key, data will be returned to you.

Example request:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml](https://data.issuelab.org/fetch/key/your_key/format/xml)

Example request:

[https://data.issuelab.org/fetch/key/your\\_key/format/json](https://data.issuelab.org/fetch/key/your_key/format/json)

### Parameter: issues

#### **Optional**

Accepts one or multiple issue areas.

#### **Single issue request**

To request resources from a single issue area in XML format, simply include the issue area identifier in your request like so:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/issues/Education and Literacy](https://data.issuelab.org/fetch/key/your_key/format/xml/issues/Education%20and%20Literacy)

## Multiple issue request

To limit results to resources indexed under a specific combination of issue areas, format your request like so: `+[issue1|issue2|issue3]`. Here, a resource must be indexed under 'issue1' *and* 'issue2' *and* 'issue3' to make it into your result set.

Example - "Fetch all resources in XML format indexed under Aging *and* Health *and* Women."

<https://data.issuelab.org/fetch/key/yourkey/format/xml/issues/>**+**[Aging|Health|Women]****

To limit results to resources indexed under any or all specified issue areas, format your request like so: - `[issue1|issue2|issue3]`. Here, any resource indexed under 'issue1' or 'issue2' or 'issue3' will make it into your result set.

Example - "Fetch all resources in XML format indexed under Aging *or* Health *or* Women."

<https://data.issuelab.org/fetch/key/yourkey/format/xml/issues/>**-**[Aging|Health|Women]****

## Available issue areas

Aging	Humanitarian and Disaster Relief
Agriculture and Food	Hunger
Animal Welfare	Immigration
Arts and Culture	International Development
Athletics and Sports	Journalism and Media
Children and Youth	LGBTQI
Civil Society	Men
Community and Economic Development	Nonprofits and Philanthropy
Computers and Technology	Parenting and Families
Consumer Protection	Peace and Conflict
Crime and Safety	Poverty
Disabilities	Prison and Judicial Reform
Education and Literacy	Race and Ethnicity
Employment and Labor	Religion
Energy and Environment	Science
Government Reform	Substance Abuse and Recovery
Health	Transportation
Housing and Homelessness	Welfare and Public Assistance
Human Rights and Civil Liberties	Women

## Parameter: organization

### ***Optional***

Accepts one organization name. If you do not include the organization parameter in your request you will receive all resources available regardless of which organization published the resources.

*Tip:* search for your organization's name on Issue Lab to obtain the exact name we have in our database.

To request resources in JSON format from a single publishing organization, simply include the organization name in your request:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/organization/Foundation Center](https://data.issuelab.org/fetch/key/your_key/format/json/organization/Foundation%20Center)

## Parameter: pub\_date and pub\_date\_end

### **Optional**

Use the pub\_date and pub\_date\_end parameters together to request resources published on an exact date or in a particular date range.

The pub\_date and pub\_date\_end parameters accept a date formatted like so: YYYY-MM-DD. Be sure to include two digits for months and days -- January through September require a leading zero (01 for January, 02 for February...). Days 1-9 also require a leading zero (01, 02...).

### **Examples**

“Fetch resources in XML format published in 2012.”

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/pub\\_date/2012-01-01/pub\\_date\\_end/2012-12-31](https://data.issuelab.org/fetch/key/your_key/format/xml/pub_date/2012-01-01/pub_date_end/2012-12-31)

“Fetch resources in JSON format published in June 2011.”

[https://data.issuelab.org/fetch/key/your\\_key/format/json/pub\\_date/2011-06-01/pub\\_date\\_end/2011-06-30](https://data.issuelab.org/fetch/key/your_key/format/json/pub_date/2011-06-01/pub_date_end/2011-06-30)

“Fetch resources in XML format published on June 5, 2010.”

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/pub\\_date/2010-06-05/pub\\_date\\_end/2010-06-05](https://data.issuelab.org/fetch/key/your_key/format/xml/pub_date/2010-06-05/pub_date_end/2010-06-05)

## Parameter: add\_date and add\_date\_end

### **Optional**

Use the add\_date and add\_date\_end parameters together to request resources added to Issue Lab on an exact date or in a particular date range.

