

Package: naomi.options (via r-universe)

August 26, 2024

Title Contains model and calibration options and helper functions for Naomi

Version 1.2.1

Description Contains model and calibration option JSON and default values and helper functions for Naomi.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.1

URL <https://github.com/mrc-ide/naomi.options>

BugReports <https://github.com/mrc-ide/naomi.options/issues>

Additional_repositories <https://mrc-ide.r-universe.dev>

Imports data.table, glue, jsonlite, traduire (>= 0.0.5),

Suggests jsonvalidate, testthat (>= 3.0.0), mockery

Remotes reside-ic/traduire

Config/testthat/edition 3

Repository <https://mrc-ide.r-universe.dev>

RemoteUrl <https://github.com/mrc-ide/naomi.options>

RemoteRef main

RemoteSha 1dc392d9951a29e48407214f62c0c4c89887741d

Contents

get_calibration_option_labels	2
get_controls_json	2
translate_labels	3

Index	4
--------------	----------

```
get_calibration_option_labels
```

Map calibration option ID to JSON calibration option labels

Description

Map calibration option ID to JSON calibration option labels

Usage

```
get_calibration_option_labels(options)
```

Arguments

options	Key-value (calibration option name - calibration option ID) list of model options to be mapped.
---------	-------------------------------------------------------------------------------------------------

Value

Mapped key-value (calibration option name - calibration option label) list of model options

```
get_controls_json      Get the options JSON
```

Description

This gets the JSON for the model or calibration options for a particular country. The control info will be filled in in this order of precedence

1. override_values arg (if set)
2. hardcoded default values from default_options.csv

Usage

```
get_controls_json(type, iso3, options, override_values, config = list())
```

Arguments

type	The type of options to get, model or calibration
iso3	The iso3 of the country to get options for
options	A named list of options to override those in JSON template. Must be in the format of a list of control options i.e. list(control_id = list(list(id = "1", label = "one"), list(id = "2", label = "two")), ...)

override_values	A list of named values to use as default values. This will take precedent over a hardcoded value if both are set.
config	Additional configuration options for returned JSON. If type is 'model' can set boolean 'include_art' and 'include_anc' to include options for ART and ANC controls. You can also pass additional_control_groups which will be appended to the top of the controls JSON. additional_control_groups must be a list representing a valid control group object. No affect for 'calibration' type.

Value

The complete controls JSON

translate_labels	<i>Take a vector of labels and translate</i>
------------------	----------------------------------------------

Description

Take a vector of labels and translate

Usage

```
translate_labels(labels, lang = traduire::translator()$language())
```

Arguments

labels	Vector of labels, the translation keys
lang	Language to return output in

Value

Translated labels for current language

Index

`get_calibration_option_labels`, [2](#)

`get_controls_json`, [2](#)

`translate_labels`, [3](#)