Package 'rpcdsearch'

August 29, 2016

Title Tools for the Construction of Clinical Code Lists for Primary
Care Database Studies

Version 1.0

Author David Springate [aut, cre],
Evangelos Kontopantelis [ctb],
Ivan Olier [ctb]

Maintainer David Springate <daspringate@gmail.com>

Description Allows users to identify relevant clinical codes and
automate the construction of clinical code lists for primary care database
studies. This package is analogous to the Stata command pcdsearch.

License GPL (>= 2)

Depends R (>= 3.1.0)

Imports assertthat, stringr, dplyr, combinat, xlsx, methods, utils

Suggests testthat, roxygen2, knitr

VignetteBuilder knitr

LazyData true

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-01-18 10:05:38

R topics documented:

---

definition_search   *This function is used to build new definition lists based on medical definitions*

Description

This function is used to build new definition lists based on medical definitions

Usage

```
definition_search(def, medical_table = NULL, test_table = NULL,
  drug_table = NULL, lookup = NULL)
```

Arguments

def	an object of class MedicalDefinition
medical_table	Dataframe lookup table of clinical codes
test_table	dataframe lookup table of test codes
drug_table	dataframe lookup table of medication product codes
lookup	list containing elements: "codes", "terms", "tests", "drugs", "drugcodes" (see details)

Details

You may get an invalid multibyte string error, in which case, set fileEncoding="latin1" on read.delim when reading in the lookup tables Lookup tables are

Examples

```
## Not run:
medical_table <- read.delim("medical.txt", fileEncoding="latin1", stringsAsFactors = FALSE)
drug_table <- read.delim("product.txt", fileEncoding="latin1", stringsAsFactors = FALSE)
def2 <- import_definition_lists(system.file("extdata", "example_search.csv",
                                            package = "rpcdsearch"))
draft_lists <- definition_search(def2, medical_table, drug_table = drug_table)

## End(Not run)
```

---

export_definition_search

Exports definition searches to an excel file

Description

Exports definition searches to an excel file

Usage

```
export_definition_search(definition_search, out_file)
```

Arguments

definition_search

a list of dataframes as produced by build_definition_lists

out_file   file path to the excel file to be exported

Examples

```
## Not run:
medical_table <- read.delim("medical.txt", fileEncoding="latin1", stringsAsFactors = FALSE)
drug_table <- read.delim("product.txt", fileEncoding="latin1", stringsAsFactors = FALSE)
def2 <- import_definition_lists(system.file("extdata", "example_search.csv",
                                            package = "rpcdsearch"))
draft_lists <- definition_search(def2, medical_table, drug_table = drug_table)
out_file <- "def_searches.xlsx"
export_definition_search(draft_lists, out_file)

## End(Not run)
```

---

get_EHR_attribute   *Return the value of an attribute in the .ehr environment*

Description

Return the value of an attribute in the .ehr environment

Usage

```
get_EHR_attribute(x = NULL)
```

Arguments

x   an attribute name

Examples

```
{
set_CPRD()
get_EHR_attribute()
get_EHR_attribute(patient_id)
}
```

---

import_definitions	*Imports definitions to be searched from a csv file into a MedicalDefinition object*

Description

Imports definitions to be searched from a csv file into a MedicalDefinition object

Usage

```
import_definitions(input_file)
```

Arguments

input_file   character path to the input file

Examples

```
def2 <- import_definitions(system.file("extdata", "example_search.csv",
                                            package = "rpcdsearch"))
```

---

list_EHR_attributes	*Lists all of the EHR attribute names in .ehr*

Description

Lists all of the EHR attribute names in .ehr

Usage

```
list_EHR_attributes()
```

MedicalDefinition   *Constructor function for MedicalDefinition class*

Description

Constructor function for MedicalDefinition class

Usage

```
MedicalDefinition(terms = NULL, codes = NULL, tests = NULL,
  drugs = NULL, drugcodes = NULL)
```

Arguments

terms	list of character vectors or NULL
codes	list of character vectors or NULL
tests	list of character vectors or NULL
drugs	list of character vectors or NULL
drugcodes	list of character vectors or NULL

Details

Elements marked with a "-" are excluded. Elements marked with a "r

Examples

```
def <- MedicalDefinition(terms = list(c("angina", "unstable"), c("angina", "Crescendo "),
                                      c("angina", "Refractory")),
                        codes = list("G33..00", "G330.00", "%r212H", "-G617"))
class(def)
```

---

print.MedicalDefinition
                Basic print method for medical definition classes

---

Description

Basic print method for medical definition classes

Usage

```
## S3 method for class 'MedicalDefinition'
print(x, ...)
```

Arguments

x	an object of class "medical_definition"
...	Potential further arguments (required for method/generic reasons)

---

rpcdsearch	*The rpcdsearch package.*

Description

The rpcdsearch package.

---

set_CPRD	*Sets EHR metadata to CPRD format When this is run, most functions in rEHR act as though the EHR database is CPRD*

Description

Sets EHR metadata to CPRD format When this is run, most functions in rEHR act as though the EHR database is CPRD

Usage

```
set_CPRD()
```

Details

Metadata on EHR type is stored in the .ehr environment. This allows the same functions to work across different data sources. The .ehr environment is not desgined to be accessible to the user, but accessor functions are provided. CPRD is the default EHR setting.

See Also

get_EHR_value set_EHR_value

set_EHR_attribute   *Sets the value of an attribute in the .ehr environment*

Description

Sets the value of an attribute in the .ehr environment

Usage

```
set_EHR_attribute(x, value)
```

Arguments

x	an ehr attribute name
value	the value to set to the attribute

Examples

```
 {
set_CPRD()
set_EHR_attribute(practice_id, "pracid")
}
```

Index