The add\_date and add\_date\_end parameters accept a date formatted like so: YYYY-MM-DD. Be sure to include two digits for months and days -- January through September require a leading zero (01 for January, 02 for February...). Days 1-9 also require a leading zero.

### **Examples**

“Fetch resources in XML format added to Issue Lab in 2012”

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/add\\_date/2012-01-01/add\\_date\\_end/2012-12-31](https://data.issuelab.org/fetch/key/your_key/format/xml/add_date/2012-01-01/add_date_end/2012-12-31)

“Fetch resources in JSON format added to Issue Lab in June 2011”

[https://data.issuelab.org/fetch/key/your\\_key/format/jsonl/add\\_date/2011-06-01/add\\_date\\_end/2011-06-30](https://data.issuelab.org/fetch/key/your_key/format/jsonl/add_date/2011-06-01/add_date_end/2011-06-30)

“Fetch resources in XML format added to Issue Lab on June 15, 2010”

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/add\\_date/2010-06-15/add\\_date\\_end//2010-06-15](https://data.issuelab.org/fetch/key/your_key/format/xml/add_date/2010-06-15/add_date_end//2010-06-15)

## Parameter: `modified_date` and `modified_date_end`

### *Optional*

Use the `modified_date` and `modified_date_end` parameters together to request resources last modified on an exact date or in a particular date range.

The `modified_date` and `modified_date_end` parameters accept a date formatted like so: YYYY-MM-DD. Be sure to include two digits for months and days -- January through September require a leading zero (01 for January, 02 for February...). Days 1-9 also require a leading zero.

### **Examples**

“Fetch resources in XML format modified in 2012”

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/modified\\_date/2012-01-01/modified\\_date\\_end/2012-12-31](https://data.issuelab.org/fetch/key/your_key/format/xml/modified_date/2012-01-01/modified_date_end/2012-12-31)

“Fetch resources in JSON format modified in June 2011”

[https://data.issuelab.org/fetch/key/your\\_key/format/jsonl/modified\\_date/2011-06-01/modified\\_date\\_end/2011-06-30](https://data.issuelab.org/fetch/key/your_key/format/jsonl/modified_date/2011-06-01/modified_date_end/2011-06-30)

“Fetch resources in XML format modified on June 15, 2010”

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/modified\\_date/2010-06-15/modified\\_date\\_end//2010-06-15](https://data.issuelab.org/fetch/key/your_key/format/xml/modified_date/2010-06-15/modified_date_end//2010-06-15)

## Parameter: number and page\_num

### **Optional**

By default, the API returns up to 1,000 results. To retrieve more than 1,000 results use a combination of the “number” and “page\_num” parameters.

The “number” parameter limits the number of results you receive. Accepts a number (eg., 1 or 23 or 900). This parameter can be used without also including the “page\_num” parameter. The maximum request accepted is 1,000.

The “page\_num” parameter can be used to paginate results and is the only way that you can retrieve more than 1,000 results. Use the “page\_num” and “number” parameters together to create the equivalent of an offset in MySQL. The “page\_num” parameter must be used in conjunction with the “number” parameter.

Remember to use only digits and do not include punctuation (eg., a comma) when using either parameter.

### **Examples**

Fetch 3 records in XML format:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/number/3](https://data.issuelab.org/fetch/key/your_key/format/xml/number/3)

Fetch 2,000 records in XML format added to Issue Lab since September 1, 2012

Call 1: [https://data.issuelab.org/fetch/key/your\\_key/format/xml/add\\_date/2012-09-01/page\\_num/1/number/1000](https://data.issuelab.org/fetch/key/your_key/format/xml/add_date/2012-09-01/page_num/1/number/1000)

Call 2: [https://data.issuelab.org/fetch/key/your\\_key/format/xml/add\\_date/2012-09-01/page\\_num/2/number/1000](https://data.issuelab.org/fetch/key/your_key/format/xml/add_date/2012-09-01/page_num/2/number/1000)

## Parameter: sort

### **Optional**

The “sort” parameter lets you sort your result set in ascending (sort\_asc) or descending (sort\_desc) order based on the publication date (pub\_date), date added to Issue Lab (add\_date), or title (title).

