B-Translator API Reference

Table of Contents

report

statistics

General statistics (number of votes and translations) for the last week, month and year.

- **GET api/report/statistics**
  - **lng**

    language of translations
  - **origin**

    (optional) origin of the project
  - **project**

    (optional) name of the project

- **POST api/report/statistics**
  - **lng**

    language of translations
  - **origin**

    (optional) origin of the project
  - **project**

    (optional) name of the project

Examples:

- http://dev.btranslator.org/api-examples-php/report/statistics.php
- http://info.btranslator.org/api-examples-js/#report/statistics_get.js
- http://info.btranslator.org/api-examples-js/#report/statistics_post.js

```
curl 'https://btranslator.org/api/report/statistics?lng=sq&origi
n=LibreOffice&project=cui'
curl -H 'Accept: application/json' 'https://btranslator.org/api/
report/statistics?lng=sq'
curl -H 'Content-type: application/json' -X POST https://btransl
ator.org/api/report/statistics -d '{"lng": "sq"}'
```

project_stats

Getting project details and statistics (number of translated and untranslated strings, number of votes and translations, users and contributers, etc.)

- **GET api/report/project_stats**
  - **origin**

    origin of the project
  - **project**

    name of the project
  - **lng**

    language of translations

- **POST api/report/project_stats**
  - **origin**

    origin of the project
  - **project**

    name of the project
  - **lng**

    language of translations

Examples:

- http://dev.btranslator.org/api-examples-php/report/project_stats.php
- http://info.btranslator.org/api-examples-js/#report/project_stats_get.js
- http://info.btranslator.org/api-examples-js/#report/project_stats_post.js

```
curl 'https://btranslator.org/api/report/project_stats?origin=Li
breOffice&project=cui&lng=sq'
curl -H 'Accept: application/json' 'https://btranslator.org/api/
report/project_stats?origin=LibreOffice&project=cui&lng=sq'
curl -H 'Content-type: application/json' -X POST https://btransl
ator.org/api/report/project_stats -d '{"origin": "LibreOffice",
"project": "cui", "lng": "sq"}'
```

topcontrib

Statistics about top contributors.

- **GET api/report/topcontrib**
    - **lng**
        language of translations
    - **period**
        period of report (last `day` | `week` | `month` | `year`)
    - **size**
        number of results (top contributers), between 5 and 100
    - **origin**
        (optional) origin of the project
    - **project**
        (optional) name of the project

- **POST api/report/topcontrib**
    - **lng**
        language of translations
    - **period**
        period of report (last `day` | `week` | `month` | `year`)
    - **size**
        number of results (top contributers), between 5 and 100
    - **origin**
        (optional) origin of the project
    - **project**
        (optional) name of the project
```

Examples:

- http://dev.btranslator.org/api-examples-php/report/topcontrib.php
- http://info.btranslator.org/api-examples-js/#report/topcontrib_get.js
- http://info.btranslator.org/api-examples-js/#report/topcontrib_post.js

```
curl -H 'Accept: application/json' 'https://btranslator.org/api/
report/topcontrib?lng=sq&period=week'
curl -H 'Accept: application/json' -H 'Content-type: application
/json' -X POST https://btranslator.org/api/report/topcontrib -d
'{"lng": "sq", "period": "week", "size": 10}' -H 'Accept: applic
ation/json'
```

translations

get_random_sguid

Get the sguid of a random string according to the target.

- **POST api/translations/get_random_sguid**
  - **target**

    Can be one of: `random | translated | untranslated`
  - **lng**

    Code of the language (required for translated and untranslated)
  - **scope**

    (optional) Restricts the pool of selection to the given list of project. It is a space separated list of origin/project like this:

    `origin1/project1 origin2 origin1/project2`

Examples:

- http://dev.btranslator.org/api-examples-php/translations/sguid.php
- http://info.btranslator.org/api-examples-js/#translations/get_random_sguid.js

get

Get a string, along with its translations, votes and other details.

- **GET api/translations/$sguid**
  - **sguid**

    The string (globally unique) id, calculated as:

    `$sguid = sha1($string . $context)` It can also be:

    `random | translated | untranslated`.
  - **lng**

    The language of translations, like fr, de, etc.

- **POST api/translations/get**
  - **sguid**

    The string (globally unique) id, calculated as:

    `$sguid = sha1($string . $context)` It can also be:

    `random | translated | untranslated`.
  - **lng**

    The language of translations, like fr, de, etc.

- **GET translations/$lng/$sguid**

  Returns the details of a string and its translations. If the request header `Accept: application/json` is set, then the output will be in JSON (otherwise it will be an HTML page). The parameter $sguid can also be one of `random | translated | untranslated`.

  Examples:

  - http://info.btranslator.org/api-examples-js/#translations/retrieve.js
  - http://dev.btranslator.org/api-examples-php/translations/get.php
  - http://info.btranslator.org/api-examples-js/#translations/get.js

