StackExchange

Navigation

## Stack Exchange API v2.3

This is the documentation for the v2.3 Stack Exchange API (with both authentication and write support). If you have additional questions, or believe you have encountered a bug, don't hesitate to post a question on Stack Apps.

If your application is in a runnable state (even a beta one), Stack Apps is also the place to list it.

## General

Applications should be registered on Stack Apps to get a request key. Request keys grant more requests per day, and are necessary for using access_tokens created via authentication.

All API responses are JSON, we do support JSONP with the `callback` query parameter. Every response in the API is returned in a common "wrapper" object, for easier and more consistent parsing.

Additionally, all API responses are compressed. The `Content-Encoding` header is always set, but some proxies will strip this out. The proper way to decode API responses can be found here.

API usage is subject to a number of throttles. In general, applications should strive to make as few requests as possible to satisfy their function.

API responses are heavily cached. Polling for changes should be done sparingly in any case, and polling at a rate faster than once a minute (for semantically identical requests) is considered abusive.

Developers can trim API responses down to just the fields they are interested in using custom filters. Many types have fields that are not normally returned (question bodies, for example) that can likewise be requested via a custom filter.

There are a few methods which require that the application be acting on behalf of a user in order to be invoked. For authentication purposes, the Stack Exchange API implements OAuth 2.0 (templated on Facebook's implementation in pursuit of developer familiarity).

A number of methods in the Stack Exchange API accept dates as parameters and return dates as properties, the format of these dates is consistent and documented. The cliff-notes version is, all dates in the Stack Exchange API are in unix epoch time.

Unless otherwise noted, the maximum size of any page is `100`, any `{ids}` parameter likewise is capped at `100` elements, all indexes start at `1`.

If a parameter name is plural it accepts vectorized requests, otherwise a single value may be passed.

Compare users/{id}/inbox and /users/{ids}.

It is possible to compose reasonably complex queries against the live Stack Exchange sites using the `min`, `max`, `fromdate`, `todate`, and `sort` parameters. Most, but not all, methods accept some or all of these parameters, the documentation for individual methods will highlight which do. Most methods also have a common set of paging parameters.

Some methods require access_tokens with particular scopes, such as private_info (/users/{id}/reputation-history/full for example) or write_access (/questions/add). Certain fields require access_tokens with the private_info scope, such as answer.upvoted; the documentation for each time makes note of these fields.

by category by type

## Per-Site Methods

Each of these methods operates on a single site at a time, identified by the `site` parameter. This parameter can be the full domain name (ie. "stackoverflow.com"), or a short form identified by `api_site_parameter` on the site object.

### Answers

answers
Get all answers on the site.

answers/{ids}
Get answers identified by a set of ids.

answers/{id}/accept
Casts an accept vote on the given answer. [auth required]

answers/{id}/accept/undo
Undoes an accept vote on the given answer. [auth required]

answers/{ids}/comments
Get comments on the answers identified by a set of ids.

answers/{id}/delete
Deletes the given answer. [auth required]

answers/{id}/downvote
Casts a downvote on the given answer. [auth required]

answers/{id}/downvote/undo
Undoes a downvote on the given answer. [auth required]

answers/{id}/edit

Edits the given answer. [auth required]

answers/{id}/flags/options
Returns valid flag options for the given answer. [auth required]

answers/{id}/flags/add
Casts a flag on the given answer. [auth required]

answers/{ids}/questions
Gets all questions the answers identified by ids are on.

answers/{id}/upvote
Casts an upvote on the given answer. [auth required]

answers/{id}/upvote/undo
Undoes an upvote on the given answer. [auth required]

answers/{id}/recommend
Casts a recommendation on the given answer. [auth required]

answers/{id}/recommend/undo
Undoes an recommendation on the given answer. [auth required]

answers/{id}/suggested-edit/add
Creates a suggested edit on an existing answer. [auth required]

### Articles

Add/Edit/Delete Only available for Business Teams via the Teams API.

articles
Get all articles on the site.

articles/{ids}
Get the articles identified by a set of ids.

articles/{ids}/linked
Get the questions that are linked to the articles identified by a set of ids.

articles/{id}/delete
Deletes the given article. [auth required]

articles/{id}/edit
Edits the given article. [auth required]

articles/add
Creates a new article. [auth required]

## Badges

badges
Get all badges on the site, in alphabetical order.

badges/{ids}
Get the badges identified by ids.

badges/name
Get all non-tagged-based badges in alphabetical order.

badges/recipients
Get badges recently awarded on the site.

badges/{ids}/recipients
Get the recent recipients of the given badges.

badges/tags
Get all tagged-based badges in alphabetical order.

