

Package: serpstatr (via r-universe)

October 5, 2026

Type Package

Title 'Serpstat' API Wrapper

Version 0.5.0

URL <https://api-docs.serpstat.com/docs/serpstat-public-api/jenasqbwtxdlr-introduction-to-serpstat-api>

Description The primary goal of 'Serpstat' API
<<https://api-docs.serpstat.com/docs/serpstat-public-api/jenasqbwtxdlr-introduction-to-serpstat-api>>
is to reduce manual SEO (search engine optimization) and PPC
(pay-per-click) tasks. You can automate your keywords research
or competitors analysis with this API wrapper.

License MIT + file LICENSE

Encoding UTF-8

Imports httr (>= 1.4.2)

Suggests testthat

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Alex Danilin [aut, cre]

Maintainer Alex Danilin <alexnikdanilin@gmail.com>

Config/pak/sysreqs libssl-dev

Repository <https://alex-danilin.r-universe.dev>

Date/Publication 2026-07-07 20:00:08 UTC

RemoteUrl <https://github.com/cran/serpstatr>

RemoteRef HEAD

RemoteSha c3bc9829b8e3f32c5f09c75e95cef719b1fea538

Contents

sst_au_export	5
sst_au_get_audit	5
sst_au_get_categories_statistic	6
sst_au_get_default_settings	7
sst_au_get_error_elements	8
sst_au_get_error_rows	9
sst_au_get_history_by_count_error	10
sst_au_get_list	11
sst_au_get_page_audit	11
sst_au_get_pages_list	12
sst_au_get_report_without_details	13
sst_au_get_reports_list_by_page	14
sst_au_get_scan_user_url_list	14
sst_au_get_settings	15
sst_au_get_sub_elements_by_crc	16
sst_au_get_summary	17
sst_au_page_names	18
sst_au_remove	19
sst_au_rescan	19
sst_au_scan	20
sst_au_set_settings	21
sst_au_start	22
sst_au_stop_page	23
sst_au_stop_site	24
sst_au_user_log	25
sst_bl_anchors	25
sst_bl_changes_history	27
sst_bl_domain_summary	28
sst_bl_domains_intersection	29
sst_bl_domains_intersection_summary	30
sst_bl_lost_backlinks	31
sst_bl_lost_outlinks	32
sst_bl_new_backlinks	33
sst_bl_out_domains	34
sst_bl_out_threats	35
sst_bl_out_threats_links	36
sst_bl_outlinks	37
sst_bl_redirected_domains	38
sst_bl_referring_domains	39
sst_bl_sdr_distribution	41
sst_bl_threats	42
sst_bl_threats_links	43
sst_bl_tld_distribution	44
sst_bl_top_anchors	45
sst_bl_top_pages	46
sst_call_api_method	47

sst_cr_get_audit_stats	47
sst_dc_add_task	48
sst_dc_get_task	48
sst_dc_get_task_list	49
sst_lists_to_df	50
sst_pm_create_project	50
sst_pm_delete_project	51
sst_pm_list_projects	52
sst_return_check	53
sst_rt_add_project_competitor	54
sst_rt_add_project_keywords	55
sst_rt_add_project_mirror	56
sst_rt_add_project_region	56
sst_rt_add_rt_project	57
sst_rt_competitors	58
sst_rt_create_and_run_project_tracking	60
sst_rt_create_project_tags	61
sst_rt_delete_project_competitor	61
sst_rt_delete_project_keywords	62
sst_rt_delete_project_mirror	63
sst_rt_delete_project_region	64
sst_rt_delete_project_tags	64
sst_rt_get_cities	65
sst_rt_get_countries	66
sst_rt_get_districts	67
sst_rt_get_project_competitors	67
sst_rt_get_project_keywords	68
sst_rt_get_project_mirrors	69
sst_rt_get_project_positions	70
sst_rt_get_project_regions	71
sst_rt_get_project_status	72
sst_rt_get_project_tags	72
sst_rt_get_projects	73
sst_rt_get_rt_project	74
sst_rt_get_rt_schedule	75
sst_rt_positions_history	75
sst_rt_project_regions	77
sst_rt_run_project_tracking	78
sst_rt_serp_history	79
sst_rt_set_rt_schedule	81
sst_rt_update_project_keywords	82
sst_sa_category_top_domains	83
sst_sa_database_info	84
sst_sa_domain_ad_competitors	85
sst_sa_domain_ad_keywords	86
sst_sa_domain_aio_brand_opportunities	88
sst_sa_domain_aio_summary	90
sst_sa_domain_all_regions_traffic	91

sst_sa_domain_competitors	92
sst_sa_domain_export_positions	93
sst_sa_domain_history	95
sst_sa_domain_keywords	96
sst_sa_domain_keywords_by_language	98
sst_sa_domain_keywords_per_region	100
sst_sa_domain_organic_competitors	101
sst_sa_domain_top_pages	102
sst_sa_domain_urls	104
sst_sa_domains_info	106
sst_sa_domains_intersection	107
sst_sa_domains_rating_data	108
sst_sa_domains_unique_keywords	110
sst_sa_keyword_export_keywords_phrase	111
sst_sa_keyword_export_suggestions	112
sst_sa_keyword_get_ad_keywords	113
sst_sa_keyword_get_ads_competitors	114
sst_sa_keyword_get_competitors	115
sst_sa_keyword_get_keyword_top	116
sst_sa_keyword_get_suggestions	117
sst_sa_keyword_get_top_urls	118
sst_sa_keyword_top	119
sst_sa_keyword_trends	120
sst_sa_keywords	121
sst_sa_keywords_info	123
sst_sa_market_categories	124
sst_sa_related_keywords	125
sst_sa_stats	126
sst_sa_url_get_summary_traffic	127
sst_sa_url_get_url_competitors	128
sst_sa_url_get_url_keywords	129
sst_sa_url_get_url_missing_keywords	130
sst_sc_add_keyword_list	131
sst_sc_add_task	132
sst_sc_get_keyword_serp	133
sst_sc_get_list	134
sst_sc_get_parsing_balance	135
sst_sc_get_task_result	135
sst_tm_activate_user	136
sst_tm_add_user	137
sst_tm_deactivate_user	138
sst_tm_get_list	138
sst_tm_remove_user	139
sst_vc_add_keyword_list_freq	140
sst_vc_get_task_result	141
sst_vc_get_task_status	141

sst_au_export	<i>export</i>
---------------	---------------

Description

export

Usage

```
sst_au_export(  
  report_id,  
  export_type,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

- report_id (required)
- export_type (required) Type of export file:
Type | Description -----|----- mgxlsx | XLSX file mgxlsx_mfiles | Google Sheet puppeteer_pdf | PDF file
- return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
- api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_audit	<i>getAudit</i>
------------------	-----------------

Description

getAudit

Usage

```
sst_au_get_audit(  
    report_id,  
    compare_report_id = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

report_id	(required)
compare_report_id	(optional) 'id' of your another report from your project to compare.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_au_get_categories_statistic  
    getCategoriesStatistic
```

Description

getCategoriesStatistic

Usage

```
sst_au_get_categories_statistic(  
    report_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

report_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_default_settings
<i>getDefaultSettings</i>

Description

getDefaultSettings

Usage

```
sst_au_get_default_settings(  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_error_elements
<i>getErrorElements</i>

Description

getErrorElements

Usage

```
sst_au_get_error_elements(  
    report_id,  
    compare_report_id,  
    project_id,  
    error_name,  
    mode = NULL,  
    limit = NULL,  
    offset = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

report_id	(required)
compare_report_id	(required) Another unique identifier for an audit report from the same project to compare
project_id	(required)
error_name	(required)
mode	(optional) Error display mode: Mode Description ——— ———— all all errors new new errors solved fixed errors
limit	(optional) count of returned items in response
offset	(optional) batch number required for pagination
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

`sst_au_get_error_rows` *getErrorRows*

Description

`getErrorRows`

Usage

```
sst_au_get_error_rows(  
    report_id,  
    error,  
    compare_report_id = NULL,  
    mode = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>report_id</code>	(required)
<code>error</code>	(required)
<code>compare_report_id</code>	(optional) 'id' of your another report from your project to compare.
<code>mode</code>	(optional) type of errors (all new solved) by default all
<code>page</code>	(optional)
<code>size</code>	(optional)
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_au_get_history_by_count_error  
    getHistoryByCountError
```

Description

`getHistoryByCountError`

Usage

```
sst_au_get_history_by_count_error(  
    project_id,  
    error_name,  
    limit,  
    offset,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>project_id</code>	(required)
<code>error_name</code>	(required)
<code>limit</code>	(required) count of returned items in response
<code>offset</code>	(required) batch number required for pagination
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_list	<i>getList</i>
-----------------	----------------

Description

getList

Usage

```
sst_au_get_list(  
    project_id,  
    limit = NULL,  
    offset = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
limit	(optional) count of returned items in response
offset	(optional) batch number required for pagination
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_page_audit	<i>getPageAudit</i>
-----------------------	---------------------

Description

getPageAudit

Usage

```
sst_au_get_page_audit(
    page_id,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

page_id (required)

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_pages_list *getPagesList*

Description

getPagesList

Usage

```
sst_au_get_pages_list(
    limit = NULL,
    offset = NULL,
    team_member_id = NULL,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

limit (optional) count of returned items in response

offset (optional) batch number required for pagination

team_member_id (optional)

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_report_without_details
<i>getReportWithoutDetails</i>

Description

getReportWithoutDetails

Usage

```
sst_au_get_report_without_details(  
    report_id,  
    compare_report_id = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

report_id	(required)
compare_report_id	(optional) Another unique identifier for an audit report from the same project to compare
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_au_get_reports_list_by_page  
    getReportsListByPage
```

Description

`getReportsListByPage`

Usage

```
sst_au_get_reports_list_by_page(  
    page_id,  
    limit = NULL,  
    offset = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>page_id</code>	(required)
<code>limit</code>	(optional) count of returned items in response
<code>offset</code>	(optional) batch number required for pagination
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_au_get_scan_user_url_list  
    getScanUserUrList
```

Description

`getScanUserUrList`

Usage

```
sst_au_get_scan_user_url_list(  
    project_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id (required)

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_settings	<i>getSettings</i>
---------------------	--------------------

Description

getSettings

Usage

```
sst_au_get_settings(  
    project_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id (required)

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_sub_elements_by_crc
<i>getSubElementsByCrc</i>

Description

getSubElementsByCrc

Usage

```
sst_au_get_sub_elements_by_crc(  
    report_id,  
    project_id,  
    error_name,  
    crc,  
    compare_report_id = NULL,  
    mode = NULL,  
    limit = NULL,  
    offset = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

report_id	(required)
project_id	(required)
error_name	(required)
crc	(required) URL crc
compare_report_id	(optional)
mode	(optional) Error display mode: Mode Description ——— ———— all all errors new new errors solved fixed errors
limit	(optional) count of returned items in response
offset	(optional) batch number required for pagination
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_get_summary	<i>Website audit summary</i>
--------------------	------------------------------

Description

Returns the basic stats for the finished audit returns, including number of checked pages, issues by priority, domain optimization score.

Usage