Note: Sorting in ascending order will provide results sorted by date with the earliest date first, and results sorted by title in A-to-Z order. Sorting in descending order will provide results sorted by date with the most recent date first, and results sorted by title in Z-to-A order.

### **Examples**

Fetch records in JSON format sorted by title in ascending order:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/sort\\_asc/title](https://data.issuelab.org/fetch/key/your_key/format/json/sort_asc/title)

Fetch records in XML format sorted by publication date in descending order:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/sort\\_desc/pub\\_date](https://data.issuelab.org/fetch/key/your_key/format/xml/sort_desc/pub_date)

Fetch records in XML format sorted by date added to Issue Lab in ascending order:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/sort\\_asc/add\\_date](https://data.issuelab.org/fetch/key/your_key/format/xml/sort_asc/add_date)

## Parameter: keyword

### *Optional*

The “keyword” parameter lets you request results based on a keyword or phrase, or on multiple keywords and phrases. Quotations around a phrase are not necessary -- whatever you supply will be treated as an exact-phrase to match against.

### **Examples**

Fetch all records in JSON format that include the keyword “teenagers”:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/keyword/teenagers](https://data.issuelab.org/fetch/key/your_key/format/json/keyword/teenagers)

Fetch records in XML format that include the phrases “charter schools” and “small schools”:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/keyword/charter schools AND small schools](https://data.issuelab.org/fetch/key/your_key/format/xml/keyword/charter schools AND small schools)

Fetch records in XML format that include the keywords/phrases “renewable energy” or “alternative energy” or “photovoltaics” and are categorized under “Community and Economic Development”:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/keyword/renewable energy OR alternative energy OR photovoltaics/issues/Community and Economic Development](https://data.issuelab.org/fetch/key/your_key/format/xml/keyword/renewable energy OR alternative energy OR photovoltaics/issues/Community and Economic Development)

Fetch records in JSON format that include the phrase “restorative justice” and are categorized under “Nonprofits and Philanthropy”:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/keyword/restorative justice/issues/Nonprofits and Philanthropy](https://data.issuelab.org/fetch/key/your_key/format/json/keyword/restorative justice/issues/Nonprofits and Philanthropy)

## Parameter: coverage and coverage\_strict

### *Optional*

The “coverage” and “coverage\_strict” parameters let you request results based on a geographic area. Either parameter accepts one geographic area per query.

**Important:** you must use the pipe - | - character to separate the parts of your geographic location. For example, to query on the country of Mexico use: North America | Mexico. The geographic metadata returned will be formatted with a forward slash in place of the pipe, eg., North America / Mexico.

Use “coverage\_strict” to request resources indexed under a specific geographic area. For example, if you want all resources indexed under the geographic location California -- meaning the resources returned in your result set are indexed at the state-level:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/coverage\\_strict/North America | United States \(Western\) | California](https://data.issuelab.org/fetch/key/your_key/format/json/coverage_strict/North America | United States (Western) | California)

The result set returned with this query will include resources indexed with the exact term “North America-United States (Western) California”.

Continuing with this example, if you would like to broaden the result set for California so that you receive not only resources indexed on the state-level, but also resources indexed on the county, city, and neighborhood levels, use “coverage”:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/coverage/North America | United States \(Western\) | California](https://data.issuelab.org/fetch/key/your_key/format/json/coverage/North America | United States (Western) | California)

The result set returned with this query will include resources indexed under North America-United States (Western) California and also, for example,

- North America / United States (Western) / California (San Fernando Valley)
- North America / United States (Western) / California / Alameda County
- North America / United States (Western) / California-Alameda County / Oakland (San Antonio)

## Examples

Fetch all records in JSON format that are indexed under “Africa” (returning continent-level):

[https://data.issuelab.org/fetch/key/your\\_key/format/json/coverage\\_strict/Africa](https://data.issuelab.org/fetch/key/your_key/format/json/coverage_strict/Africa)

Fetch all records in JSON format that are indexed under “Africa” including the continent, regions, countries, cities:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/coverage/Africa](https://data.issuelab.org/fetch/key/your_key/format/json/coverage/Africa)

## Parameter: subtype and not\_subtype

### **Optional**

The subtype and not\_subtype parameters let you request results based on inclusion or exclusion of a publication type or types. You can request multiple subtypes in either parameter. You can also use subtype and not\_subtype in the same query.

