

## NAME

Locale::Country - standard codes for country identification

## SYNOPSIS

```
use Locale::Country;

$country = code2country('jp' [,CODESET]);      # $country gets 'Japan'
$code     = country2code('Norway' [,CODESET]);  # $code gets 'no'

@codes    = all_country_codes( [CODESET]);
@names    = all_country_names();

# semi-private routines
Locale::Country::alias_code('uk' => 'gb');
Locale::Country::rename_country('gb' => 'Great Britain');
```

## DESCRIPTION

The `Locale::Country` module provides access to several code sets that can be used for identifying countries, such as those defined in ISO 3166-1.

Most of the routines take an optional additional argument which specifies the code set to use. If not specified, the default ISO 3166-1 two-letter codes will be used.

## SUPPORTED CODE SETS

There are several different code sets you can use for identifying countries. A code set may be specified using either a name, or a constant that is automatically exported by this module.

For example, the two are equivalent:

```
$country = code2country('jp', 'alpha-2');
$country = code2country('jp', LOCALE_CODE_ALPHA_2);
```

The codesets currently supported are:

### **alpha-2, LOCALE\_CODE\_ALPHA\_2**

This is the set of two-letter (lowercase) codes from ISO 3166-1, such as 'tv' for Tuvalu.

This is the default code set.

### **alpha-3, LOCALE\_CODE\_ALPHA\_3**

This is the set of three-letter (lowercase) codes from ISO 3166-1, such as 'brb' for Barbados. These codes are actually defined and maintained by the U.N. Statistics division.

### **numeric, LOCALE\_CODE\_NUMERIC**

This is the set of three-digit numeric codes from ISO 3166-1, such as 064 for Bhutan. These codes are actually defined and maintained by the U.N. Statistics division.

If a 2-digit code is entered, it is converted to 3 digits by prepending a 0.

### **dom, LOCALE\_CODE\_DOM**

The IANA is responsible for delegating management of the top level country domains. The country domains are the two-letter (lowercase) codes from ISO 3166 with a few other additions.

NOTE: As of version 3.27, the FIPS code set is no longer supported. See the `Locale::Codes::Changes` document for details.

## ROUTINES

`code2country ( CODE [,CODESET] [, 'retired'] )`  
`country2code ( NAME [,CODESET] [, 'retired'] )`  
`country_code2code ( CODE ,CODESET ,CODESET2 )`  
`all_country_codes ( [CODESET] [, 'retired'] )`  
`all_country_names ( [CODESET] [, 'retired'] )`  
`Locale::Country::rename_country ( CODE ,NEW_NAME [,CODESET] )`  
`Locale::Country::add_country ( CODE ,NAME [,CODESET] )`  
`Locale::Country::delete_country ( CODE [,CODESET] )`  
`Locale::Country::add_country_alias ( NAME ,NEW_NAME )`  
`Locale::Country::delete_country_alias ( NAME )`  
`Locale::Country::rename_country_code ( CODE ,NEW_CODE [,CODESET] )`  
`Locale::Country::add_country_code_alias ( CODE ,NEW_CODE [,CODESET] )`  
`Locale::Country::delete_country_code_alias ( CODE [,CODESET] )`

These routines are all documented in the `Locale::Codes::API` man page.

### `alias_code ( ALIAS, CODE [,CODESET] )`

Version 2.07 included 2 functions for modifying the internal data: `rename_country` and `alias_code`. Both of these could be used only to modify the internal data for country codes.

As of 3.10, the internal data for all types of codes can be modified.

The `alias_code` function is preserved for backwards compatibility, but the following two are identical:

```
alias_code(ALIAS, CODE [ , CODESET] );  
rename_country_code(CODE, ALIAS [ , CODESET] );
```

and the latter should be used for consistency.

The `alias_code` function is deprecated and will be removed in the December 2014 release.

**Note:** this function was previously called `_alias_code`, but the leading underscore has been dropped. The old name was supported for all 2.X releases, but has been dropped as of 3.00.

## SEE ALSO

### **Locale::Codes**

The Locale-Codes distribution.

### **Locale::Codes::API**

The list of functions supported by this module.

### **Locale::SubCountry**

ISO codes for country sub-divisions (states, counties, provinces, etc), as defined in ISO 3166-2. This module is not part of the Locale-Codes distribution, but is available from CPAN in `CPAN/modules/by-module/Locale/`

[http://www.iso.org/iso/home/standards/country\\_codes.htm](http://www.iso.org/iso/home/standards/country_codes.htm)

Official home page for the ISO 3166 maintenance agency.

[http://www.iso.org/iso/home/standards/country\\_codes/iso-3166-1\\_decoding\\_table.htm](http://www.iso.org/iso/home/standards/country_codes/iso-3166-1_decoding_table.htm)

The source of ISO 3166-1 two-letter codes used by this module.

<http://www.iana.org/domains/root/db/>

Official source of the top-level domain names.

**<http://unstats.un.org/unsd/methods/m49/m49alpha.htm>**

The source of the official ISO 3166-1 three-letter codes and three-digit codes.

For some reason, this table is incomplete! Several countries are missing from it, and I cannot find them anywhere on the UN site. I no longer use this as a source of data.

**[https://www.cia.gov/library/publications/the-world-factbook/appendix/print\\_appendix-d.html](https://www.cia.gov/library/publications/the-world-factbook/appendix/print_appendix-d.html)**

The World Factbook maintained by the CIA is a potential source of the data. Unfortunately, it adds/preserves non-standard codes, so it is no longer used as a source of data.

**<http://www.statoids.com/wab.html>**

Another unofficial source of data. Currently, it is not used to get data, but the notes and explanatory material were very useful for understanding discrepancies between the sources.

## AUTHOR

See Locale::Codes for full author history.

Currently maintained by Sullivan Beck ([sbeck@cpan.org](mailto:sbeck@cpan.org)).

## COPYRIGHT

Copyright (c) 1997-2001 Canon Research Centre Europe (CRE).

Copyright (c) 2001-2010 Neil Bowers

Copyright (c) 2010-2015 Sullivan Beck

This module is free software; you can redistribute it and/or modify it under the same terms as Perl itself.