```
sst_au_get_summary(  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN"),  
    report_id = NULL,  
    return_method = "list"  
)
```

Arguments

api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').
report_id	(required) ID of the audit report to get data from.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

Value

Returns the basic metrics for audited website.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0.

Examples

```
## Not run:
sst_au_get_summary(
  report_id = report_id
)$data

## End(Not run)
```

sst_au_page_names	<i>pageNames</i>
-------------------	------------------

Description

pageNames

Usage

```
sst_au_page_names(
  team_member_id = NULL,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

`team_member_id` (optional)

`return_method` (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

`api_token` (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_remove	<i>remove</i>
---------------	---------------

Description

remove

Usage

```
sst_au_remove(  
    page_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

page_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_rescan	<i>rescan</i>
---------------	---------------

Description

rescan

Usage

```
sst_au_rescan(  
    page_id,  
    name,  
    user_agent,  
    http_auth_login = NULL,  
    http_auth_pass = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

page_id	(required)
name	(required) Name of your project. You can change project name in this method if necessary
user_agent	(required)
http_auth_login	(optional) Login for Basic HTTP authentication
http_auth_pass	(optional) Password for Basic HTTP authentication
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_scan

scan

Description

scan

Usage

```
sst_au_scan(
  name,
  url,
  user_agent,
  http_auth_login = NULL,
  http_auth_pass = NULL,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

name	(required) Name of your project
url	(required) Page url to scan
user_agent	(required)

http_auth_login (optional) Login for Basic HTTP authentication
 http_auth_pass (optional) Password for Basic HTTP authentication
 return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
 api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_set_settings	<i>setSettings</i>
---------------------	--------------------

Description

setSettings

Usage

```

sst_au_set_settings(
  project_id,
  main_settings,
  mail_trigger_settings,
  schedule_settings,
  scan_setting,
  dont_scan_keywords_block = NULL,
  only_scan_keywords_block = NULL,
  base_auth_block = NULL,
  errors_settings = NULL,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

project_id (required)
 main_settings (required)
 mail_trigger_settings (required) Trigger mail settings
 schedule_settings (required) Scan schedule settings

scan_setting (required) Scan type settings

dont_scan_keywords_block (optional) The Serpstat robot will exclude pages from crawling if the URL contains any of the specified words. This setting applies when the scan type is set to Sitemap too. Optional block, defaults to '{checked: false, keywords: ""}'.

only_scan_keywords_block (optional) The Serpstat robot will crawl pages with the following words in the URL. The setting is relevant for the scan type Sitemap too. Optional block, defaults to '{checked: false, keywords: ""}'.

base_auth_block (optional) If the site is restricted access, provide access for verification. Optional block, defaults to '{login: "", password: ""}'.

errors_settings (optional) Optional settings for error thresholds. If omitted, default values are used.

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_start	<i>Start a website audit</i>
--------------	------------------------------

Description

Starts a project website audit with the current audit settings. Uses default audit settings if no settings were changed.

Usage

```
sst_au_start(
  api_token = Sys.getenv("SERPSTAT_API_TOKEN"),
  project_id = NULL,
  return_method = "list"
)
```

Arguments

api_token (required) Serpstat API token from [your profile](#). Default is Sys.getenv('SERPSTAT_API_TOKEN').

project_id (required) ID of the project in Serpstat.

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

Value

Returns the ID of the audit report.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 audit credit for each checked page.

Examples

```
## Not run:
sst_au_start(
  project_id = 12345
)$data$reportId

## End(Not run)
```

sst_au_stop_page	stop
------------------	------

Description

stop

Usage

```
sst_au_stop_page(
  page_id,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

page_id (required)

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_stop_site	<i>stop</i>
------------------	-------------

Description

stop

Usage

```
sst_au_stop_site(  
    project_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

- project_id (required)
- return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
- api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_au_user_log	<i>userLog</i>
-----------------	----------------

Description

userLog

Usage

```
sst_au_user_log(  
    report_id = NULL,  
    page_size = 100,  
    page = 1,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

report_id	(optional)
page_size	(optional) Number of results per page in response
page	(optional) page number for pagination
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_bl_anchors	<i>Referring anchors</i>
----------------	--------------------------

Description

Returns anchors of backlinks for the analyzed domain.

Usage

```
sst_bl_anchors(  
    domain,  
    search_type = "domain",  
    anchor = NULL,  
    count = NULL,  
    sort = NULL,  
    order = NULL,  
    page = 1,  
    size = 100,  
    filter = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
anchor	(optional) Anchor text to filter by.
count	(optional) Number of words in anchor to filter by.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
filter	(optional) The nested list of filtering options.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns anchors data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned anchor.

`sst_bl_changes_history`*Backlinks changes history*

Description

Returns history of backlinks changes (new and lost backlinks) over time.

Usage

```
sst_bl_changes_history(  
    domain,  
    search_type = "domain",  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>domain</code>	(required) A domain name to analyze.
<code>search_type</code>	(optional) Default value is 'domain' for domain data only.
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns changes history data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per request.

sst_bl_domain_summary *Backlinks summary*

Description

Returns the overview of the backlinks profile for the domain.

Usage

```
sst_bl_domain_summary(  
    domain,  
    search_type = "domain",  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain only (site.com). See API docs for more details.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns aggregated backlinks data for the domain.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per request.

Examples

```
## Not run:  
sst_bl_domain_summary(  
    domain      = 'serpstat.com',  
    search_type = 'domain',  
    return_method = 'list'  
)$data  
  
## End(Not run)
```

sst_bl_domains_intersection
Intersecting referring domains

Description

Returns referring domains that link to comparison domains but not to the target domain.

Usage

```
sst_bl_domains_intersection(  
    domain,  
    compare_domains,  
    sort = NULL,  
    order = NULL,  
    page = 1,  
    size = 100,  
    filter = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) Target domain name.
compare_domains	(required) A character vector of domains for comparison.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
filter	(optional) The nested list of filtering options.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns common referring domains data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per each referring domain in response.

`sst_bl_domains_intersection_summary`*Intersecting referring domains summary*

Description

Returns summary of intersecting referring domains.

Usage

```
sst_bl_domains_intersection_summary(  
  domain,  
  compare_domains,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>domain</code>	(required) Target domain name.
<code>compare_domains</code>	(required) A character vector of domains to find common donors with.
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns summary of common referring domains.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per request.

`sst_bl_lost_backlinks` *Lost backlinks*

Description

Returns lost backlinks for the analyzed domain.

Usage

```
sst_bl_lost_backlinks(  
    domain,  
    search_type = "domain",  
    sort = NULL,  
    order = NULL,  
    links_per_domain = NULL,  
    filter = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>domain</code>	(required) A domain name to analyze.
<code>search_type</code>	(optional) Default value is 'domain' for domain data only.
<code>sort</code>	(optional) A field to sort the response.
<code>order</code>	(optional) The order of sorting.
<code>links_per_domain</code>	(optional) Limit backlinks returned from a single domain.
<code>filter</code>	(optional) The nested list of filtering options.
<code>page</code>	(optional) Response page number. Default is 1.
<code>size</code>	(optional) Response page size. Default is 100.
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns lost backlinks data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned lost backlink.

sst_bl_lost_outlinks	<i>Lost outgoing links</i>
----------------------	----------------------------

Description

Returns lost outgoing links from the analyzed domain/page.

Usage

```
sst_bl_lost_outlinks(
    domain,
    search_type = "domain",
    sort = NULL,
    order = NULL,
    links_per_domain = NULL,
    filter = NULL,
    page = 1,
    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
links_per_domain	(optional) Limit outgoing links returned to a single domain.
filter	(optional) The nested list of filtering options.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns lost outgoing links data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned lost outgoing link.

sst_bl_new_backlinks	<i>New backlinks</i>
----------------------	----------------------

Description

Returns new backlinks for the analyzed domain.

Usage

```
sst_bl_new_backlinks(
    domain,
    search_type = "domain",
    sort = NULL,
    order = NULL,
    links_per_domain = NULL,
    filter = NULL,
    page = 1,
    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
links_per_domain	(optional) Limit backlinks returned from a single domain.
filter	(optional) The nested list of filtering options.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns new backlinks data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned new backlink.

sst_bl_out_domains	<i>Outgoing domains</i>
--------------------	-------------------------

Description

Returns domains linked from the analyzed domain.

Usage

```
sst_bl_out_domains(  
    domain,  
    search_type = "domain",  
    sort = NULL,  
    order = NULL,  
    filter = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
filter	(optional) The nested list of filtering options.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns outgoing domains data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned outgoing domain.

sst_bl_out_threats	<i>Outgoing backlinks threats</i>
--------------------	-----------------------------------

Description

Returns domain-level malicious or vulnerability threats in outgoing links.

Usage

```
sst_bl_out_threats(
    domain,
    search_type = "domain",
    sort = NULL,
    order = NULL,
    links_per_domain = NULL,
    filter = NULL,
    page = 1,
    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
links_per_domain	(optional) Limit threats returned from a single domain.
filter	(optional) The nested list of filtering options.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns outgoing threats data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned threat.

sst_bl_out_threats_links
<i>Outgoing backlinks threats links</i>

Description

Returns link-level malicious or vulnerability threats in outgoing links.

Usage

```
sst_bl_out_threats_links(  
    domain,  
    search_type = "domain",  
    sort = NULL,  
    order = NULL,  
    links_per_domain = NULL,  
    filter = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

- domain (required) A domain name to analyze.
- search_type (optional) Default value is 'domain' for domain data only.
- sort (optional) A field to sort the response.
- order (optional) The order of sorting.
- links_per_domain (optional) Limit threats returned from a single domain.
- filter (optional) The nested list of filtering options.
- page (optional) Response page number. Default is 1.
- size (optional) Response page size. Default is 100.

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns outgoing malicious links data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned malicious link.

sst_bl_outlinks	<i>Outgoing links</i>
-----------------	-----------------------

Description

Returns outgoing links from the analyzed domain/page.

Usage

```
sst_bl_outlinks(  
  domain,  
  search_type = "domain",  
  sort = NULL,  
  order = NULL,  
  links_per_domain = NULL,  
  filter = NULL,  
  page = 1,  
  size = 100,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain (required) A domain name to analyze.

search_type (optional) Default value is 'domain' for domain data only.

sort (optional) A field to sort the response.

order (optional) The order of sorting.

links_per_domain (optional) Limit outgoing links returned to a single domain.

filter	(optional) The nested list of filtering options.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns outgoing links data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned outgoing link.

sst_bl_redirected_domains

Redirected domains

Description

Returns domains from which redirects were performed to the analyzed domain.

Usage

```
sst_bl_redirected_domains(
  domain,
  sort = NULL,
  order = NULL,
  filter = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) A domain name to analyze.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
filter	(optional) The nested list of filtering options.

page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns redirected domains data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned redirected domain.

sst_bl_referring_domains
Referring domains

Description

Returns the list of referring domains with main backlinks metrics for each domain.