### **Examples**

Fetch all records in JSON format that are not indexed under the “Annual Report” subtype:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/not\\_subtype/Annual Report](https://data.issuelab.org/fetch/key/your_key/format/json/not_subtype/Annual%20Report)

Fetch records in XML format that are indexed under the “Report/Whitepaper” and “Dataset” subtypes:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/subtype/Whitepaper AND Dataset](https://data.issuelab.org/fetch/key/your_key/format/xml/subtype/Whitepaper%20AND%20Dataset)

Fetch records in XML format that are indexed under the “Report/Whitepaper” or “Issue/Policy Brief” or “CaseStudy” subtypes, but not indexed under the “Annual Report” subtype:

[https://data.issuelab.org/fetch/key/your\\_key/format/xml/not\\_subtype/Annual Report/subtype/Whitepaper OR Policy Brief OR Casestudy](https://data.issuelab.org/fetch/key/your_key/format/xml/not_subtype/Annual%20Report/subtype/Whitepaper%20OR%20Policy%20Brief%20OR%20Casestudy)

### **Available subtypes**

Annual Report	Interactive Resource
Article	Policy Brief ( <i>for Issue/Policy Brief</i> )
Audio	Legislative Document
Book	Research Review ( <i>for Literature/Research Review</i> )
CaseStudy	MovingImage
Conference Report	Presentation ( <i>for Presentation/Slides</i> )
Dataset	Whitepaper ( <i>for Report/Whitepaper</i> )
Evaluation	Survey
FactSheet	Toolkit
Infographic ( <i>for Infographic/StillImage</i> )	

## Parameter: ein\_any, ein\_gm, and ein\_org

### ***Optional***

You can query the API by employee identification number (EIN) which will let you request results based on the EIN number of a publishing organization (the organization that created and owns a resource) or a grantmaker (the organization that funded creation of a resource). There are three parameters available:

- ein\_any: use this parameter to request results regardless of whether the EIN you are interested in belongs to a publishing organization or a grantmaker.
- ein\_gm: use this parameter to request results that were funded by an organization that the EIN you are interested in belongs to.
- ein\_org: use this parameter to request results that were published by the organization that the EIN you are interested in belongs to.

Note: You cannot look up EINs through Issue Lab’s API. You need to have the EIN you’d like to query on available before attempting to use this parameter.

### **Examples**

“Fetch everything available in XML format that was funded by a grantmaker with EIN # 123456789.”

[https://data.issuelab.org/fetch/key/XXX/format/xml/ein\\_gm/123456789](https://data.issuelab.org/fetch/key/XXX/format/xml/ein_gm/123456789)

“Fetch everything available in XML format that was published by an organization with EIN # 123456789.”

[https://data.issuelab.org/fetch/key/XXX/format/xml/ein\\_org/123456789](https://data.issuelab.org/fetch/key/XXX/format/xml/ein_org/123456789)

“Fetch everything available in XML format that was either funded by a grantmaker with EIN # 123456789 or published by an organization with EIN # 123456789.”

[https://data.issuelab.org/fetch/key/XXX/format/xml/ein\\_any/123456789](https://data.issuelab.org/fetch/key/XXX/format/xml/ein_any/123456789)

## Parameter: pcs\_subject

### ***Optional***

The “pcs\_subject” parameter lets you request results based on a subject defined in Candid’s Philanthropy Classification System (PCS). PCS subjects describe what is being supported – grantmaker areas of focus, primary work of nonprofits and activities being funded. The complete list of subject codes is available at <https://taxonomy.candid.org/subjects>.

This parameter accepts one or multiple subject codes.

### **Single subject code request**

To request resources from a single subject code in JSON format, include the subject code in your request like so:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/pcs\\_subject/SA060100](https://data.issuelab.org/fetch/key/your_key/format/json/pcs_subject/SA060100)

This request would return all resources that are coded to the Arts & Culture / Performing Arts / Dance PCS subject.