## Collectives

collectives
Get all Collectives on the site, in alphabetical order.

collectives/{slugs}
Get Collectives identified by a set of slugs.

collectives/{slugs}/questions
Get questions identified by a set of slugs.

collectives/{slugs}/answers
Get answers identified by a set of slugs.

collectives/{slugs}/tags
Get tags identified by a set of slugs.

collectives/{slugs}/users
Get users identified by a set of slugs.

collectives/{slugs}/reports
Get reports identified by a set of slugs. [auth required]

collectives/{slugs}/reports/add
Creates a new report request for the Collective [auth required]

## Comments

comments
Get all comments on the site.

comments/{ids}
Get comments identified by a set of ids.

comments/{id}/delete
Delete a comment identified by its id. [auth required]

comments/{id}/edit
Edit a comment identified by its id. [auth required]

comments/{id}/flags/add
Casts a flag on the given comment. [auth required]

comments/{id}/flags/options
Returns valid flag options for the given comment. [auth required]

comments/{id}/upvote
Casts an upvote on the given comment. [auth required]

comments/{id}/upvote/undo
Undoes an upvote on the given comment. [auth required]

## Events

events
Get recent events that have occurred on the site. Effectively a stream of new users and content. [auth required]

## Info

info
Get information about the entire site.

## Posts

posts
Get all posts (questions and answers) in the system.

posts/{ids}
Get all posts identified by a set of ids. Useful for when the type of post (question or answer) is not known.

posts/{ids}/comments
Get comments on the posts (question or answer) identified by a set of ids.

posts/{id}/comments/add

Create a new comment on the post identified by id. [auth required]

posts/{id}/comments/render
Renders a hypothetical comment on the given post.

posts/{ids}/revisions
Get revisions on the set of posts in ids.

posts/{ids}/suggested-edits
Get suggested edits on the set of posts in ids.

## Privileges

privileges
Get all the privileges available on the site.

## Questions

questions
Get all questions on the site.

questions/{ids}
Get the questions identified by a set of ids.

questions/{ids}/answers
Get the answers to the questions identified by a set of ids.

questions/{id}/answers/add
Creates an answer on the given question. [auth required]

questions/{id}/answers/render
Renders a hypothetical answer to a question.

questions/{id}/close/options
Returns valid flag options which are also close reasons for the given question. [auth required]

questions/{ids}/comments
Get the comments on the questions identified by a set of ids.

questions/{id}/delete
Deletes the given question. [auth required]

questions/{id}/downvote
Casts a downvote on the given question. [auth required]

questions/{id}/downvote/undo
Undoes a downvote on the given question. [auth required]

questions/{id}/edit

Edits the given question. [auth required]

questions/{id}/favorite

Bookmarks the given question. (Previously known as "favoriting" a question) [auth required]

questions/{id}/favorite/undo

Undoes bookmarking the given question. (Previously known as "favoriting" a question) [auth required]

questions/{id}/flags/add

Casts a flag on the given question. [auth required]

questions/{id}/flags/options

Returns valid flag options for the given question. [auth required]

questions/{ids}/linked

Get the questions that link to the questions identified by a set of ids.

questions/{ids}/related

Get the questions that are related to the questions identified by a set of ids.

questions/{id}/suggested-edit/add

Creates a suggested_edit on an existing question. [auth required]

questions/{ids}/timeline

Get the timelines of the questions identified by a set of ids.

questions/{id}/upvote

Casts an upvote on the given question. [auth required]

questions/{id}/upvote/undo

Undoes an upvote on the given question. [auth required]

questions/add

Creates a new question. [auth required]

questions/featured

Get all questions on the site with active bounties.

questions/no-answers

Get all questions on the site with **no** answers.

questions/render

Renders a hypothetical question. [auth required]

questions/unanswered
Get all questions the site considers unanswered.

questions/unanswered/my-tags
Get questions the site considers unanswered within a user's favorite or interesting tags. [auth required]

### Revisions

revisions/{ids}
Get all revisions identified by a set of ids.

### Search

search
Search the site for questions meeting certain criteria.

search/advanced
Search the site for questions using most of the on-site search options.

similar
Search the site based on similarity to a title.

search/excerpts
Searches a site.

### Suggested Edits

suggested-edits
Get all the suggested edits on the site.

suggested-edits/{ids}
Get the suggested edits identified by a set of ids.

### Tags

tags
Get the tags on the site.

tags/{tags}/info
Get tags on the site by their names.

tags/moderator-only
Get the tags on the site that only moderators can use.

tags/required
Get the tags on the site that fulfill required tag constraints.

tags/synonyms

tags/synonyms

Get all the tag synonyms on the site.

tags/{tags}/faq

Get frequently asked questions in a set of tags.

tags/{tags}/related

Get related tags, based on common tag pairings.

tags/{tags}/synonyms

Get the synonyms for a specific set of tags.

tags/{tag}/top-answerers/{period}

Get the top answer posters in a specific tag, either in the last month or for all time.

tags/{tag}/top-askers/{period}

Get the top question askers in a specific tag, either in the last month or for all time.

tags/{tags}/wikis

Get the wiki entries for a set of tags.