Usage

```
sst_bl_referring_domains(  
  domain,  
  search_type = "domain",  
  page = 1,  
  size = 100,  
  sort = "domain_rank",  
  order = "desc",  
  filter = NULL,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(required) Default value is 'domain' for domain data only (site.com). See API docs for more details.

page	(optional) Response page number if there are many pages in response. Default is 1.
size	(optional) Response page size. Default is 100.
sort	(optional) A field to sort the response. Default is 'domain_rank'. See API docs for more details.
order	(optional) The order of sorting. Default is 'desc' for descending order. See API docs for more details.
filter	(optional) The nested list of filtering options. See API docs for more details.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns aggregated backlinks data for each referring domain.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per each domain in response.

Sorting

You can sort the response using `sort` argument. The sorting order is defined using `order` argument.

Examples

```
## Not run:
sst_bl_referring_domains(
  domain      = 'serpstat.com',
  page        = 1,
  size        = 100,
  sort        = 'domain_rank',
  order       = 'desc',
  filter      = NULL,
  return_method = 'list'
)$data

## End(Not run)
```

sst_bl_sdr_distribution
<i>SDR distribution</i>

Description

Returns distribution of SDR (Serpstat Domain Rank) for the referring domains.

Usage

```
sst_bl_sdr_distribution(  
    domain,  
    search_type = "domain",  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns SDR distribution data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per request.

`sst_bl_threats`*Backlinks threats*

Description

Returns domain-level malicious or vulnerability threats found in the backlinks profile.

Usage

```
sst_bl_threats(  
    domain,  
    search_type = "domain",  
    sort = NULL,  
    order = NULL,  
    links_per_domain = NULL,  
    filter = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>domain</code>	(required) A domain name to analyze.
<code>search_type</code>	(optional) Default value is 'domain' for domain data only.
<code>sort</code>	(optional) A field to sort the response.
<code>order</code>	(optional) The order of sorting.
<code>links_per_domain</code>	(optional) Limit threats returned from a single domain.
<code>filter</code>	(optional) The nested list of filtering options.
<code>page</code>	(optional) Response page number. Default is 1.
<code>size</code>	(optional) Response page size. Default is 100.
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns threats data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned threat.

sst_bl_threats_links *Backlinks threats links*

Description

Returns link-level malicious or vulnerability threats found in the backlinks profile.

Usage

```
sst_bl_threats_links(
    domain,
    search_type = "domain",
    sort = NULL,
    order = NULL,
    links_per_domain = NULL,
    filter = NULL,
    page = 1,
    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
links_per_domain	(optional) Limit threats returned from a single domain.
filter	(optional) The nested list of filtering options.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns malicious links data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned malicious link.

sst_bl_tld_distribution
TLD distribution

Description

Returns distribution of top-level domains (TLDs) for the referring domains.

Usage

```
sst_bl_tld_distribution(  
    domain,  
    search_type = "domain",  
    language = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
language	(optional) Language in which the country_name field is returned.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns TLD distribution data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per request.

sst_bl_top_anchors	<i>Top anchors</i>
--------------------	--------------------

Description

Returns top anchors for the analyzed domain.

Usage

```
sst_bl_top_anchors(  
    domain,  
    search_type = "domain",  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns top anchors data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per request.

sst_bl_top_pages	<i>Top pages by backlinks</i>
------------------	-------------------------------

Description

Returns top pages of the analyzed domain by backlinks count.

Usage

```
sst_bl_top_pages(
    domain,
    search_type = "domain",
    sort = NULL,
    order = NULL,
    filter = NULL,
    page = 1,
    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) A domain name to analyze.
search_type	(optional) Default value is 'domain' for domain data only.
sort	(optional) A field to sort the response.
order	(optional) The order of sorting.
filter	(optional) The nested list of filtering options.
page	(optional) Response page number. Default is 1.
size	(optional) Response page size. Default is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns top pages data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned page.

sst_call_api_method	<i>Make a request to Serpstat API endpoint</i>
---------------------	--

Description

Make a request to Serpstat API endpoint

Usage

```
sst_call_api_method(api_token, api_method, api_params = NULL)
```

Arguments

api_token	Serpstat API token from the profile page . Default is <code>Sys.getenv('SERPSTAT_API_TOKEN')</code> .
api_method	Internal name of API method.
api_params	A list of API parameters used by api_method. More information about parameters in the official docs .

Value

The list with a response data.

sst_cr_get_audit_stats	<i>getAuditStats</i>
------------------------	----------------------

Description

getAuditStats

Usage

```
sst_cr_get_audit_stats(
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_dc_add_task	<i>addTask</i>
-----------------	----------------

Description

addTask

Usage

```
sst_dc_add_task(  
    domains,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domains	(required) An array of domain names
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_dc_get_task	<i>getTask</i>
-----------------	----------------

Description

getTask

Usage

```
sst_dc_get_task(  
    task_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

task_id	(required) Task identifier
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_dc_get_task_list	<i>getTaskList</i>
----------------------	--------------------

Description

getTaskList

Usage

```
sst_dc_get_task_list(  
  page = 1,  
  size = 100,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

page	(required) Page number in response
size	(required) Number of results per page in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_lists_to_df	<i>Convert list of lists to data.frame</i>
-----------------	--

Description

API response might contain nested lists with different number of elements. This function fills missing elements and combine lists to a data.frame.

Usage

```
sst_lists_to_df(lists, fill = NA)
```

Arguments

lists	- a list of nested lists with different number of elements
fill	- a value to fill missing values in lists

Value

A data.frame with all missing values filled with specified value.

Examples

```
sst_lists_to_df(  
  lists = list(  
    first_list = list(a = 1, b = 2),  
    second_list = list(a = 2, c = 3)  
  ),  
  fill = 'empty'  
)
```

sst_pm_create_project	<i>Create a new project</i>
-----------------------	-----------------------------

Description

Creates a new project in Serpstat.

Usage

```
sst_pm_create_project(  
  domain,  
  name,  
  groups = NULL,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) Domain to get data for.
name	(required) The name of the project. Can be different from the domain.
groups	(optional) A list of project groups the project should be added to.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns the project ID for the created project.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 project credit.

Examples

```
## Not run:
sst_pm_create_project(
  domain = 'serpstat.com',
  name   = 'Serpstat'
)$data$project_id

## End(Not run)
```

sst_pm_delete_project *Delete the existing project*

Description

Deletes the existing project in Serpstat by project ID.

Usage

```
sst_pm_delete_project(
  api_token = Sys.getenv("SERPSTAT_API_TOKEN"),
  project_id = NULL,
  return_method = "list"
)
```

Arguments

api_token (required) Serpstat API token from [your profile](#). Default is `Sys.getenv('SERPSTAT_API_TOKEN')`.
project_id (required) ID of the project in Serpstat.
return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

Value

Returns the state of the deletion operation.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

returns 1 project credit.

Examples

```
## Not run:  
sst_pm_delete_project(  
  project_id = 12345  
)  
  
## End(Not run)
```

sst_pm_list_projects *List existing projects*

Description

Gets a list of existing projects available for the users with their basic information.

Usage

```
sst_pm_list_projects(  
  page = 1,  
  size = 100,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns basic information on all the projects.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0.

Examples

```
## Not run:
sst_pm_list_projects(
  page      = 2,
  size      = 10
)

## End(Not run)
```

sst_return_check	<i>Preprocess the API response</i>
------------------	------------------------------------

Description

Every API call returns a JSON object. This object is transformed to a list. Depending on return_method parameter the data element of this list will be a list or data.frame.

Usage

```
sst_return_check(response_content, return_method)
```

Arguments

response_content	The result of sst_call_api_method call.
return_method	Accepted values are 'list' to return data object as list or 'df' to return data object as data.frame.

Value

response_content with a data object as list or data.frame.

```
sst_rt_add_project_competitor  
    addProjectCompetitor
```

Description

addProjectCompetitor

Usage

```
sst_rt_add_project_competitor(  
  project_id = NULL,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(optional)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_rt_add_project_keywords  
    addProjectKeywords
```

Description

`addProjectKeywords`

Usage

```
sst_rt_add_project_keywords(  
    project_id,  
    keywords,  
    tags = NULL,  
    url = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>project_id</code>	(required)
<code>keywords</code>	(required) Array of keywords
<code>tags</code>	(optional) Tags for keywords (created automatically if don't exist)
<code>url</code>	(optional) Target URL for rank checking
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_rt_add_project_mirror  
    addProjectMirror
```

Description

addProjectMirror

Usage

```
sst_rt_add_project_mirror(  
    project_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_rt_add_project_region  
    addProjectRegion
```

Description

addProjectRegion

Usage

```
sst_rt_add_project_region(  
    project_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_add_rt_project *addRtProject*

Description

addRtProject

Usage

```
sst_rt_add_rt_project(  
  name,  
  domain,  
  group = NULL,  
  parsing_type = NULL,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

name	(required) Project name
domain	(required) Main domain to track
group	(optional) Project group
parsing_type	(optional)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_competitors	<i>Get the data on competitors in search results</i>
--------------------	--

Description

This method returns the competing in top20 search results domains that rank for at least two keywords that are added the project.

Usage

```
sst_rt_competitors(
    project_id,
    region_id,
    date_from = Sys.Date() - 8,
    date_to = Sys.Date() - 1,
    domains = NULL,
    sort = "sum_traffic",
    sort_range = "top1",
    order = "desc",
    page = 1,
    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

project_id	(required) The ID of your project in Serpstat. You can find this ID in the URL of any rank tracker report. As an example, in https://serpstat.com/rank-tracker/keywords/12345/positions?get_params the ID would be 12345.
region_id	(required) The ID of a region returned by sst_rt_project_regions .
date_from	(optional) The date string in 'YYYY-MM-DD' format to specify the initial date of retrieved data. Default value is current date minus 8 days.
date_to	(optional) The date string in 'YYYY-MM-DD' format to specify the final date of retrieved data. Must not exceed date_from + 30 days. Default value is yesterday.
domains	(optional) A vector of domain names for which the data should be retrieved. By default the data is retrieved for all domains that rank for at least two keywords that are added to the project.
sort	(optional) Must be one of 'sum_traffic' (default, domain search traffic distribution), 'keywords_count' (number of keywords), 'avg_position' (average domain position), 'position_ranges' (ranges of positions), 'ads_count' (number of ads in search engine results).

sort_range	(optional) The subcategory of ranges of positions to sort by. Must be one of 'top1', 'top3', 'top5', 'top10', 'top20', 'top101' to sort by the number of keywords in specific positions range; or 'keywords_count_top', 'keywords_count_bottom' to sort by the number of keywords in search ads blocks; or 'avg_position_top', 'avg_position_bottom' to sort by the average position in search ads blocks.
order	(optional) The sorting order. Must be one of string 'desc' (default) for descending sorting or 'asc' for ascending sorting.
page	(optional) Response page number if there are many pages in response. The default value is 1.
size	(optional) Response page size. Must be one of 20, 50, 100, 200, 500. The default value is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns traffic and keywords distributions, average positions of the domains by date.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0