### Multiple subject code request

To limit results to resources indexed under a specific combination of PCS subjects, format your request like so: `+{subject1|subject2|subject3}`. Here, a resource must be indexed under ‘subject1’ *and* ‘subject2’ *and* ‘subject3’ to make it into your result set.

Example - “Fetch all resources indexed under Dance (SA060100) *and* Music (SA060300) *and* Theater (SA060200).”

[https://data.issuelab.org/fetch/key/your\\_key/format/json/pcs\\_subject/+\[SA060100|SA060300|SA060200\]](https://data.issuelab.org/fetch/key/your_key/format/json/pcs_subject/+[SA060100|SA060300|SA060200])

To limit results to resources indexed under any or all specified subjects, format your request like so: `-{subject1|subject2|subject3}`. Here, any resource indexed under ‘subject1’ *or* ‘subject2’ *or* ‘subject3’ will make it into your result set.

Example - “Fetch all resources indexed under Dance (SA060100) *or* Music (SA060300) *or* Theater (SA060200).”

[https://data.issuelab.org/fetch/key/your\\_key/format/json/pcs\\_subject/-\[SA060100|SA060300|SA060200\]](https://data.issuelab.org/fetch/key/your_key/format/json/pcs_subject/-[SA060100|SA060300|SA060200])

### Wildcards

Top level subject codes can be treated as wildcards.

Example - “Fetch all resource coded to the “Arts and Culture” subject code:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/pcs\\_subject/SA](https://data.issuelab.org/fetch/key/your_key/format/json/pcs_subject/SA)

## Parameter: pcs\_population

### Optional

The “pcs\_population” parameter lets you request results based on a population defined in Candid’s Philanthropy Classification System (PCS). PCS populations describe who is being supported - details characteristics (age, sex, race/ethnic heritage, etc.) of the population(s) being targeted as the intended beneficiary of support and services being provided. The complete list of population codes is available at <https://taxonomy.candid.org/populations>.

This parameter accepts one or multiple population codes.

## Single population code request

To request resources from a single population code include the population code in your request like so:

<https://data.issuelab.org/fetch/key/your> key/format/json/**pcs\_population/PJ050000**

This request would return all resources that are coded to the Work status and occupations / Academics PCS population.

## Multiple subject code request

To limit results to resources indexed under a specific combination of PCS populations, format your request like so: +[population1|population2|population3]. Here, a resource must be indexed under ‘population1’ *and* ‘population2’ *and* ‘population3’ to make it into your result set.

Example - “Fetch all resources indexed under Academics (PJ050000) *and* Activists (PJ120000) *and* Retired people (PJ040000).”

<https://data.issuelab.org/fetch/key/your> key/format/json/**pcs\_population/+[PJ050000|PJ120000|PJ040000]**

To limit results to resources indexed under any or all specified populations, format your request like so: -[population1|population2|population3]. Here, any resource indexed under ‘population1’ *or* ‘population2’ *or* ‘population3’ will make it into your result set.

Example - “Fetch all resources indexed under Academics (PJ050000) *or* Activists (PJ120000) *or* Retired people (PJ040000).”

<https://data.issuelab.org/fetch/key/your> key/format/json/**pcs\_population/-[PJ050000|PJ120000|PJ040000]**

## Wildcards

Top level population codes can be treated as wildcards.

Example - “Fetch all resource coded to the “Work status and occupations” population code:

<https://data.issuelab.org/fetch/key/your> key/format/json/**pcs\_population/PJ**

## Parameter: `issuelab_id`

### *Optional*

The `issuelab_id` parameter lets you request results based on the unique ID of a particular resource in Issue Lab. You can request one or multiple unique IDs. To request multiple IDs separate ID numbers with a pipe (`|`). Multiple IDs are processed as an “OR” request.

### **Examples**

Fetch a record in JSON format that has unique ID # 35008:

<https://data.issuelab.org/fetch/key/your> key/format/json/**issuelab\_id/35008**

Fetch records in JSON format that have unique ID # 35008 OR unique ID # 35009:

[https://data.issuelab.org/fetch/key/your\\_key/format/json/issuelab\\_id/35008|35009](https://data.issuelab.org/fetch/key/your_key/format/json/issuelab_id/35008|35009)