```
curl -k -H 'Accept: application/json' \
  'http://dev.btranslator.org/translations/sq/ee178b82f7ff3b5fb4
8537b834db673b42d48556'
curl -k -H 'Accept: application/json' http://dev.btranslator.org
/translations/sq/random
```

add

Add a new translation to a source string.

- **POST api/translations/add**
  - **sguid**

    The string ID for which a new translation should be added.
  - **lng**

    The language (code) of the new translation.
  - **translation**

    The new translation as a string. If the string has plural version(s) as well, they are concatenated with NULL bytes ("\0") between them.

  Examples:

  - http://dev.btranslator.org/api-examples-php/translations/add.php
  - http://info.btranslator.org/api-examples-js/#translations/add.js

vote

Add a vote for the given translation.

- **POST api/translations/vote**
  - **tguid**

    ID of the translation

  Examples:

  - http://dev.btranslator.org/api-examples-php/translations/vote.php
  - http://info.btranslator.org/api-examples-js/#translations/vote.js

del

Delete a translation

- **POST api/translations/del**
  - **tguid**

    ID of the translation

  Examples:

  - http://dev.btranslator.org/api-examples-php/translations/add.php
  - http://info.btranslator.org/api-examples-js/#translations/add.js

del_vote

Delete a vote from the given translation.

- **POST api/translations/del_vote**
  - **tguid**

    ID of the translation

Examples:

- http://dev.btranslator.org/api-examples-php/translations/vote.php
- http://info.btranslator.org/api-examples-js/#translations/vote.js

search

Search strings and translations using various filters.

- **GET api/translations**

  This provides the same functionality and has the same parameters as the search operation. The only difference is that the parameters are provided as a GET request (in the URL), whereas the parameters of 'search' are provided as a POST request. Sometimes this may be more convenient.

- **POST api/translations/search**
  - **lng**

    The language of translations.
  - **limit**

    The number of results to be returned (min: 5, max: 50).
  - **words**

    Words to be searched for.
  - **sguid**

    If the words is empty and there is a sguid then the words of this string are used for searching.
  - **mode**

    Search mode. Can be one of:
    - natural-strings : Natural search on strings (default).
    - natural-translations : Natural search on translations.
    - boolean-strings : Boolean search on strings.
    - boolean-translations : Boolean search on translations.

- **page**

  Page of results to be displayed.

- **project**

  Limit search only to this project

- **origin**

  Limit search only to the projects of this origin.

- **only_mine**

  (boolean) Limit search only to the strings touched (translated or voted) by the current user.

- **translated_by**

  Limit search by the author of translations (can be used only by admins).

- **voted_by**

  Limit search by a voter (can be used only by admins).

- **date_filter**

  Which date to filter (used only by admins). Can be one of:

  - strings : Filter Strings By Date
  - translations : Filter Translations By Date (default)
  - votes : Filter Votes By Date

- **from_date**

  Date and time.

- **to_date**

  Date and time.

- **list_mode**

  Used when listing the strings of a project. Can be one of: all, translated, untranslated

- **GET translations/search**

  Search strings and translations using various filters. If the request header `Accept: application/json` is set, then the output will be in JSON (otherwise it will be an HTML page). It gets the same parameters as `GET api/translations`

  Examples:

  - http://dev.btranslator.org/api-examples-php/translations/search.php

- http://info.btranslator.org/api-examples-js/#translations/list.js
- http://info.btranslator.org/api-examples-js/#translations/search.js