### Users

All user methods that take an `{ids}` parameter have a `/me` equivalent method that takes an `access_token` instead. These methods are provided for developer convenience, with the exception of plain /me, which is actually necessary for discovering which user authenticated to an application.

users

Get all users on the site.

users/{ids}

/me

Get the users identified by a set of ids.

users/{ids}/answers

/me/answers

Get the answers posted by the users identified by a set of ids.

users/{ids}/badges

/me/badges

Get the badges earned by the users identified by a set of ids.

users/{ids}/comments

/me/comments

Get the comments posted by the users identified by a set of ids.

users/{ids}/comments/{toid}

/me/comments/{toid}

Get the comments posted by a set of users in reply to another user.

---

users/{ids}/favorites

/me/favorites

Get the questions bookmarked (previously known as "favorited") by users identified by a set of ids.

---

users/{ids}/mentioned

/me/mentioned

Get the comments that mention one of the users identified by a set of ids.

---

users/{id}/network-activity

/me/network-activity

Gets a user's activity across the Stack Exchange network.

---

users/{id}/notifications

/me/notifications

Get a user's notifications.

---

users/{id}/notifications/unread

/me/notifications/unread

Get a user's unread notifications.

---

users/{ids}/posts

/me/posts

Get all posts (questions and answers) owned by a set of users.

---

users/{id}/privileges

/me/privileges

Get the privileges the given user has on the site.

---

users/{ids}/questions

/me/questions

Get the questions asked by the users identified by a set of ids.

---

users/{ids}/questions/featured

/me/questions/featured

Get the questions on which a set of users, have active bounties.

---

users/{ids}/questions/no-answers

/me/questions/no-answers

Get the questions asked by a set of users, which have **no** answers.

---

users/{ids}/questions/unaccepted

/me/questions/unaccepted

Get the questions asked by a set of users, which have at least one answer but no accepted answer.

---

users/{ids}/questions/unanswered

/me/questions/unanswered

Get the questions asked by a set of users, which are not considered to be adequately answered.

---

users/{ids}/reputation

/me/reputation

Get a subset of the reputation changes experienced by the users identified by a set of ids.

---

users/{ids}/reputation-history

/me/reputation-history

Get a history of a user's reputation, excluding private events.

---

users/{id}/reputation-history/full

/me/reputation-history/full

Get a full history of a user's reputation. [auth required]

---

users/{ids}/suggested-edits

/me/suggested-edits

Get the suggested edits provided by users identified by a set of ids.

---

users/{ids}/tags

/me/tags

Get the tags that the users (identified by a set of ids) have been active in.

---

users/{id}/tags/{tags}/top-answers

/me/tags/{tags}/top-answers

Get the top answers a user has posted on questions with a set of tags.

---

users/{id}/tags/{tags}/top-questions

/me/tags/{tags}/top-questions

Get the top questions a user has posted with a set of tags.

---

users/{id}/tag-preferences

/me/tag-preferences

Get a given user's tag preferences. [auth required]

---

users/{id}/tag-preferences/edit

/me/tag-preferences/edit

Edit a user's tag preferences. [auth required]

---

users/{ids}/timeline

/me/timeline

Get a subset of the actions of that have been taken by the users identified by a set of ids.

---

users/{id}/top-answer-tags

/me/top-answer-tags

Get the top tags (by score) a single user has posted answers in.

---

users/{id}/top-question-tags

/me/top-question-tags

Get the top tags (by score) a single user has asked questions in.

---

users/{id}/top-tags

/me/top-tags

Get the top tags (by score) a single user has posted in.

---

users/{id}/write-permissions

/me/write-permissions

Get the write access a user has via the API.

---

users/moderators

Get the users who have moderation powers on the site.

---

users/moderators/elected

Get the users who are active moderators who have also won a moderator election.

---

users/{id}/inbox

/me/inbox

Get a user's inbox. [auth required]

---

users/{id}/inbox/unread

/me/inbox/unread

Get the unread items in a user's inbox. [auth required]

## Network Methods

These methods return data across the entire Stack Exchange network of sites. Accordingly, you do not pass a site parameter to them.

### Access Tokens

access-tokens/{accessTokens}/invalidate

Allows an application to dispose of `access_tokens` when it is done with them.

---

access-tokens/{accessTokens}

Allows an application to inspect `access_tokens` it has, useful for debugging.

### Achievements

users/{id}/achievements

/me/achievements

Get a user's recent network-wide achievements. [auth required]

### Applications

apps/{accessTokens}/de-authenticate

Allows an application to de-authorize itself for a set of users.

### Errors

errors

Get descriptions of all the errors that the API could return.

errors/{id}

Simulate an API error for testing purposes.

### Filters

filters/create

Create a new filter.

filters/{filters}

Decode a set of filters, useful for debugging purposes.

### Inbox

inbox

Get a user's inbox, outside of the context of a site. [auth required]

inbox/unread

Get the unread items in a user's inbox, outside of the context of a site.
[auth required]

### Notifications

notifications

Get a user's notifications, outside of the context of a site.
[auth required]

notifications/unread

Get a user's unread notifications, outside of the context of a site.
[auth required]

### Sites

sites

Get all the sites in the Stack Exchange network.

### Users

users/{ids}/associated

/me/associated

Get a user's associated accounts.

---

users/{ids}/merges

/me/merges

Get the merges a user's accounts has undergone.

---