LanguageData Documentation

Wikimedia Foundation

Sep 05, 2022

User Documentation

This library contains language related data, and utility libraries written in PHP and Node.js to interact with that data.

The language related data comprises of the following,

1. The script in which a language is written
2. The script code
3. The language code
4. The regions in which the language is spoken
5. The autonym - language name written in its own script
6. The directionality of the text

This data is populated from the current version of CLDR supplemental data and various other sources.

CHAPTER 1

Using the PHP library

1.1 Installation

You can add this library to your project by running:

```
composer install wikimedia/language-data
```

1.2 Basic usage

The basic usage is like this:

```php
<?php
use Wikimedia\LanguageData\LanguageUtil;

$languageUtil = LanguageUtil::get();
// Returns English
$languageUtil->getAutonym( 'en' );
```

For a full list of methods see the documentation for the LanguageUtil class.

CHAPTER 2

Using the Node.js library

2.1 Installation

You can add this library to your project by running,

```
npm i @wikimedia/language-data
```

2.2 Basic usage

The basic usage is like this:

```
const languageData = require('@wikimedia/language-data');

// Returns English
languageData.getAutonym( 'en');
```

The exposed methods are similar to the methods present in the PHP LanguageUtil class.

CHAPTER 3

Changelog

The full changelog is available here.

CHAPTER 4

Contribute

- Issue Tracker: https://github.com/wikimedia/language-data/issues
- Source Code: https://github.com/wikimedia/language-data

CHAPTER 5

Navigation

5.1 Adding new languages

New languages must be added to the `data/langdb.yaml` file.

The file format is: ISO 639 code: `[writing system code, [regions list], autonym]`

The writing system is indicated using ISO 15924 codes. Make sure that the code appears in the `scriptgroups` section towards the end of the file, and add it, if it doesn't.

The list of region codes appears at the end of `data/langdb.yaml`.

The autonym is the name of the language in the language itself. In some cases, for example for extinct languages such as Jewish Babylonian Aramaic (tmr), the name can be something that is useful for modern users, but in most cases it should be the natural name in the language itself. Please do your best to verify that it's spelled correctly in reliable sources.

Example: `myv: [Cyrl, [EU], ]`

This is the Erzya language. Its writing system is Cyrillic (ISO 15924: Cyrl). It's spoken in Europe (EU). Its autonym is "".

Some languages are listed as redirects. In this case, the only value in the square brackets is the target language code.

Example: `fil: [tl]`

This is the Filipino language (fil), which is a redirect to Tagalog (tl).

After adding a language to `data/langdb.yaml`, run: `php src/util/ulsdata2json.php` in the base directory to generate the `language-data.json` file. Don't edit `language-data.json` manually.

Before running `php src/util/ulsdata2json.php` for the first time on your machine, you also need to run `composer install` in the base directory to install some dependencies. You also need to install PHP curl support on your machine to allow downloading the CLDR language data file.

5.2 LanguageUtil

A singleton utility class to query the language data.

Qualified name `Wikimedia\LanguageData\LanguageUtil`

class LanguageUtil

addLanguage (*string $languageCode*, *array $options*)
Adds a language in run time and sets its options as provided. If the target option is provided, the language is defined as a redirect. Other possible options are script (string), regions (array) and autonym (string).

Parameters

- **$languageCode** (*string*) – New language code.
- **$options** (*array*) – Language properties.

getAutonym (*string $languageCode*)
Returns the autonym of the language

Parameters

- **$languageCode** (*string*) – Language code

Returns string|bool Autonym of the language or false if the language is unknown

getAutonyms () → array
Returns all language codes and corresponding autonyms

Returns array – The key is the language code, and the values are corresponding autonym

getDir (*string $languageCode*)
Return the direction of the language

Parameters

- **$languageCode** (*string*) – Language code

Returns string|bool Returns 'rtl' or 'ltr'. If the language code is unknown, returns false.

getGroupOfScript (*string $script*) → string
Returns the script group of a script or "Other" if it doesn't belong to any group

Parameters

- **$script** (*string*) – Name of the script

Returns string – Script group name or "Other" if the script doesn't belong to any group

getLanguages ()
Get all the languages. The properties in the returned object are ISO 639 language codes The value of each property is an array that has, [writing system code, [regions list], autonym]

Returns stdClass

getLanguagesByScriptGroup (*array $languageCodes*) → array
Return the list of languages passed, grouped by their script group

Parameters

- **$languageCodes** (*array*) – List of language codes to group

Returns array – List of language codes grouped by script group

getLanguagesByScriptGroupInRegion (*string $region*) → LanguageUtil::getLanguagesByScriptGroupInRegions
Returns an associative array of languages in a region, grouped by their script

Parameters

- **$region** (*string*) – Region code

Returns *LanguageUtil::getLanguagesByScriptGroupInRegions* –

getLanguagesByScriptGroupInRegions (*array $regions*) → array
Returns an associative array of languages in several regions, grouped by script group

Parameters

- **$regions** (*array*) – List of strings representing region codes

Returns array – Returns an associative array. They key is the script group name, and the value is a list of language codes in that region.

getLanguagesInScript (*string $script*) → array
Returns all languages written in the given script

Parameters

- **$script** (*string*) – Name of the script

Returns array –

getLanguagesInScripts (*array $scripts*) → array
Returns all languages written in the given scripts

Parameters

- **$scripts** (*array*) – List of strings, each being the name of a script

Returns array –

getLanguagesInTerritory (*string $territory*)
Returns the languages spoken in a territory

Parameters

- **$territory** (*string*) – Territory code

Returns array|bool List of language codes in the territory, or else false if invalid territory is passed

getRegions (*string $languageCode*)
Returns the regions in which a language is spoken

Parameters

- **$languageCode** (*string*) – Language code

Returns array|bool List of regions or false if language is unknown

getScript (*string $languageCode*)
Returns the script of the language

Parameters

- **$languageCode** (*string*) – Language code

Returns string|bool Language script or false if the language is unknown

getScriptGroupOfLanguage(*string $languageCode*) → string
Returns the script group of a language. Language belongs to a script, and the script belongs to a script group

Parameters

- **$languageCode** (*string*) – Language code

Returns string – script group name

isKnown(*string $languageCode*) → bool
Checks if a language code is valid

Parameters

- **$languageCode** (*string*) – Language code

Returns bool –

isRedirect(*string $languageCode*)
Checks if the language is a redirect and returns the target language code

Parameters

- **$languageCode** (*string*) – Language code

Returns string|bool Target language code if it's a redirect or false if it's not

isRtl(*string $languageCode*) → bool
Check if a language is right-to-left

Parameters

- **$languageCode** (*string*) – Language code

Returns bool – true if it is an RTL language, else false. Returns false if an unknown language code is passed.

sortByAutonym(*array $languageCodes*) → array
Sort languages by their autonym

Parameters

- **$languageCodes** (*array*) – List of language codes to sort

Returns array – List of sorted language codes returned by their autonym

sortByScriptGroup(*array $languageCodes*) → array
Return the list of languages sorted by their script groups

Parameters

- **$languageCodes** (*array*) – List of language codes to sort

Returns array – Sorted list of strings containing language codes

static get → self
Returns an instance of the class that can be used to then call the other methods in the class.

Returns self –

CHAPTER 6

Indices and tables

- genindex
- search

Index