```
curl -k -H 'Accept: application/json' \
    'http://dev.btranslator.org/translations/search?lng=sq&word
s=space'
curl -k -H 'Accept: application/json' \
    'https://l10n.org.al/translations/search?lng=sq&words=space
'
```

submit

Submit multiple actions at once.

- **POST api/translations/submit**

  Array of actions. Each action is an associated array that contains the items:
  - **action**

    The action to be done: add | vote | del | del_vote
  - **params**

    Associative array of the needed parameters for this action.

  Examples:

  - http://dev.btranslator.org/api-examples-php/translations/submit.php
  - http://info.btranslator.org/api-examples-js/#translations/submit.js

import

Import translations from PO files.

- **POST api/translations/import**
  - **lng**

    Language of translations.
  - **file**

    A PO file with translations, or an archive of PO files (accepted extensions: .tar, .gz, .tgz, .bz2, .xz, .7z, .zip).

    Returns associative array containing:

- **messages**

  Array of notification messages; each notification message is an array of a message and a type, where type can be one of `status`, `warning`, `error`.

This is like a bulk translation and voting service. For any translation in the PO files, it will be added as a suggestion if such a translation does not exist, or it will just be voted if such a translation already exists. In case that the translation already exists but its author is not known, then you (the user who makes the import) will be recorded as the author of the translation.

This can be useful for translators if they prefer to work off-line with PO files. They can export the PO files of a project, work on them with desktop tools (like Lokalize) to translate or correct exported translations, and then import back to B-Translator the translated/corrected PO files.

Examples:

- http://dev.btranslator.org/api-examples-php/translations/import.php
- http://info.btranslator.org/api-examples-js/#translations/import.js

project

subscribe

Subscribe user to a project.

- **POST api/project/subscribe**
  - **origin**

    Origin of the project.
  - **project**

    Name of the project.

Unsubscribe user from a project.

- **POST api/project/unsubscribe**
  - **origin**

    Origin of the project.
  - **project**

Name of the project.

Get user subscriptions.

- **POST api/project/subscriptions**

Examples:

- http://dev.btranslator.org/api-examples-php/project/subscribe.php
- http://info.btranslator.org/api-examples-js/#project/subscribe.js

list/all

- **GET project/list/all**

  Return a full list of all the imported projects and languages.

Examples:

- http://dev.btranslator.org/api-examples-php/project/list_all.php
- http://info.btranslator.org/api-examples-js/#project/list_all.js

```
curl -s https://btranslator.org/project/list/all | python -m jso
n.tool
```

list

Get a list of existing projects, filtered by origin/project.

- **POST api/project/list**
  - **origin**

    Filter by origin.
  - **project**

    Filter by project.

    Parameters origin and project can contain *, which is replaced by % (for LIKE matches).

    If project is - then only a list of 'origin' is outputed, otherwise a list of 'origin/project'.

- **GET project/list/$origin/$project[/$format]**

Return a plain-text list of all the imported projects, filtered by the given origin/project. Parameters origin and project can contain *, which is replaced by % (for LIKE matches). If project is - then only a list of 'origin' is outputed, otherwise a list of 'origin/project'. The third variable can be JSON or TEXT (default).

Examples:

- http://dev.btranslator.org/api-examples-php/project/list.php
- http://info.btranslator.org/api-examples-js/#project/list.js

```
curl -k -X POST http://dev.btranslator.org/api/project/list
curl -k -X POST http://dev.btranslator.org/api/project/list \
    -H "Content-type: application/json" \
    -d '{"origin": "t*"}'
```

