Generated Documentation

Contents

Package sc2ranks Procedural Elements . . . 2
sc2ranks.php . . . 2
Package sc2ranks Classes . . . 3
Class sc2ranks_request . . . 3
Constructor __construct . . . 3
Method get_bracket_data . . . 3
Method get_character_data . . . 4
Method get_character_data_by_bnet . . . 4
Method get_character_data_by_profile . . . 4
Method get_latest_map_usage . . . 5
Method get_map_data . . . 5
Method is_json_last_error_enabled . . . 6
Method json_last_error . . . 6
Method last_request . . . 6
Method last_response . . . 6
Method __destruct . . . 6
Appendices . . . 8
Appendix A - Class Trees . . . 9
sc2ranks . . . 9

Package sc2ranks Procedural Elements

sc2ranks.php

This file contains a basic class that handles interaction when generating requests and pulling data back from the sc2ranks.com API. Uses the json interface and the built in PHP json deserializer.

- **Package** sc2ranks
- **Author** Ameer Ayoub < ameer.ayoub@gmail.com>
- **Version** 0.3.3

Package sc2ranks Classes

Class sc2ranks_request

[line 17]

The main class that wraps the requests, all interaction with the API is done through this class, see the example for an example implementation of how to use this class.

- **Package** sc2ranks

Constructor *void* function sc2ranks_request::__construct([$site_key = null], [$method = "fopen"]) *[line 100]*

Function Parameters:

- *string* **$site_key** optional sitekey to use with request, defaults to the value of $_SERVER['SERVER_NAME'].
- *string* **$method** the method to grab url contents, eiher curl or fopen (for file_get_contents).

Constructor

object the function sc2ranks_request::get_bracket_data([$response_object = null], [$bracket = 1]) *[line 292]*

Function Parameters:

- *object* **$response_object** optional response object, defaults
- *int* **$bracket** optional bracket number, defaults to 1, a bracket

Returns the team ranking object for a particular bracket.

- **Access** public

object the function sc2ranks_request::get_character_data($name, $code, [$region = "us"]) *[line 184]*

Function Parameters:

- *string* **$name** character name
- *string|int* **$code** 3 digit character code
- *string* **$region** optional region information, defaults to US

Returns the deserialized character data object from the info provided to the function.

- **Access** public

object the function sc2ranks_request::get_character_data_by_bnet($name, $bnet, [$region = "us"]) *[line 272]*

Function Parameters:

- *string* **$name** character name
- *string|int* **$bnet** bnet ID code
- *string* **$region** optional region information, defaults to US

Returns the deserialized character data object from the info provided to the function.

- **Access** public

object the function sc2ranks_request::get_character_data_by_profile($name, $subtype, $value, [$type = "1t"], [$region = "us"]) *[line 250]*

Function Parameters:

- *string* **$name** character name
- *string* **$subtype** the subtype of the profile search; division, points, etc.
- *string* **$value** the value corresponding to the subtype; Thor Xi, 1700, etc.
- *string* **$type** optional type information, defaults to 1v1 team
- *string* **$region** optional region information, defaults to US

Returns the deserialized character data object from the info provided to the function.

- **Access** public

$object function sc2ranks_request::get_latest_map_usage([$response_object = null]) *[line 217]*

Function Parameters:

- *object* **$response_object** optional response object, defaults

Returns the latest map usage statistics

- **Access** public

object deserialized function sc2ranks_request::get_map_data($map_id) *[line 201]*

Function Parameters:

- *int* **$map_id** the map id to get data for

Gets map usage data from sc2ranks.com

- **Access** public

boolean function sc2ranks_request::is_json_last_error_enabled() *[line 346]*

Returns whether or not json_last_error is enabled

- **Access** public

int|object the function sc2ranks_request::json_last_error() *[line 318]*

Returns the last json error if supported, otherwise returns null

- **Link** json_last_error documentation
- **Access** public

string function sc2ranks_request::last_request() *[line 330]*

Returns the last request url string

- **Access** public

object the function sc2ranks_request::last_response() *[line 338]*

Returns the last repsonse object

- **Access** public

void function sc2ranks_request::__destruct() *[line 117]*

Destructor

Just cleans up the cURL session if we used it

Appendices

Appendix A - Class Trees

Package sc2ranks

sc2ranks_request

- sc2ranks_request

Index

C

constructor sc2ranks_request::__construct() . . . 3
Constructor

S

sc2ranks_request::json_last_error() . . . 6
Returns the last json error if supported, otherwise returns null
sc2ranks_request::is_json_last_error_enabled() . . . 6
Returns whether or not json_last_error is enabled
sc2ranks_request::last_request() . . . 6
Returns the last request url string
sc2ranks_request::last_response() . . . 6
Returns the last repsonse object
sc2ranks_request::__destruct() . . . 6
Destructor
sc2ranks_request::get_map_data() . . . 5
Gets map usage data from sc2ranks.com
sc2ranks_request::get_latest_map_usage() . . . 5
Returns the latest map usage statistics
sc2ranks_request::get_bracket_data() . . . 3
Returns the team ranking object for a particular bracket.
sc2ranks_request . . . 3
The main class that wraps the requests, all interaction with the API is done through this class, see the example for an example implementation of how to use this class.
sc2ranks_request::get_character_data() . . . 4
Returns the deserialized character data object from the info provided to the function.
sc2ranks_request::get_character_data_by_bnet() . . . 4
Returns the deserialized character data object from the info provided to the function.
sc2ranks_request::get_character_data_by_profile() . . . 4
Returns the deserialized character data object from the info provided to the function.
sc2ranks.php . . . 2
This file contains a basic class that handles interaction when generating requests and pulling data back from the sc2ranks.com API. Uses the json interface and the built in PHP json deserializer.