Examples

```
## Not run:
project_id <- 12345
region_id <- sst_rt_competitors(
  project_id = project_id
)$data$regions[[1]]$id
sst_rt_competitors(
  api_token      = api_token,
  project_id     = project_id,
  region_id      = region_id,
  date_from      = '2020-12-01',
  date_to        = '2020-12-30',
  domains        = c('serpstat.com', 'serpstatbot.com'),
  sort           = 'sum_traffic',
  sort_range     = 'top1',
  order          = 'desc',
  page           = 1,
  size           = 20,
  return_method  = 'list'
)

## End(Not run)
```

```
sst_rt_create_and_run_project_tracking
    createAndRunProjectTracking
```

Description

createAndRunProjectTracking

Usage

```
sst_rt_create_and_run_project_tracking(
    domain,
    keywords,
    name = NULL,
    group = NULL,
    regions = NULL,
    parsing_type = NULL,
    mirrors = NULL,
    schedule = NULL,
    competitors = NULL,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) Main domain
keywords	(required)
name	(optional) Project name
group	(optional) Project group
regions	(optional) Tracking regions (auto-detected if not specified)
parsing_type	(optional)
mirrors	(optional) Domain mirrors (max. 10)
schedule	(optional)
competitors	(optional) Competitors (max. 10)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_create_project_tags
createProjectTags

Description

createProjectTags

Usage

```
sst_rt_create_project_tags(  
    project_id,  
    tags,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
tags	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_delete_project_competitor
deleteProjectCompetitor

Description

deleteProjectCompetitor

Usage

```
sst_rt_delete_project_competitor(
    competitor_uuid,
    project_id,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

competitor_uuid	(required) Competitor UUID, can be obtained from ‘getProjectCompetitors’
project_id	(required)
return_method	(optional) Accepted values are ‘list’ (default) to return data object as list or ‘df’ to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_rt_delete_project_keywords
      deleteProjectKeywords
```

Description

deleteProjectKeywords

Usage

```
sst_rt_delete_project_keywords(
    project_id,
    keyword_ids,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

project_id	(required)
keyword_ids	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_rt_delete_project_mirror  
    deleteProjectMirror
```

Description

deleteProjectMirror

Usage

```
sst_rt_delete_project_mirror(  
  project_id,  
  mirror_uuid,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
mirror_uuid	(required) Mirror UUID (from getProjectMirrors)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_delete_project_region
deleteProjectRegion

Description

deleteProjectRegion

Usage

```
sst_rt_delete_project_region(  
    project_id,  
    region_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
region_id	(required) Tracking region ID (from getProjectRegions)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_delete_project_tags
deleteProjectTags

Description

deleteProjectTags

Usage

```
sst_rt_delete_project_tags(
    project_id,
    tag_uuids,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

`project_id` (required)

`tag_uuids` (required) Array of tag UUIDs to delete

`return_method` (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

`api_token` (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_cities	<i>getCities</i>
-------------------	------------------

Description

getCities

Usage

```
sst_rt_get_cities(
    district_id,
    search_engine,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

`district_id` (required) District ID (from getDistricts method)

`search_engine` (required)

`return_method` (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

`api_token` (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_countries	<i>getCountries</i>
----------------------	---------------------

Description

getCountries

Usage

```
sst_rt_get_countries(  
    search_engine,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

search_engine	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_districts *getDistricts*

Description

getDistricts

Usage

```
sst_rt_get_districts(  
  country_id,  
  search_engine,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

country_id	(required) Country ID (from getCountries method)
search_engine	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_project_competitors
 getProjectCompetitors

Description

getProjectCompetitors

Usage

```
sst_rt_get_project_competitors(  
  project_id,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_rt_get_project_keywords
      getProjectKeywords
```

Description

getProjectKeywords

Usage

```
sst_rt_get_project_keywords(
  project_id,
  keyword_ids = NULL,
  search_string = NULL,
  tags = NULL,
  page = 1,
  size = 100,
  sort = NULL,
  order = NULL,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

project_id	(required)
keyword_ids	(optional) Filter by keyword ID array
search_string	(optional)
tags	(optional) Filter by tags (empty array = all tags)
page	(optional)
size	(optional)

sort	(optional)
order	(optional)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_rt_get_project_mirrors  
    getProjectMirrors
```

Description

getProjectMirrors

Usage

```
sst_rt_get_project_mirrors(  
  project_id,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_project_positions
getProjectPositions

Description

getProjectPositions

Usage

```
sst_rt_get_project_positions(  
    project_id,  
    project_region_ids,  
    domains = NULL,  
    filter = NULL,  
    page = 1,  
    page_size = 100,  
    date_from = NULL,  
    date_to = NULL,  
    sort = NULL,  
    order = NULL,  
    sort_region = NULL,  
    tag_ids = NULL,  
    keyword_ids = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
project_region_ids	(required) Filter by project regions (get via getProjectRegions)
domains	(optional) Filter by domains (UUID) - main + mirrors + competitors
filter	(optional)
page	(optional)
page_size	(optional)
date_from	(optional)
date_to	(optional)
sort	(optional)
order	(optional)
sort_region	(optional)
tag_ids	(optional)

keyword_ids (optional)
return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_project_regions
getProjectRegions

Description

getProjectRegions

Usage

```
sst_rt_get_project_regions(  
  project_id,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id (required)
return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_project_status
getProjectStatus

Description

getProjectStatus

Usage

```
sst_rt_get_project_status(  
    project_id,  
    region_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required) Project identifier in numeric representation
region_id	(required) Search region ID (get via getProjectRegions)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_project_tags
getProjectTags

Description

getProjectTags

Usage

```
sst_rt_get_project_tags(  
    project_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_projects	<i>getProjects</i>
---------------------	--------------------

Description

getProjects

Usage

```
sst_rt_get_projects(  
    page = 1,  
    page_size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

page	(optional) Page number in the projects list
page_size	(optional) Number of results per page in the project list.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

`sst_rt_get_rt_project` *getRtProject*

Description

getRtProject

Usage

```
sst_rt_get_rt_project(  
  project_id,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>project_id</code>	(required)
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_get_rt_schedule	
	<i>getRtSchedule</i>

Description

getRtSchedule

Usage

```
sst_rt_get_rt_schedule(  
    project_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_positions_history	
	<i>Get ranking history for the domain or URL in selected search region</i>

Description

This method method returns the rankings for specified domain is selected search region.

Usage

```

sst_rt_positions_history(
  project_id,
  region_id,
  date_from = Sys.Date() - 8,
  date_to = Sys.Date() - 1,
  keywords = NULL,
  url = NULL,
  tags = FALSE,
  sort = "keyword",
  order = "desc",
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

project_id	(required) The ID of your project in Serpstat. You can find this ID in the URL of any rank tracker report. As an example, in https://serpstat.com/rank-tracker/keywords/12345/positions?get_params the ID would be 12345.
region_id	(required) The ID of a region returned by sst_rt_project_regions .
date_from	(optional) The date string in 'YYYY-MM-DD' format to specify the initial date of retrieved data. Default value is current date minus 8 days.
date_to	(optional) The date string in 'YYYY-MM-DD' format to specify the final date of retrieved data. Must not exceed date_from + 30 days. Default is yesterday.
keywords	(optional) A vector of keywords for which the data should be retrieved. Maximum 1000 keywords per request. By default all the data for all keywords in the project is returned.
url	(optional) The domain name (e.g. domain.com) or web page address (e.g. https://domain.com/page) to get the data for. By default the results are returned for the projects' domain name.
tags	(optional) TRUE if keyword tags should be retrieved. Default is FALSE.
sort	(optional) Must be one of 'keyword' (default) to sort the results alphabetically or 'date' to sort the results by date.
order	(optional) The sorting order. Must be one of string 'desc' (default) for descending sorting or 'asc' for ascending sorting.
page	(optional) Response page number if there are many pages in response. The default value is 1.
size	(optional) Response page size. Must be one of 20, 50, 100, 200, 500. The default value is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns positions of selected domain in search engine results in selected region with corresponding URLs for these positions.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0

Examples

```
## Not run:
project_id <- 12345
region_id <- sst_rt_project_regions(
  project_id = project_id
)$data$regions[[1]]$id
sst_rt_positions_history(
  api_token    = api_token,
  project_id   = project_id,
  region_id    = region_id,
  date_from    = '2020-12-01',
  date_to      = '2020-12-30',
  keywords     = c('seo', 'ppc', 'serpstat'),
  url          = 'serpstat.com',
  sort         = 'keyword',
  order        = 'desc',
  page         = 1,
  size         = 100,
  return_method = 'list'
)

## End(Not run)
```

sst_rt_project_regions

Get all regions for the project

Description

In Serpstat you are able to track ranking of your website in multiple regions. This method returns all the regions in your Serpstat project. You will need the results of this method to get rankings in selected region.

Usage

```
sst_rt_project_regions(
    project_id,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

project_id	(required) The ID of your project in Serpstat. You can find this ID in the URL of any rank tracker report. As an example, in https://serpstat.com/rank-tracker/keywords/12345/positions?get_params the ID would be 12345.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns the regions of the project with their ID, state (active or not) and other region attributes.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0

Examples

```
## Not run:
sst_rt_project_regions(project_id = 12345)

## End(Not run)
```

```
sst_rt_run_project_tracking
      runProjectTracking
```

Description

runProjectTracking

Usage

```
sst_rt_run_project_tracking(  
    project_id,  
    keyword_ids = NULL,  
    region_ids = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
keyword_ids	(optional) Keyword IDs (if empty, runs for all keywords)
region_ids	(optional) Region IDs (if empty, runs for all regions)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_rt_serp_history	<i>Get search results history in search region by keywords</i>
---------------------	--

Description

This method returns top 100 search results in Google.

Usage

```
sst_rt_serp_history(  
    project_id,  
    region_id,  
    date_from = Sys.Date() - 8,  
    date_to = Sys.Date() - 1,  
    keywords = NULL,  
    tags = FALSE,  
    sort = "keyword",  
    order = "desc",  
    page = 1,  
    size = 100,
```

```

    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

project_id	(required) The ID of your project in Serpstat. You can find this ID in the URL of any rank tracker report. As an example, in https://serpstat.com/rank-tracker/keywords/12345/positions?get_params the ID would be 12345.
region_id	(required) The ID of a region returned by sst_rt_project_regions .
date_from	(optional) The date string in 'YYYY-MM-DD' format to specify the initial date of retrieved data. Default value is current date minus 8 days.
date_to	(optional) The date string in 'YYYY-MM-DD' format to specify the final date of retrieved data. Must not exceed date_from + 30 days. Default is yesterday.
keywords	(optional) A vector of keywords for which the data should be retrieved. Maximum 1000 keywords per request. By default all the data for all keywords in the project is returned.
tags	(optional) TRUE if keyword tags should be retrieved. Default is FALSE.
sort	(optional) Must be one of 'keyword' (default) to sort the results alphabetically or 'date' to sort the results by date.
order	(optional) The sorting order. Must be one of string 'desc' (default) for descending sorting or 'asc' for ascending sorting.
page	(optional) Response page number if there are many pages in response. The default value is 1.
size	(optional) Response page size. Must be one of 20, 50, 100, 200, 500. The default value is 100.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns the search engine results for specific dates and region including positions and URLs.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0