```
curl 'https://btranslator.org/project/list/*/-'
curl 'https://btranslator.org/project/list/*/-/json'
curl 'https://btranslator.org/project/list/LibreOffice/s*'
curl 'https://btranslator.org/project/list/LibreOffice'
curl 'https://btranslator.org/project/list/*/nautil*'
curl 'https://btranslator.org/project/list/*/*/json'
curl 'https://btranslator.org/project/list'
```

langs

- **GET project/langs/$origin/$project**

    Get a list of languages for the given origin/project.

Examples:

```
curl -s https://btranslator.org/project/langs/LibreOffice/cui |
python -m json.tool
```

export

- **POST api/project/export**

Schedule a project for export. When the request is completed, the user will be notified by email. Accepts these parameters:

- **origin**

  The origin of the project.

- **project**

  The name of the project.

- **lng**

  Translation to be exported.

- **export_mode**

  The export mode that should be used. It can be one of:

  `original | most_voted | preferred`.

  - The mode `original` exports the translations of the original files that were imported.
  - The mode `most_voted` exports the translations with the highest number of votes.
  - The mode `preferred` gives precedence to the translations voted by a user or a list of users, despite the number of votes.

- **preferred_voters**

  Comma separated list of usernames. Used only when export_mode is `preferred`.

- **GET/POST project/export**

  Return an archive of PO files for a given origin/project/lng. Does not allow concurrent exports because they may affect the performance of the server. Parameters are received from the request (either GET or POST). They are:

  - **origin**

    Origin of the project.

  - **project**

    The name of the project.

  - **lng**

    Translation to be exported.

  - **export_mode**

    Mode of export: `most_voted | preferred | original`

    Default is `most_voted` which exports the most voted translations and

suggestions.

The export mode `original` exports the translations of the original file that was imported (useful for making diffs).

The export mode `preferred` gives priority to translations that are voted by a certain user or a group of users. It requires an additional argument (preferred_voters) to specify the user (or a list of users) whose translations are preferred.

- **preferred_voters**
  Comma separated list of usernames of the preferred voters.

Examples:

- http://dev.btranslator.org/api-examples-php/project/export.php
- http://info.btranslator.org/api-examples-js/#project/export.js

