

Items

GEThttp://hn.algolia.com/api/v1/items/:id

```
{
  • id: 1,
  • created_at: "2006-10-09T18:21:51.000Z",
  • author: "pg",
  • title: "Y Combinator",
  • url: "http://ycombinator.com",
  • text: null,
  • points: 57,
  • parent_id: null,
  • children:
    [
      • {
        ▪ id: 15,
        ▪ created_at: "2006-10-09T19:51:01.000Z",
        ▪ author: "sama",
        ▪ text: "&#34;the rising star of venture capital&#34; -unknown VC eating lunch on SHR",
        ▪ points: 5,
        ▪ parent_id: 1,
        ▪ children:
          [
            ▪ {
              ▪ id: 17,
              ▪ created_at: "2006-10-09T19:52:45.000Z",
              ▪ author: "pg",
              ▪ text: "Is there anywhere to eat on Sandhill Road?",
              ▪ points: 5,
              ▪ parent_id: 15,
              ▪ children: [ ]
            }
          ]
        }
      ]
    ]
}
```

https://hn.algolia.com/api

https://hn.algolia.com/api

Users

GEThttp://hn.algolia.com/api/v1/users/:username

```
{
  • username: "pg",
  • about: "PG's bio",
  • karma: 99999
}
```

Search

Sorted by relevance, then points, then number of comments

GEThttp://hn.algolia.com/api/v1/search?query=...

Sorted by date, more recent first

GEThttp://hn.algolia.com/api/v1/search_by_date?query=...

Common query parameters:

Parameter	Description	Type
query=	full-text query filter on a specific tag. Available tags:	String
tags=	<ul style="list-style-type: none">• story• comment• poll• pollopt• show_hn• ask_hn• front_page• author_:USERNAME• story_:ID	String
numericFilters=	filter on a specific numerical condition (<, <=, =, > or >=). String Available numerical fields:	
	<ul style="list-style-type: none">• created_at_i	

- points
- num_comments

page= page number Integer

Tags are ANDed by default, can be ORED if between parenthesis. For example `author_pg,(story,poll)` filters on `author=pg AND (type=story OR type=poll)`.

By default a limited number of results are returned in each page, so a given query may be broken over dozens of pages. The number of results and page number are available as the variables `nbPages` and `hitsPerPage` respectively; they can be specified as arguments in requests, allowing for more results to be requested or iteration over the available pages eg appending to the search URL parameters like `&page=2` or `hitsPerPage=50`.

The complete list of search parameters is available in Algolia [Search API documentation](#).

Examples

All stories matching foo

`http://hn.algolia.com/api/v1/search?query=foo&tags=story`

All comments matching bar

`http://hn.algolia.com/api/v1/search?query=bar&tags=comment`

All URLs matching bar

`http://hn.algolia.com/api/v1/search?query=bar&restrictSearchableAttributes=url`

All stories that are on the front/home page right now

`http://hn.algolia.com/api/v1/search?tags=front_page`

Last stories

`http://hn.algolia.com/api/v1/search_by_date?tags=story`

Last stories OR polls

`http://hn.algolia.com/api/v1/search_by_date?tags=(story,poll)`

Comments since timestamp X (in second)

`http://hn.algolia.com/api/v1/search_by_date?tags=comment&numericFilters=created_at_i>X`

Stories between timestamp X and timestamp Y (in second)

`http://hn.algolia.com/api/v1/search_by_date?tags=story&numericFilters=created_at_i>X,created_at_i<Y`

Stories of pg

`http://hn.algolia.com/api/v1/search?tags=story,author_pg`

Comments of story X

`http://hn.algolia.com/api/v1/search?tags=comment,story_X`

```
{
  hits:
  [
    {
      title: "Y Combinator",
      url: "http://ycombinator.com",
      author: "pg",
      points: 57,
      story_text: null,
      comment_text: null,
      tags:
      [
        "story"
      ],
      num_comments: 2,
      objectID: "1",
      highlightResult:
      {
        title:
        {
          value: "Y Combinator",
          matchLevel: "none",
          matchedWords: [ ]
        },
        url:
        {
          value: "http://ycombinator.com",
          matchLevel: "none",
          matchedWords: [ ]
        },
        author:
        {
          value: "<em>pg</em>",
          matchLevel: "full",
          matchedWords:
          [
            "pg"
          ]
        }
      }
    },
    ...
  ],
  page: 0,
  nbHits: 11,
  nbPages: 1,
  hitsPerPage: 20,
  processingTimeMS: 1,
  query: "pg",
  params: "query=pg"
```

```
}
```

Rate limits

We are limiting the number of API requests from a single IP to 10,000 per hour. If you or your application has been blacklisted and you think there has been an error, please [contact us](#).