Examples

```
## Not run:
project_id <- 12345
region_id <- sst_rt_project_regions(
  project_id = project_id
)$data$regions[[1]]$id
sst_rt_serp_history(
  api_token    = api_token,
  project_id   = project_id,
  region_id    = region_id,
  date_from    = '2020-12-01',
  date_to      = '2020-12-30',
  keywords     = c('seo', 'ppc', 'serpstat'),
  sort         = 'keyword',
  order        = 'desc',
  page         = 1,
  size         = 100,
  return_method = 'list'
)

## End(Not run)
```

sst_rt_set_rt_schedule

setRtSchedule

Description

setRtSchedule

Usage

```
sst_rt_set_rt_schedule(
  project_id,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

project_id (required)

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_rt_update_project_keywords
    updateProjectKeywords
```

Description

updateProjectKeywords

Usage

```
sst_rt_update_project_keywords(  
    project_id,  
    keywords,  
    regions = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

project_id	(required)
keywords	(required) List of keywords to update. From 1 to 100 elements.
regions	(optional) List of project region IDs to apply the update to. If not provided or all provided IDs do not belong to the project, the update applies to all project regions. Max 100 elements.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

`sst_sa_category_top_domains`*Top Domains in Market Category*

Description

Returns top-performing domains in a specific market category with their SEO metrics.

Usage

```
sst_sa_category_top_domains(  
    category_id,  
    se,  
    sort = NULL,  
    order = NULL,  
    filters = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>category_id</code>	(required) Category identifier in format .X.Y.Z. (must match regex pattern <code>^\.([0-9]+\.)*\$</code>).
<code>se</code>	(required) Search engine alias (db_name) returned by sst_sa_database_info .
<code>sort</code>	(optional) Sort field name.
<code>order</code>	(optional) Sort order.
<code>filters</code>	(optional) A list of filtering options. See Filtering for more details.
<code>page</code>	(optional) Response page number if there are many pages in response.
<code>size</code>	(optional) Response page size.
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token from your profile . Default is <code>Sys.getenv('SERPSTAT_API_TOKEN')</code> .

Value

Returns top-performing domains for the category.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 credit per row in response.

Filtering

To filter the results you can use `filters` argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, `filters = list(queries_from = 0, queries_to = 10)` would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_category_top_domains(
  category_id = '.1.',
  se          = 'g_us',
  page        = 1,
  size        = 100,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_database_info *List all Serpstat databases*

Description

In every request to get data from search analytics API you must set `se` parameter to specify from what country do you want to get the data. This method returns all acceptable values for `se` parameter with corresponding country names.

Usage

```
sst_sa_database_info(
  api_token = Sys.getenv("SERPSTAT_API_TOKEN"),
  return_method = "list"
)
```

Arguments

`api_token` (required) Serpstat API token from **your profile**. Default is `Sys.getenv('SERPSTAT_API_TOKEN')`.

`return_method` (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as `data.frame`.

Value

Returns country name, `se` parameter value and local search engine domain name for each country.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0

Examples

```
## Not run:
sst_sa_database_info()$data

## End(Not run)
```

```
sst_sa_domain_ad_competitors
      Domain advertising competitors
```

Description

Returns competing domains in paid search (PPC) results.

Usage

```
sst_sa_domain_ad_competitors(
  domain,
  se,
  sort = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) Domain name to analyze.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) A field to sort the response. See Sorting for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a list of PPC competitors with their metrics.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned competitor.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Examples

```
## Not run:
sst_sa_domain_ad_competitors(
  domain      = 'serpstat.com',
  se          = 'g_us',
  page        = 1,
  size        = 100,
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_domain_ad_keywords
```

Domain advertising keywords

Description

Returns keywords in paid search results and advertising listings for a specific domain.

Usage

```
sst_sa_domain_ad_keywords(
  domain,
  se,
  url = NULL,
  keywords = NULL,
  minus_keywords = NULL,
  sort = NULL,
```

```

    filters = NULL,
    page = 1,
    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

domain	(required) Domain name to analyze.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
url	(optional) Get the results for this URL only.
keywords	(optional) A vector of words. Keywords in response will contain these words.
minus_keywords	(optional) A vector of words. Keywords in response will not contain these words.
sort	(optional) A field to sort the response. See Sorting for more details.
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a list of PPC keywords with their metrics.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per keyword in response.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Filtering

To filter the results you can use filters argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See [API docs](#) for more details. For example, filters = list(queries_from = 0, queries_to = 10) would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_ad_keywords(
  domain      = 'serpstat.com',
  se          = 'g_us',
  url         = 'https://serpstat.com/',
  keywords    = list('google'),
  minus_keywords = list('download'),
  page        = 1,
  size        = 100,
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_domain_aio_brand_opportunities
      AI Overview Brand Opportunities
```

Description

Returns a paginated list of keywords with AI Overview (AIO) data for a given brand and list of domains.

Usage

```
sst_sa_domain_aio_brand_opportunities(
  domains,
  brand,
  search_engine = "g_us",
  page = 1,
  page_size = 100,
  sort = NULL,
  order = NULL,
  filters = NULL,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domains	(required) Vector of domain names (up to 100) to analyze.
brand	(required) Brand name to search for in AIO texts.
search_engine	(optional) Search engine database. Allowed values: g_us, g_ua. Default: g_us.
page	(optional) Response page number.
page_size	(optional) Response page size. Default: 100.

sort	(optional) Sort field. Allowed values: keyword. Default is NULL.
order	(optional) Sort direction. Allowed values: asc, desc. Default is NULL.
filters	(optional) A list of filtering options. See Filtering for more details.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a paginated list of keywords with AIO data.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

2 credits per row in response.

Filtering

To filter the results you can use `filters` argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, `filters = list(queries_from = 0, queries_to = 10)` would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_aio_brand_opportunities(
  domains      = c('amazon.com'),
  brand        = 'Amazon',
  search_engine = 'g_us',
  page         = 1,
  page_size    = 100,
  return_method = 'df'
)$data

## End(Not run)
```

`sst_sa_domain_aio_summary`*AI Overview Brand Summary*

Description

Returns a summary of AI Overview (AIO) presence for a given brand and list of domains.

Usage

```
sst_sa_domain_aio_summary(  
    domains,  
    brand,  
    search_engine = "g_us",  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

<code>domains</code>	(required) Vector of domain names (up to 100) to analyze.
<code>brand</code>	(required) Brand name to search for in AIO texts.
<code>search_engine</code>	(optional) Search engine database. Allowed values: <code>g_us</code> , <code>g_ua</code> . Default: <code>g_us</code> .
<code>return_method</code>	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
<code>api_token</code>	(required) Serpstat API token from your profile . Default is <code>Sys.getenv('SERPSTAT_API_TOKEN')</code> .

Value

Returns a summary of AIO brand presence.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

5 credits per request.

Examples

```
## Not run:  
sst_sa_domain_aio_summary(  
    domains      = c('amazon.com'),  
    brand        = 'Amazon',  
    search_engine = 'g_us',  
    return_method = 'df'
```

```

)$data

## End(Not run)

```

```

sst_sa_domain_all_regions_traffic
Traffic in all regions for a domain

```

Description

Returns search traffic info for a domain across all databases. Databases with no keywords are not shown.

Usage

```

sst_sa_domain_all_regions_traffic(
  domain,
  sort = c("traff"),
  order = "desc",
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

domain	(required) Domain name to analyze.
sort	(optional) Vector of sorting fields. Default is "traff".
order	(optional) Order of sorting. Allowed values: asc, desc. Default: desc.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns traffic data for all regions.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per region in response.

Examples

```
## Not run:
sst_sa_domain_all_regions_traffic(
  domain      = 'serpstat.com',
  sort        = c('traff'),
  order       = 'desc',
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_domain_competitors
```

Domain competitors in organic search

Description

Returns organic competitors for the domain with their key metrics. This is implemented using the newer SerpstatDomainProcedure.getOrganicCompetitorsPage API.

Usage

```
sst_sa_domain_competitors(
  domain,
  se,
  sort = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) Domain to get data for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) A field to sort the response. See Sorting for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a list of organic competitors with their SEO metrics.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned page.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Examples

```
## Not run:
sst_sa_domain_competitors(
  domain      = 'serpstat.com',
  se          = 'g_us',
  sort        = list(relevance = 'desc'),
  page        = 1,
  size        = 100,
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_domain_export_positions
      Export domain positions
```

Description

Export a complete report of a domain's keyword positions in search results.

Usage

```
sst_sa_domain_export_positions(
  domain,
  se,
  sort = NULL,
  filters = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) Domain name to analyze.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) A field to sort the response. See Sorting for more details.
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a complete report of domain positions (often in CSV format).

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per row in response.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Filtering

To filter the results you can use filters argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See [API docs](#) for more details. For example, filters = list(queries_from = 0, queries_to = 10) would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_export_positions(
  domain      = 'serpstat.com',
  se          = 'g_us',
  page        = 1,
  size        = 100,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_domain_history *Domain history*

Description

Returns historical metrics for the domain with about two weeks between measurements.

Usage

```
sst_sa_domain_history(  
    domain,  
    se,  
    sort = NULL,  
    filters = NULL,  
    page = 1,  
    size = 100,  
    during_all_time = TRUE,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) Domain to get data for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) A field to sort the response. See Sorting for more details.
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
during_all_time	(optional) TRUE (default) for all the history, FALSE for year-to-date data.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a number of metrics for each date for the domain.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per date in the response.

Sorting

You can sort the response using `sort` argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be `asc` for ascending order and `desc` for descending order. For example, `sort = list(ads = 'desc')` would sort the response by `ads` parameter in descending order.

Filtering

To filter the results you can use `filters` argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, `filters = list(queries_from = 0, queries_to = 10)` would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_history(
  domain      = 'serpstat.com',
  se          = 'g_us',
  sort        = list(date = 'desc'),
  filters      = list(traff_from = 20000),
  page        = 2,
  size        = 10,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_domain_keywords

Domain organic keywords

Description

Returns up to 60 000 organic keywords from selected region for the domain with a number of metrics for each keyword.

Usage

```
sst_sa_domain_keywords(
  domain,
  se,
  url = NULL,
  keywords = NULL,
  minusKeywords = NULL,
  sort = NULL,
  filters = NULL,
  page = 1,
```

```

    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

domain	(required) Domain to get data for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
url	(optional) Get the results for this URL only.
keywords	(optional) A vector of words. Keywords in response will contain these words
minusKeywords	(optional) A vector of words. Keywords in response will not contain these words.
sort	(optional) A field to sort the response. See Sorting for more details.
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a number of metrics for each keyword.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per keyword in response.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Filtering

To filter the results you can use filters argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See [API docs](#) for more details. For example, filters = list(queries_from = 0, queries_to = 10) would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_keywords(
  domain      = 'serpstat.com',
  se          = 'g_us',
  sort        = list(keyword_length = 'desc'),
  url         = 'https://serpstat.com/',
  keywords    = list('google'),
  minusKeywords = list('download'),
  filters     = list(queries_from = 0,
                    queries_to   = 10),
  page        = 2,
  size        = 10,
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_domain_keywords_by_language
      Domain keywords by language
```

Description

Returns keywords which the analyzed domain ranks for, enriched with the detected language of each keyword.