```
curl -k 'http://dev.btranslator.org/project/export?origin=test&p
roject=kturtle&lng=sq' > test1.tgz
curl -k http://dev.btranslator.org/project/export \
    -d 'origin=test&project=kturtle&lng=sq&export_mode=original
' > test1.tgz
curl -k http://dev.btranslator.org/project/export \
    -d origin=test -d project=kturtle -d lng=sq \
    -d export_mode=preferred \
    -d preferred_voters=user1,user2 > test1.tgz
```

diff

For more details about diffs see: README.org#snapshots-and-diffs

- **GET project/diff/$origin/$project/$lng[/$nr[/$ediff]]**
  Return as plain-text the diff of the PO file for a given origin/project/lng, which contains the latest most-voted suggestions since the last snapshot. If the parameter $nr is missing, it returns a list of the saved diffs instead.
- **GET project/diff-dl/$origin/$project/$lng[/$nr[/$ediff]]**
  Similar to project_diff, but return a file for download (not plain text).

- **GET project/diff-latest/$origin/$project/$lng?format=…&export_mode=…&friends=…**

    Export the latest most voted translations and make the diffs with the last snapshot.

    Examples:

    ```
    curl https://btranslator.org/project/diff/LibreOffice/sw/sq
    curl https://btranslator.org/project/diff/LibreOffice/sw/sq/2 >
    2.diff
    curl https://btranslator.org/project/diff/LibreOffice/sw/sq/2/ed
    iff > 2.ediff
    curl https://btranslator.org/project/diff/LibreOffice/sw/sq/4 >
    4.diff
    curl https://btranslator.org/project/diff/LibreOffice/sw/sq/4/ed
    iff > 4.ediff
    ```

    ```
    curl https://btranslator.org/project/diff-dl/LibreOffice/sw/sq/2
    curl https://btranslator.org/project/diff-dl/LibreOffice/sw/sq/2
    /ediff
    curl https://btranslator.org/project/diff-dl/LibreOffice/sw/sq/4
    /diff
    curl https://btranslator.org/project/diff-dl/LibreOffice/sw/sq/4
    /ediff
    ```

    ```
    curl 'https://btranslator.org/project/diff-latest/LibreOffice/sw
    /sq?format=diff&export_mode=preferred_by_me&friends='
    ```

import

Create a project or update an existing one by importing the POT and PO files of the project.

- **POST api/project/import**
    - **origin**

The origin of the project that will be imported.

- **project**

  The name of the project that will be imported.

- **file**

  Archive of POT/PO files (tgz, bz2, 7z, zip).

Returns associative array containing:

- **messages**

  Array of notification messages; each notification message is an array of a message and a type, where type can be one of `status`, `warning`, `error`.

The structure of the uploaded archive file:

- Acceptable archive types: tgz, bz2, 7z, zip
- The POT files should be placed in the subfolder `pot/`, directly in the root folder of the archive.
- The PO files of a language should be placed in the subfolder named after the code of that language (for example: `sq/`, `de/`, `fr/`, etc.)
- The path and name of each PO file should be the same as the corresponding POT file, except for the extension (`.po` instead of `.pot`).

Examples:

- http://dev.btranslator.org/api-examples-php/project/import.php
- http://info.btranslator.org/api-examples-js/#project/import.js

delete

Delete a project.

- **POST api/project/delete**
  - **origin**

    The origin of the project that will be deleted.
  - **project**

    The name of the project that will be deleted.

Returns associative array containing:

- **message**

Notification message, which is an array of a message and a type, where type can be one of `status`, `warning`, `error`.

Examples:

- http://dev.btranslator.org/api-examples-php/project/delete.php
- http://info.btranslator.org/api-examples-js/#project/delete.js

vocabulary

Vocabularies are a special kind of projects. So all the functions that work for projects can be used for vocabularies as well. For example, if we have the vocabulary named `test1` of language `sq`, then project functions can be used with =origin = 'vocabulary'= and

`project = 'test1_sq'=`. Also, the context of each string will be =test1_sq .

list

- **GET vocabulary/list**

  Return a list of vocabularies in JSON format.

Examples:

- http://dev.btranslator.org/api-examples-php/vocabulary/list.php
- http://info.btranslator.org/api-examples-js/#vocabulary/list.js

```
curl -s https://btranslator.org/vocabulary/list | python -m json
.tool
curl -s http://dev.btranslator.org/vocabulary/list | python -m j
son.tool
```

export

- **GET vocabulary/export/$vocabulary[/$format]**

  Return the list of terms and translations of a vocabulary.
  - **$vocabulary**

    The name of the vocabulary project (which ends in `_lng`).
  - **$format**

The output format: `json | txt1 | txt2 | org`. Default is `json`.

Examples:

- http://dev.btranslator.org/api-examples-php/vocabulary/export.php
- http://info.btranslator.org/api-examples-js/#vocabulary/export.js

```
curl -s https://btranslator.org/vocabulary/export/ICT_sq | pytho
n -m json.tool
curl https://btranslator.org/vocabulary/export/ICT_sq/txt1
curl https://btranslator.org/vocabulary/export/ICT_sq/org > ICT_
sq.org
```

import

Create a vocabulary or update an existing one.

- **POST api/vocabulary/import**
  - **name**

    The name of the vocabulary.
  - **lng**

    The language of the vocabulary.
  - **file**

    File of terms and translations.

Returns associative array containing:

- **messages**

  Array of notification messages; each notification message is an array of a message and a type, where type can be one of `status`, `warning`, `error`.

The format of the uploaded text file is the same as the exported TXT1 file, where each line is like this:

