



Coaster-Count API documentation

Author and copyright: Volker Sauer

Created: 2014-01-02

Revised: 2020-10-25

Auth-Version: 1

API-Version: 3

Document-Version: 12

Legal: By law, you may not modify, copy, distribute, transmit, display, reproduce or publish any part of the document.

API-Version 2.x is deprecated by October 2020 and maybe be removed in the future!

Document history

Version 1 (2014-01-02):

Url changed.

Version 2 (2014-01-06):

Added user_id to [get_user_all](#) response.

Added special pages section in this document.

Version 3 (2014-01-08):

Added [get_user_id](#) command.

Version 4 (2014-01-14):

Added number of visited continents, countries and parks to [get_user_count](#) and

[get_user_all](#) response.

Added completed countries to [get_user_count](#) and [get_user_all](#) response.

Version 5 (2014-08-13):

Url changed for 2014 version

Added some more stats

Version 5 (2014-08-16):

Typo in example call corrected.

Version 6 (2015-05-24):

Added Coaster-Count TOTAL! check.

Version 7 (2016-09-12):

Updated logo.

Version 8 (2016-12-28):

Switched to SSL.

Version 9 (2019-05-02):

Typo in `get_user_stats` corrected.

Add stats “completed_countries_showmen”, “completed_countries_all”.

Version 10 (2020-03-23):

Add states “count_duplicates”, “visited_showmen”.

Version 11 (2020-10-23):

Added call “`get_user_definition`”.

Deprecate call “`get_user_filters`”.

Updated special pages.

Typos corrected.

Version 12 (2020-10-25):

Added documentation for asynchronous authentication Version 1

Replaced API documentation for complete new version 3

Table of contents

Document history: List of API and document revisions	2
Table of contents: Yeah baby, what do you think are you reading right now?	3
Introduction: Short introduction and hints	4
Preparation: Request API access and adapt your page	4
Asynchronous authentication: Redirect the user to our page to retrieve API key	5
Requesting the endpoint: General concepts, how to request the endpoint	6
Special pages: How to generate URLs for users ridden coasters and more	7

Introduction

The Coaster-Count API endpoint allows you to retrieve and display coaster-count.com information for specific users on your own page.

API Endpoint (SSL only): <https://coaster-count.com/api/user>

Please be aware that the API and / or the URL may change in the future. Any changes will be announced in time to your recorded email address.

Preparation

Please email us at cc@coaster-count.com along with

- Page name
- Page domain (for verifying requests)
- Contact name
- Contact email (for information on API changes)

You will get an individual **page_identifier** and a **passphrase**. Please have in mind that you may access the API endpoint only from the given domain.

After integrating your page/domain to the API, user can grant access for your page in the coaster-count.com options (at bottom of page), as a result they will get an API key, make sure, users can store this key on your page. You will need it for the API call.

Additionally, you can implement an authentication mechanism for easy retrieving the API key (see next chapter).

Asynchronous authentication

Redirect a user with a POST to <https://coaster-count.com/account/authorize> and include the following parameters:

- `version` => 1 (currently)
- `page_identifier` => your individual 8-digit page_identifier
- `url` => where to send user back
- `reference` => optional custom reference will be posted back to you, for example to identify the user
- `signature` => build a sha1 hash from `page_identifier + url + reference + pass_phrase` (all string combined, no spaces)

When no POST data is included or the `url` is missing the user will always run into a 404 error. In any other case the user is requested to log in (if now already) and then grant authorisation for your page. The user will then be directed back to your site with POST back to your given `url` with following parameters:

If an error occurred:

- `result` => "error"
- `error_message` => string explaining the problem
- `error_code` => int
- `reference` => your given reference

In case of success:

- `result` => "success"
- `reference` => your given reference
- `api_key` => the users's API key to store in your database

Requesting the endpoint

Make a POST call against the endpoint <http://coaster-count.com/api/user> by adding the following parameters:

- **version** => 3 (currently)
- **page_identifier** => your individual 8-digit page_identifier
- **api_key** => the api key for the specific user stored in your database
- **command** => the command (see below)
- **signature** => build a sha1 hash from **page_identifier** + **api_key** + **command** + **pass_phrase** (all string combined, no spaces)

The following commands are possible:

- ping => replies with pong for testing purpose
- get_user_id => the user_id to use for linking on special pages (see below)
- get_count => Number of ridden coasters strict, definition, all, with duplicates
- get_visited => Visited parks, showmen, countries and continents
- get_completed => Completed countries parks, showmen, all
- get_stats => Rare points, total and filters
- get_definition => The user's coaster definition
- get_custom => custom data summary, please include "data" parameter as CVS out of (count, visited, completed, stats, definition)

The API will reply with a json result.

If an error occurred:

- **result** => "error"
- **error_message** => string explaining the problem
- **error_code** => int

In case of success:

- **result** => "success"
- **data** => the requested data

Special pages

You can link to these special pages, please replace 123 with the user_id you retrieved from the endpoint. When a user enters his API key initially you should call the API and store the user_id

<https://coaster-count.com/user/123/numbers> => Numbers

<https://coaster-count.com/user/123/ridden> => Ridden coasters

<https://coaster-count.com/user/123/visited> => Visited continents/ countries, parks, showmen

<https://coaster-count.com/user/123/milestones> => Milestones

These pages can be accessed as a logged in user or as a guest if allowed by the user.