Usage

```
sst_sa_domain_keywords_by_language(
  domain,
  se,
  with_subdomains = TRUE,
  url = NULL,
  keywords = NULL,
  minus_keywords = NULL,
  sort = NULL,
  filters = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) Domain name to get data for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .

with_subdomains	(optional) Retrieve data for subdomains too. Default is TRUE.
url	(optional) Get the results for this URL only.
keywords	(optional) A vector of words. Keywords in response will contain these words.
minus_keywords	(optional) A vector of words. Keywords in response will not contain these words.
sort	(optional) A field to sort the response. See Sorting for more details.
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns keywords with language detection info.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 credit per member of data object array in response.

Sorting

You can sort the response using `sort` argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be `asc` for ascending order and `desc` for descending order. For example, `sort = list(ads = 'desc')` would sort the response by `ads` parameter in descending order.

Filtering

To filter the results you can use `filters` argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, `filters = list(queries_from = 0, queries_to = 10)` would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_keywords_by_language(
  domain      = 'serpstat.com',
  se          = 'g_us',
  page        = 1,
  size        = 100,
```

```

    return_method = 'df'
)$data

## End(Not run)

```

sst_sa_domain_keywords_per_region

Number of keywords for each region for a domain

Description

Returns the number of keywords for each region for a domain. Regions with no keywords are not returned.

Usage

```

sst_sa_domain_keywords_per_region(
  domain,
  sort = "keywords_count",
  order = "desc",
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

domain	(required) A domain name to analyze.
sort	(optional) A field to sort the response. Allowed values: keywords_count, db_name, country_name_en, google_domain. Default: keywords_count.
order	(optional) The order of sorting. Allowed values: asc, desc. Default: desc.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a data frame with number of keywords for each region for a domain.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned region.

Examples

```
## Not run:
sst_sa_domain_keywords_per_region(
  domain      = 'serpstat.com',
  sort        = 'keywords_count',
  order       = 'desc',
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_domain_organic_competitors
```

Domain competitors in organic search

Description

Returns organic competitors for the domain with their key metrics.

Usage

```
sst_sa_domain_organic_competitors(
  domain,
  se,
  sort = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domain	(required) Domain to get data for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) A field to sort the response. See Sorting for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a number of metrics for each organic competitor.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned page.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Filtering

To filter the results you can use filters argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, filters = list(queries_from = 0, queries_to = 10) would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_organic_competitors(
  domain      = 'serpstat.com',
  se          = 'g_us',
  sort        = list(relevance = 'desc'),
  page        = 2,
  size        = 20,
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_domain_top_pages
```

Domain top pages

Description

Returns the number of domain pages with the biggest potential traffic, number of keywords, and Facebook shares.

Usage

```
sst_sa_domain_top_pages(  
    domain,  
    se,  
    sort = NULL,  
    filters = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domain	(required) Domain to get data for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) A field to sort the response. See Sorting for more details.
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns domain top pages with their metrics.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned page.

Sorting

You can sort the response using `sort` argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be `asc` for ascending order and `desc` for descending order. For example, `sort = list(ads = 'desc')` would sort the response by `ads` parameter in descending order.

Filtering

To filter the results you can use `filters` argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, `filters = list(queries_from = 0, queries_to = 10)` would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_top_pages(
  domain      = 'serpstat.com',
  se          = 'g_us',
  sort        = list(organic_keywords = 'desc'),
  filters      = list(url_contain = 'blog'),
  page        = 2,
  size        = 50,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_domain_urls	<i>URLs in domain</i>
--------------------	-----------------------

Description

Returns a list of URLs within the analyzed domain and the number of keywords for each URL.

Usage

```
sst_sa_domain_urls(
  domain,
  se,
  sort = NULL,
  filters = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

<code>domain</code>	(required) Domain name to analyze.
<code>se</code>	(required) Search engine alias (db_name) returned by sst_sa_database_info .
<code>sort</code>	(optional) A field to sort the response. See Sorting for more details.
<code>filters</code>	(optional) A list of filtering options. See Filtering for more details.

page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a list of URLs with metrics.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned URL.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Filtering

To filter the results you can use filters argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, filters = list(queries_from = 0, queries_to = 10) would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domain_urls(
  domain      = 'serpstat.com',
  se          = 'g_us',
  page        = 1,
  size        = 100,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_domains_info *Domains summary*

Description

Returns the number of keywords for each domain in SEO and PPC, online visibility and other metrics.

Usage

```
sst_sa_domains_info(  
    domains,  
    se,  
    sort = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domains	(required) A vector of domain names to analyze.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info
sort	(optional) A field to sort the response. See Sorting for more details.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns aggregated stats for each domain.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per domain in request.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Examples

```
## Not run:
sst_sa_domains_info(
  domains      = c('amazon.com', 'ebay.com'),
  se           = 'g_us',
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_domains_intersection
```

Intersection of domains

Description

Returns common keywords for up to 3 domains.

Usage

```
sst_sa_domains_intersection(
  domains,
  se,
  filters = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domains	(required) Vector of domain names (min 2, max 3) to analyze.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns common keywords for the domains.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned keyword.

Filtering

To filter the results you can use `filters` argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, `filters = list(queries_from = 0, queries_to = 10)` would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domains_intersection(
  domains      = c('amazon.com', 'ebay.com'),
  se           = 'g_us',
  page         = 1,
  size         = 100,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_domains_rating_data

Rating Data for Domains

Description

Returns market research data for a list of specified domains, including their category classification and SEO metrics.

Usage

```
sst_sa_domains_rating_data(
  domains,
  se,
  sort = NULL,
  order = NULL,
  filters = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

domains	(required) Vector of domain names (up to 200) to analyze.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) Sort field name.
order	(optional) Sort order.
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns market research ratings for the domains.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 credit per row in response.

Filtering

To filter the results you can use `filters` argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, `filters = list(queries_from = 0, queries_to = 10)` would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domains_rating_data(
  domains      = c('amazon.com', 'ebay.com'),
  se           = 'g-us',
  page         = 1,
  size         = 100,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_domains_unique_keywords
Unique keywords of domains

Description

Returns unique keywords of one or two domains that a third domain does not rank for.

Usage

```
sst_sa_domains_unique_keywords(  
    domains,  
    minus_domain,  
    se,  
    filters = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

domains	(required) Vector of domain names (min 1, max 2) to analyze.
minus_domain	(required) Domain name with keywords to exclude.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns unique keywords for the domains.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per returned keyword.

Filtering

To filter the results you can use `filters` argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See API docs for more details. For example, `filters = list(queries_from = 0, queries_to = 10)` would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_domains_unique_keywords(
  domains      = c('amazon.com'),
  minus_domain = 'ebay.com',
  se           = 'g_us',
  page         = 1,
  size         = 100,
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_keyword_export_keywords_phrase
      exportKeywordsPhrase
```

Description

`exportKeywordsPhrase`

Usage

```
sst_sa_keyword_export_keywords_phrase(
  keyword,
  search_engine,
  filters = NULL,
  sort = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

`keyword` (required) Keyword to search for
`search_engine` (required)

filters	(optional) Filters for search. Fields are combined using the **AND** logic. Numeric range fields ('_from' and '_to') allow specifying minimum and maximum values. List fields ('_contain' and '_not_contain') specify inclusion or exclusion criteria
sort	(optional) Order of sorting the results in the format: 'field: order' - **field** — field to sort by (any field in response) - **order** — sort direction ('asc' — ascending, 'desc' — descending) Example '{"region_queries_count": "desc"}'
page	(optional) Page number in response
size	(optional) Page number in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sa_keyword_export_suggestions
      exportSuggestions
```

Description

exportSuggestions

Usage

```
sst_sa_keyword_export_suggestions(
  search_engine,
  keyword,
  filters = NULL,
  with_questions = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

search_engine	(required)
keyword	(required) Keyword to search for
filters	(optional) Filters for search. Fields are combined using the **AND** logic. Numeric range fields ('_from' and '_to') allow specifying minimum and maximum values. List fields ('_contain' and '_not_contain') specify inclusion or exclusion criteria
with_questions	(optional) With/without question keywords in report
page	(optional) Page number in response
size	(optional) Number of results per page in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sa_keyword_get_ad_keywords
      getAdKeywords
```

Description

getAdKeywords

Usage

```
sst_sa_keyword_get_ad_keywords(
  keyword,
  search_engine,
  domains,
  minus_keywords = NULL,
  filter = NULL,
  sort = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

keyword	(required) Keyword to search for
search_engine	(required)
domains	(required) Domain names list Example: <code>["apple.com","verizon.com"]</code>
minus_keywords	(optional) List of keywords to exclude from the search
filter	(optional) Filters for search. Fields are combined using the **AND** logic. Numeric range fields (<code>'_from'</code> and <code>'_to'</code>) allow specifying minimum and maximum values. List fields (<code>'_contain'</code> and <code>'_not_contain'</code>) specify inclusion or exclusion criteria
sort	(optional) Order of sorting the results in the format: <code>'field: order'</code> - **field** — field to sort by (any field in <code>*data*</code> section of response) - **order** — sort direction (<code>'asc'</code> — ascending, <code>'desc'</code> — descending) Example <code>'{"region_queries_count": "desc"}'</code>
page	(optional) Page number in response
size	(optional) Number of results per page in response
return_method	(optional) Accepted values are <code>'list'</code> (default) to return data object as list or <code>'df'</code> to return data object as <code>data.frame</code> .
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sa_keyword_get_ads_competitors
      getAdsCompetitors
```

Description

`getAdsCompetitors`

Usage

```
sst_sa_keyword_get_ads_competitors(
  keyword,
  search_engine,
  sort = NULL,
  page = 1,
```

```

    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

keyword	(required) Keyword to search for
search_engine	(required)
sort	(optional) Order of sorting the results in the format: 'field: order' - field — field to sort by (any field in <i>*domain detailed information*</i> section of response) - order — sort direction ('asc' — ascending, 'desc' — descending) Example '{"region_queries_count": "desc"}'
page	(optional) Page number in response
size	(optional) Number of results per page in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```

sst_sa_keyword_get_competitors
    getCompetitors

```

Description

getCompetitors

Usage

```

sst_sa_keyword_get_competitors(
  keyword,
  search_engine,
  filters = NULL,
  sort = NULL,
  order = NULL,
  page = 1,

```

```

    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

keyword	(required) Keyword to search for
search_engine	(required) Search engine short name. Example: "search_engine": "g_us". Refer to full list of [search engines](../serpstat-public-api/ba97ni814ao9p-search-engine-short-names)
filters	(optional) Filters for search. Fields are combined using the **AND** logic. Numeric range fields ('_from' and '_to') allow specifying minimum and maximum values.
sort	(optional) Field to sort by
order	(optional) Sort direction
page	(optional) Page number
size	(optional) Number of results per page
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```

sst_sa_keyword_get_keyword_top
      getKeywordTop