```
term <==> translation1 / translation2 / translation3
```

Examples:

- http://dev.btranslator.org/api-examples-php/vocabulary/import.php
- http://info.btranslator.org/api-examples-js/#vocabulary/import.js

add_string

Add a new string to a vocabulary.

- **POST api/vocabulary/add_string**
  - **name**

    The name of the vocabulary.
  - **lng**

    The language of the vocabulary.
  - **string**

    String to be added.
  - **notify**

    (optional) It TRUE, notify subscribers about the new string. Default is TRUE.

    Returns associative array containing:

  - **sguid**

    The sguid of the new string, or NULL if such a string already exists.
  - **messages**

    Array of notification messages; each notification message is an array of a message and a type, where type can be one of status, warning, error

Examples:

- http://dev.btranslator.org/api-examples-php/vocabulary/add_string.php
- http://info.btranslator.org/api-examples-js/#vocabulary/add_string.js

del_string

Delete a string from a vocabulary.

- **POST api/project/del_string**
  - **name**

    The name of the vocabulary.
  - **lng**

The language of the vocabulary.

- **sguid**

  ID of the string to be deleted.

  Returns associative array containing:

- **messages**

  Array of notification messages; each notification message is an array of a message and a type, where type can be one of status, warning, error

Examples:

- http://dev.btranslator.org/api-examples-php/vocabulary/add_string.php
- http://info.btranslator.org/api-examples-js/#vocabulary/add_string.js

delete

Delete a vocabulary.

- **POST api/vocabulary/delete**
  - **name**

    The name of the vocabulary.
  - **lng**

    The language of the vocabulary.

Returns associative array containing:

- **message**

  Notification message, which is an array of a message and a type, where type can be one of `status`, `warning`, `error`.

Examples:

- http://dev.btranslator.org/api-examples-php/vocabulary/delete.php
- http://info.btranslator.org/api-examples-js/#vocabulary/delete.js

misc

latest

- **GET latest/$lng[/$origin[/$project]]**

  Return a JSON array of the latest translations for the given language (origin and project are optional). Latest means yesterday. Results are cached, so last minute translations may not appear on the list.

  Examples:

  - http://dev.btranslator.org/api-examples-php/misc/latest.php
  - http://info.btranslator.org/api-examples-js/#misc/latest.js

```
curl https://btranslator.org/latest/sq
curl https://btranslator.org/latest/sq/vocabulary
curl https://btranslator.org/latest/sq/vocabulary/ICT_sq
```

rss-feed

- **GET btr/rss-feed[/$origin[/$project]]**

  Return a RSS feed of the latest translations (origin and project are optional). Latest means yesterday. Results are cached, so last minute translations may not appear on the list.

  Examples:

  - http://dev.btranslator.org/api-examples-php/misc/rss-feed.php
  - http://info.btranslator.org/api-examples-js/#misc/rss-feed.js

```
curl https://l10n.org.al/btr/rss-feed
curl https://l10n.org.al/btr/rss-feed/vocabulary
curl https://l10n.org.al/btr/rss-feed/vocabulary/ICT_sq

curl https://btranslator.net/btr/rss-feed/LibreOffice/cui/sq
curl https://btranslator.net/btr/rss-feed/LibreOffice/cui

curl https://btranslator.org/rss-feed/sq
curl https://btranslator.org/rss-feed/sq/LibreOffice
curl https://btranslator.org/rss-feed/sq/LibreOffice/cui
```

twitter

- **GET btr/tweet[/$origin[/$project]]**

  Return a random string and its translations in a suitable form for the twitter (truncated to 100 characters).

- **GET tweet/$lng[/$origin[/$project]]**

  Return a random string and its translations in a suitable form for the twitter (truncated to 100 characters).

  - **lng**

    Language of translations.

Note: This can be used in combination with cli twitter clients and with cron to post random tweets about translations. See this script as an example: https://github.com/B-Translator/btr_server/blob/master/utils/twitter.sh

Examples:

- http://dev.btranslator.org/api-examples-php/misc/twitter.php
- http://info.btranslator.org/api-examples-js/#misc/twitter.js