```

Description

`getKeywordTop`

Usage

```

sst_sa_keyword_get_keyword_top(
  keyword,
  search_engine,
  filters = NULL,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)

```

Arguments

keyword	(required) Keyword to search for
search_engine	(required)
filters	(optional) Filters for search. Fields are combined using the **AND** logic. Numeric range fields ('_from' and '_to') allow specifying minimum and maximum values. List fields ('_contain' and '_not_contain') specify inclusion or exclusion criteria Default: '{"top_size": 100}'
size	(optional) Number of results per page in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_sa_keyword_get_suggestions
getSuggestions

Description

getSuggestions

Usage

```
sst_sa_keyword_get_suggestions(  
  keyword,  
  search_engine,  
  filters = NULL,  
  page = 1,  
  size = 100,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

keyword	(required) Keyword to search for
search_engine	(required)
filters	(optional) Filter conditions
page	(optional) Page number in response
size	(optional) Number of results per page in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_sa_keyword_get_top_urls
getTopUrls

Description

getTopUrls

Usage

```
sst_sa_keyword_get_top_urls(  
  keyword,  
  search_engine,  
  sort = NULL,  
  order = NULL,  
  page = 1,  
  page_size = 100,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

keyword	(required) Keyword to search for
search_engine	(required)
sort	(optional) Sorting by parameters(Any field in *urls* section of response)
order	(optional) Sorting order
page	(optional) Page number
page_size	(optional) Number of results per page
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_sa_keyword_top	<i>Top for a keyword</i>
--------------------	--------------------------

Description

Returns a list of results (URLs) from search engine results page (SERP) including organic results, paid results and different types of SERP features.

Usage

```
sst_sa_keyword_top(
  keyword,
  se,
  sort = NULL,
  size = 100,
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

keyword	(required) A keyword to search for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) A field to sort the response. See API docs for more details.
size	(optional) Response page size. Should be $\Rightarrow 10$ and ≤ 100 . Default is 100.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a list with the data about search engine results page for the keyword.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per URL in response.

Examples

```
## Not run:
sst_sa_keyword_top(
  keyword = 'serpstat',
  se      = 'g_us',
  size    = 10
)

## End(Not run)
```

sst_sa_keyword_trends *Keyword trends*

Description

Returns monthly search volume trends for a list of keywords over a specified period (up to 48 months). You can send up to 100 keywords per request.

Usage

```
sst_sa_keyword_trends(
  keywords,
  se,
  months = 48,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

keywords	(required) A vector of keywords to analyze.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
months	(optional) Number of months to get data for (up to 48). Default is 48.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns monthly search volume trends for each keyword.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

10 credits per keyword that returned data. If no keywords returned data — 1 credit is charged for the whole request.

Examples

```
## Not run:
sst_sa_keyword_trends(
  keywords      = c('seo', 'ppc', 'serpstat'),
  se            = 'g_us',
  months        = 12,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_keywords	<i>Phrase match keywords</i>
-----------------	------------------------------

Description

A full-text search to find all the keywords that match the queried term with a number of metrics for each keyword like search volume, CPC and competition level.

Usage

```
sst_sa_keywords(
  keyword,
  se,
  minusKeywords = NULL,
  sort = NULL,
  filters = NULL,
  page = 1,
  size = 100,
  with_intents = TRUE,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

keyword	(required) A keyword to search for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
minusKeywords	(optional) A vector of words. Keywords in response will not contain these words.
sort	(optional) A field to sort the response. See Sorting for more details.
filters	(optional) A list of filtering options. See Filtering for more details.
page	(optional) Response page number if there are many pages in response.
size	(optional) Response page size.
with_intents	(optional) TRUE if keyword intent should be retrieved. This parameter works for g_uu and g_us database only.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a number of metrics for each keyword.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per keyword in response.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Filtering

To filter the results you can use filters argument. It must be a list of named elements. The name of the element must match one of the filtering parameters. See [API docs](#) for more details. For example, filters = list(queries_from = 0, queries_to = 10) would narrow the results to include only the keywords that have a search volume between 0 and 10.

Examples

```
## Not run:
sst_sa_keywords(
  keyword      = 'serpstat',
  se           = 'g_us',
  minusKeywords = c('free'),
  sort         = list(keyword_length = 'asc'),
  page         = 2,
  size         = 10,
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_keywords_info *Keywords summary*

Description

Returns a number of metrics for each keyword like search volume, CPC and competition level.

Usage

```
sst_sa_keywords_info(
  keywords,
  se,
  sort = NULL,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

keywords	(required) A vector of keywords to analyze.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
sort	(optional) A field to sort the response. See Sorting for more details.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a number of metrics for each keyword.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per keyword in request.

Sorting

You can sort the response using sort argument. It must be a list with a single named element. The name of the element must match one of parameters in response. The value of the element must be asc for ascending order and desc for descending order. For example, sort = list(ads = 'desc') would sort the response by ads parameter in descending order.

Examples

```
## Not run:
sst_sa_keywords_info(
  keywords      = c('seo', 'ppc', 'serpstat'),
  se            = 'g_us',
  sort          = list(cost = 'asc'),
  return_method = 'df'
)$data

## End(Not run)
```

```
sst_sa_market_categories
      Market Research Categories
```

Description

Returns a complete list of available market research categories used for domain classification.

Usage

```
sst_sa_market_categories(
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token from **your profile**. Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns a list of category identifiers and names.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0 credits.

Examples

```
## Not run:
sst_sa_market_categories(return_method = 'df')$data

## End(Not run)
```

sst_sa_related_keywords

Related keywords for a keyword

Description

Returns a list of semantically related keywords for a given keyword.

Usage

```
sst_sa_related_keywords(
  keyword,
  se,
  with_intents = FALSE,
  filters = NULL,
  sort = NULL,
  page = 1,
  size = 100,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

keyword	(required) A keyword to search for.
se	(required) Search engine alias (db_name) returned by sst_sa_database_info .
with_intents	(optional) TRUE if keyword intent should be retrieved. This parameter works for g_ua and g_us database only.
filters	(optional) A list of filtering options. See API docs for more details.
sort	(optional) A field to sort the response. See API docs for more details.
page	(optional) Response page number if there are many pages in response.

size	(optional) Response page size.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').

Value

Returns related keywords for a keyword and their metrics.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

1 per keyword in response.

Examples

```
## Not run:
sst_sa_related_keywords(
  keyword      = 'serpstat',
  se           = 'g_us',
  return_method = 'df'
)$data

## End(Not run)
```

sst_sa_stats

Get the number of API rows left

Description

With most API request you spend some amount of API rows. The total amount of API rows available for you is based on your [plan](#). Use this method to control the amount of API rows left.

Usage

```
sst_sa_stats(api_token = Sys.getenv("SERPSTAT_API_TOKEN"))
```

Arguments

api_token	(required) Serpstat API token from your profile . Default is Sys.getenv('SERPSTAT_API_TOKEN').
-----------	--

Value

Returns a number of API rows left. Also returns some additional information about user and [Serpstat plugin](#) limits.

API credits consumption

0

Examples

```
## Not run:  
sst_sa_stats()$summary_info$left_lines  
  
## End(Not run)
```

```
sst_sa_url_get_summary_traffic  
getSummaryTraffic
```

Description

getSummaryTraffic

Usage

```
sst_sa_url_get_summary_traffic(  
  search_engine,  
  domain,  
  url_contains,  
  output_data = NULL,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

search_engine	(required)
domain	(required) The domain for which to retrieve traffic and keyword data
url_contains	(required) Searched part of URL
output_data	(optional) Output data value. Both options enable by default, choose one to spend less credits.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sa_url_get_url_competitors  
    getUrlCompetitors
```

Description

getUrlCompetitors

Usage

```
sst_sa_url_get_url_competitors(  
    search_engine,  
    url,  
    sort = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

search_engine	(required)
url	(required) URL for finding competing URLs
sort	(optional) Order of sorting the results
page	(optional) Page number in response
size	(optional) Number of results per page in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sa_url_get_url_keywords
    getUrlKeywords
```

Description

getUrlKeywords

Usage

```
sst_sa_url_get_url_keywords(
    search_engine,
    url,
    with_intents = NULL,
    sort = NULL,
    filters = NULL,
    page = 1,
    size = 100,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

search_engine	(required)
url	(required) Analyzed page URL
with_intents	(optional) Keyword intent *This parameter works for 'g_ua' and 'g_us' database only*
sort	(optional) Order of sorting the results
filters	(optional) Filters for search. Fields are combined using the "AND" logic. Numeric range fields ('_from' and '_to') allow specifying minimum and maximum values. List fields ('_contain' and '_not_contain') specify inclusion or exclusion criteria
page	(optional) Page number in response
size	(optional) Number of results per page in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sa_url_get_url_missing_keywords  
    getUrlMissingKeywords
```

Description

getUrlMissingKeywords

Usage

```
sst_sa_url_get_url_missing_keywords(  
    url,  
    search_engine,  
    sort = NULL,  
    filters = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

url	(required) Analyzed URL
search_engine	(required)
sort	(optional) Order of sorting the results
filters	(optional) Filters for search. Fields are combined using the "AND" logic. Numeric range fields ('_from' and '_to') allow specifying minimum and maximum values. List fields ('_contain' and '_not_contain') specify inclusion or exclusion criteria
page	(optional) Page number in response
size	(optional) Number of results per page in response
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sc_add_keyword_list
    addKeywordList
```

Description

addKeywordList

Usage

```
sst_sc_add_keyword_list(
    keywords,
    search_engine_id,
    country_id,
    region_id = NULL,
    language_id = NULL,
    type_id = NULL,
    type = NULL,
    pages = NULL,
    return_method = "list",
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

keywords	(required) Keywords with commas for parsing. Example: <code>["samsung, iphone", "nike, adidas"]</code>
search_engine_id	(required) Search engine identifier <code>***I**</code> — Google
country_id	(required)
region_id	(optional)
language_id	(optional)
type_id	(optional)
type	(optional)
pages	(optional)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_sc_add_task	<i>Add crawling task</i>
-----------------	--------------------------

Description

Submits keywords to crawl regular SERP and local results.

Usage

```
sst_sc_add_task(
    keywords,
    country_id,
    se_id = 1,
    region_id = NULL,
    language_id = 1,
    device_type_id = 1,
    results_type = "regular_aio",
    pages = 1,
    api_token = Sys.getenv("SERPSTAT_API_TOKEN"),
    return_method = "list"
)
```

Arguments

keywords	(required) A character vector or comma-separated string of keywords.
country_id	(required) Country identifier (e.g. 23 for United States). Range: 1-247.
se_id	(optional) Search engine ID. Default is 1 (Google).
region_id	(optional) Region or city identifier.
language_id	(optional) Language ID. Default is 1 (English). Range: 1-48.
device_type_id	(optional) Device type ID. Default is 1 (Desktop). Range: 1-2.
results_type	(optional) Type of Google results. Default is "regular_aio". Allowed values: "regular", "local", "regular_aio".
pages	(optional) Number of SERP pages to parse. Default is 1. Range: 1-10.
api_token	(optional) Serpstat API token. Default is Sys.getenv("SERPSTAT_API_TOKEN").
return_method	(optional) Accepted values are "list" (default) to return data object as list or "df" to return data object as data.frame.

Value

Returns the task ID.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

Depends on the number of keywords and pages to parse.

Examples

```
## Not run:
sst_sc_add_task(
  keywords = c("explain the gauss rifle principle",
               "why iphone better than samsung"),
  country_id = 23
)

## End(Not run)
```

sst_sc_get_keyword_serp

getKeywordSerp

Description

getKeywordSerp

Usage

```
sst_sc_get_keyword_serp(
  task_id,
  keyword_id,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

task_id	(required)
keyword_id	(required) Id number of a required keyword for getting a raw SERP (you can get it in the SERP crawling response from the method 'tasks.getTaskResult')
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_sc_get_list	<i>getList</i>
-----------------	----------------

Description

getList

Usage

```
sst_sc_get_list(  
    page = 1,  
    page_size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

- page (optional) Page Number
- page_size (optional) Page size
- return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
- api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sc_get_parsing_balance  
    getParsingBalance
```

Description

getParsingBalance

Usage

```
sst_sc_get_parsing_balance(  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.

api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

```
sst_sc_get_task_result  
    Get crawling task result
```

Description

Retrieves parsing results of a crawling task.

Usage

```
sst_sc_get_task_result(  
    task_id,  
    page = 1,  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN"),  
    return_method = "list"  
)
```

Arguments

task_id	(required) Crawling identifier.
page	(optional) Page number for pagination. Results are grouped in pages of 100 keywords. Default is 1.
api_token	(optional) Serpstat API token. Default is Sys.getenv("SERPSTAT_API_TOKEN").
return_method	(optional) Accepted values are "list" (default) to return data object as list or "df" to return data object as data.frame.

Value

Returns the parsing results for the specified task.

API docs

Check all the values for request and response fields [here](#).

API credits consumption

0

Examples

```
## Not run:
sst_sc_get_task_result(
  task_id = 5484945
)

## End(Not run)
```

sst_tm_activate_user *activateUser*

Description

activateUser

Usage

```
sst_tm_activate_user(
  user_id,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

user_id	(required) User identifier at Serpstat.
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_tm_add_user	<i>addUser</i>
-----------------	----------------

Description

addUser

Usage

```
sst_tm_add_user(  
  email,  
  return_method = "list",  
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

email	(required) user email
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_tm_deactivate_user	<i>deactivateUser</i>
------------------------	-----------------------

Description

deactivateUser

Usage

```
sst_tm_deactivate_user(  
    user_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

user_id	(required) User identifier at Serpstat. Can be obtained via ‘TeamManagement.getList’ method
return_method	(optional) Accepted values are ‘list’ (default) to return data object as list or ‘df’ to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_tm_get_list	<i>getList</i>
-----------------	----------------

Description

getList

Usage

```
sst_tm_get_list(  
    search = NULL,  
    page = 1,  
    size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

search	(optional) Optional part of email of member you need to find
page	(optional) page number for pagination
size	(optional) records per page
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_tm_remove_user	<i>removeUser</i>
--------------------	-------------------

Description

removeUser

Usage

```
sst_tm_remove_user(
  user_id,
  merge_projects,
  return_method = "list",
  api_token = Sys.getenv("SERPSTAT_API_TOKEN")
)
```

Arguments

user_id	(required) User id at Serpstat. Can be found using 'TeamManagement.getList' method.
merge_projects	(required) 'true' - transfer user's projects to the team owner. 'false' - projects will be deleted
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_vc_add_keyword_list_freq
<i>addKeywordListFreq</i>

Description

addKeywordListFreq

Usage

```
sst_vc_add_keyword_list_freq(  
    keywords,  
    region_id,  
    type_id = NULL,  
    search_engine_id = NULL,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

- keywords (required) Keywords to get metrics (max 50000)
- region_id (required)
- type_id (optional)
- search_engine_id (optional)
- return_method (optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
- api_token (required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_vc_get_task_result
getTaskResult

Description

getTaskResult

Usage

```
sst_vc_get_task_result(  
    task_id,  
    page = 1,  
    page_size = 100,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

task_id	(required)
page	(optional) Page number
page_size	(optional) Number of results per a page
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

sst_vc_get_task_status
getTaskStatus

Description

getTaskStatus

Usage

```
sst_vc_get_task_status(  
    task_id,  
    return_method = "list",  
    api_token = Sys.getenv("SERPSTAT_API_TOKEN")  
)
```

Arguments

task_id	(required)
return_method	(optional) Accepted values are 'list' (default) to return data object as list or 'df' to return data object as data.frame.
api_token	(required) Serpstat API token.

Value

Returns the API response.

API docs

Check all the values for request and response fields [here](#).

Index

sst_au_export, [5](#)
sst_au_get_audit, [5](#)
sst_au_get_categories_statistic, [6](#)
sst_au_get_default_settings, [7](#)
sst_au_get_error_elements, [8](#)
sst_au_get_error_rows, [9](#)
sst_au_get_history_by_count_error, [10](#)
sst_au_get_list, [11](#)
sst_au_get_page_audit, [11](#)
sst_au_get_pages_list, [12](#)
sst_au_get_report_without_details, [13](#)
sst_au_get_reports_list_by_page, [14](#)
sst_au_get_scan_user_url_list, [14](#)
sst_au_get_settings, [15](#)
sst_au_get_sub_elements_by_crc, [16](#)
sst_au_get_summary, [17](#)
sst_au_page_names, [18](#)
sst_au_remove, [19](#)
sst_au_rescan, [19](#)
sst_au_scan, [20](#)
sst_au_set_settings, [21](#)
sst_au_start, [22](#)
sst_au_stop_page, [23](#)
sst_au_stop_site, [24](#)
sst_au_user_log, [25](#)
sst_bl_anchors, [25](#)
sst_bl_changes_history, [27](#)
sst_bl_domain_summary, [28](#)
sst_bl_domains_intersection, [29](#)
sst_bl_domains_intersection_summary, [30](#)
sst_bl_lost_backlinks, [31](#)
sst_bl_lost_outlinks, [32](#)
sst_bl_new_backlinks, [33](#)
sst_bl_out_domains, [34](#)
sst_bl_out_threats, [35](#)
sst_bl_out_threats_links, [36](#)
sst_bl_outlinks, [37](#)
sst_bl_redirected_domains, [38](#)
sst_bl_referring_domains, [39](#)
sst_bl_sdr_distribution, [41](#)
sst_bl_threats, [42](#)
sst_bl_threats_links, [43](#)
sst_bl_tld_distribution, [44](#)
sst_bl_top_anchors, [45](#)
sst_bl_top_pages, [46](#)
sst_call_api_method, [47](#), [53](#)
sst_cr_get_audit_stats, [47](#)
sst_dc_add_task, [48](#)
sst_dc_get_task, [48](#)
sst_dc_get_task_list, [49](#)
sst_lists_to_df, [50](#)
sst_pm_create_project, [50](#)
sst_pm_delete_project, [51](#)
sst_pm_list_projects, [52](#)
sst_return_check, [53](#)
sst_rt_add_project_competitor, [54](#)
sst_rt_add_project_keywords, [55](#)
sst_rt_add_project_mirror, [56](#)
sst_rt_add_project_region, [56](#)
sst_rt_add_rt_project, [57](#)
sst_rt_competitors, [58](#)
sst_rt_create_and_run_project_tracking, [60](#)
sst_rt_create_project_tags, [61](#)
sst_rt_delete_project_competitor, [61](#)
sst_rt_delete_project_keywords, [62](#)
sst_rt_delete_project_mirror, [63](#)
sst_rt_delete_project_region, [64](#)
sst_rt_delete_project_tags, [64](#)
sst_rt_get_cities, [65](#)
sst_rt_get_countries, [66](#)
sst_rt_get_districts, [67](#)
sst_rt_get_project_competitors, [67](#)
sst_rt_get_project_keywords, [68](#)
sst_rt_get_project_mirrors, [69](#)
sst_rt_get_project_positions, [70](#)
sst_rt_get_project_regions, [71](#)

sst_rt_get_project_status, 72
sst_rt_get_project_tags, 72
sst_rt_get_projects, 73
sst_rt_get_rt_project, 74
sst_rt_get_rt_schedule, 75
sst_rt_positions_history, 75
sst_rt_project_regions, 58, 76, 77, 80
sst_rt_run_project_tracking, 78
sst_rt_serp_history, 79
sst_rt_set_rt_schedule, 81
sst_rt_update_project_keywords, 82
sst_sa_category_top_domains, 83
sst_sa_database_info, 83, 84, 85, 87, 92, 94, 95, 97, 98, 101, 103, 104, 106, 107, 109, 110, 119, 120, 122, 123, 125
sst_sa_domain_ad_competitors, 85
sst_sa_domain_ad_keywords, 86
sst_sa_domain_aio_brand_opportunities, 88
sst_sa_domain_aio_summary, 90
sst_sa_domain_all_regions_traffic, 91
sst_sa_domain_competitors, 92
sst_sa_domain_export_positions, 93
sst_sa_domain_history, 95
sst_sa_domain_keywords, 96
sst_sa_domain_keywords_by_language, 98
sst_sa_domain_keywords_per_region, 100
sst_sa_domain_organic_competitors, 101
sst_sa_domain_top_pages, 102
sst_sa_domain_urls, 104
sst_sa_domains_info, 106
sst_sa_domains_intersection, 107
sst_sa_domains_rating_data, 108
sst_sa_domains_unique_keywords, 110
sst_sa_keyword_export_keywords_phrase, 111
sst_sa_keyword_export_suggestions, 112
sst_sa_keyword_get_ad_keywords, 113
sst_sa_keyword_get_ads_competitors, 114
sst_sa_keyword_get_competitors, 115
sst_sa_keyword_get_keyword_top, 116
sst_sa_keyword_get_suggestions, 117
sst_sa_keyword_get_top_urls, 118
sst_sa_keyword_top, 119
sst_sa_keyword_trends, 120
sst_sa_keywords, 121
sst_sa_keywords_info, 123
sst_sa_market_categories, 124
sst_sa_related_keywords, 125
sst_sa_stats, 126
sst_sa_url_get_summary_traffic, 127
sst_sa_url_get_url_competitors, 128
sst_sa_url_get_url_keywords, 129
sst_sa_url_get_url_missing_keywords, 130
sst_sc_add_keyword_list, 131
sst_sc_add_task, 132
sst_sc_get_keyword_serp, 133
sst_sc_get_list, 134
sst_sc_get_parsing_balance, 135
sst_sc_get_task_result, 135
sst_tm_activate_user, 136
sst_tm_add_user, 137
sst_tm_deactivate_user, 138
sst_tm_get_list, 138
sst_tm_remove_user, 139
sst_vc_add_keyword_list_freq, 140
sst_vc_get_task_result, 141
sst_vc_get_task_status, 141