```
curl https://l10n.org.al/btr/tweet
curl https://l10n.org.al/btr/tweet/vocabulary
curl https://l10n.org.al/btr/tweet/vocabulary/ICT_sq


curl https://btranslator.net/btr/tweet/vocabulary/ICT_sq/sq


curl https://btranslator.org/tweet/sq
curl https://btranslator.org/tweet/sq/vocabulary
curl https://btranslator.org/tweet/sq/vocabulary/ICT_sq


curl -H 'Accept: application/json' \
     https://btranslator.org/tweet/sq | python -m json.tool
```

autocomplete

- **GET auto/project/$str**

  Retrieve a JSON list of autocomplete suggestions for projects.

- - **str**

    Beginning of a project name.

- **GET auto/origin/$str**

    Retrieve a JSON list of autocomplete suggestions for origins of projects.

  - **str**

    Beginning of an origin.

- **GET auto/user/$lng/$str**

    Retrieve a JSON list of autocomplete suggestions for users.

  - **lng**

    Language of translations.

  - **str**

    Beginning of a user name.

- **GET auto/string/$origin/$project/$str**

    Retrieve a JSON list of autocomplete suggestions for strings.

  - **origin**

    Limit search to this origin. If it is *, then don't limit by origin.

  - **project**

    Limit search to this project. If it is *, then don't limit by project.

  - **str**

    Beginning of a string.

Examples:

- http://dev.btranslator.org/api-examples-php/misc/autocomplete.php
- http://info.btranslator.org/api-examples-js/#misc/autocomplete.js

```
curl https://btranslator.org/auto/project/kd
curl https://btranslator.org/auto/origin/G
curl https://btranslator.org/auto/user/sq/d


curl https://btranslator.org/auto/string/vocabulary/ICT_sq/a
curl https://btranslator.org/auto/string/vocabulary/*/a
curl https://btranslator.org/auto/string/*/ICT_sq/a
curl https://btranslator.org/auto/string/*/*/a
```

other

- **GET languages**

    Return a json list of languages installed on the system and their details.
- **GET string_details/$sguid**

    Return string context, a list of related projects (where the string appears), etc. provider.

    - **sguid**

        Id of the string.

    Examples:

    ```
    curl https://btranslator.org/languages | python -m json.tool
    curl https://btranslator.org/string_details/0f27be1118ae4807f00e
    d5fa21a0f17fcc3649ed | python -m json.tool
    ```

oauth2

- **POST oauth2/token**

    Get an OAuth2 token.

    ```
    $.ajax(base_url + '/oauth2/token', {
        type: 'POST',
        data: {
            grant_type: 'password',
            username: 'user1',
            password: 'pass1',
            scope: 'user_profile',
        },
        headers: {
            'Authorization': 'Basic ' + btoa(client_id + ':' + client
    _secret'),
        },

    });
    ```

- **GET oauth2/tokens/<access_token>**

Verify the validity of an OAuth2 token.

```javascript
$.ajax(base_url + '/oauth2/tokens/' + access_token)
    .fail(function () {
        console.log('Token is not valid.');
    })
    .done(function (response) {
        console.log(response);
    });
```

- **POST oauth2/user/profile**

  Get the profile of the user.

```javascript
$.ajax(base_url + '/oauth2/user/profile', {
    type: 'POST',
    headers: {
        'Authorization': 'Bearer ' + access_token,
    },
    dataType: 'json',
});
```

Examples:

- http://info.btranslator.org/api-examples-js/#oauth2/get_token.js
- http://info.btranslator.org/api-examples-js/#oauth2/verify_token.js
- http://info.btranslator.org/api-examples-js/#oauth2/get_user_profile.js

For more details and examples see: https://github.com/dashohoxha/oauth2-js

For PHP examples see:

- https://btranslator.org/api-examples-php/config.php
- https://btranslator.org/api-examples-php/get_access_token.php
- https://btranslator.org/api-examples-php/translations/add.php