

Package ‘pmeasyr’

November 19, 2017

Type Package

Title Donnees PMSI avec R

Version 0.2.0

Date 2017-09-15

Imports readr,
tidyr,
stringr,
dplyr,
lubridate,
magrittr,
forcats,
purrr,
sjlabelled,
knitr,
DBI,
jsonlite,
sqldf

Author Guillaume Pressiat

Maintainer Guillaume Pressiat <guillaume.pressiat@aphp.fr>

Description Import de donnees PMSI. Gestion des archives. Formats depuis 2011. Connexion et interface avec une db. requetr.

License GPL-2 | file LICENSE

LazyData TRUE

RoxygenNote 6.0.1

VignetteBuilder knitr

Encoding UTF-8

R topics documented:

adelete	3
adezip	4
adezip2	5
adezip3	6
astat	7
collect_rsa_from_db	7
creer_json	8

db_generique	9
db_had_out	9
db_liste_tables	10
db_mco_in	11
db_mco_out	11
db_psy_out	12
db_rsf_out	13
db_ssr_out	14
dico	14
enrobeur	15
formats	16
iano_had	16
iano_mco	17
iano_psy	18
iano_rafael	19
iano_ssr	20
idiap	21
idmi_mco	22
iium	23
iium_ssr	24
ileg_had	25
ileg_mco	26
ileg_ssr	27
imed_had	28
imed_mco	29
imed_ssr	30
inner_tra	31
ipo	32
ir3a	33
irafael	34
irapss	35
irha	36
irpsa	37
irsa	38
irum	39
issrha	40
itra	41
labeleasier	43
lancer_requete	43
lancer_requete_db	44
noyau_pmeasyr	45
noyau_skeleton	46
prepare_rsa	47
requete	47
requete_db	48
tbl_had	49
tbl_mco	49
tbl_psy	50
tbl_rsf	50
tbl_ssr	51
tdiag	51
%+%	52

adelete ~ *.zip - *Suppression des fichiers en fin de traitement*

Description

Supprime les fichiers de l'archive PMSI dezippés en début de traitement

Usage

```
adelete(finess, annee, mois, path, liste, type)
```

Arguments

finess	Finess du fichier à supprimer
annee	Annee du fichier
mois	Mois du fichier
path	Chemin d'accès aux fichiers
liste	Liste des fichiers à effacer : par défaut a "", efface tous les fichiers finess.annee.mois.
type	Type de fichier In / Out : par défaut a "", efface tous les fichiers finess.annee.mois.

Author(s)

G. Pressiat

See Also

[adezip](#), [adezip2](#), [astat](#), utiliser un noyau de paramètres avec [noyau_pmeasyr](#)

Examples

```
## Not run:  
adelete('750712184',2016,2, path = '~/Exemple', liste = c("rss","ano"), type = "in")  
  
adelete('750712184',2016,2, path = '~/Exemple')  
  
## End(Not run)
```

adezip ~ *.zip - Dezippe des fichiers de l'archive PMSI

Description

Dezipper une archive PMSI au besoin

Usage

```
adezip(finess, annee, mois, path, liste, pathto = "", type, recent = T)
```

Arguments

finess	Finess du fichier a dezipper
annee	Annee du fichier
mois	Mois du fichier
path	Chemin d'accès au fichier
liste	des fichiers à dezipper ex: ano, rss, rsa, dmi, ... ; si liste = "", dezippe la totalite de l'archive
type	Type de l'archive : in / out
recent	par default a T, l'archive la plus recente sera utilisee, sinon propose a l'utilisateur de choisir quelle archive dezipper
pathto	par default a "", dezipper la ou est l'archive, sinon preciser le chemin ou dezipper les fichiers (ailleurs)

Author(s)

G. Pressiat

See Also

[adezip2](#), [astat](#), [adelete](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
adezip('750712184',2016,2, path = '~/Documents/R/sources/2016',
      liste = 'med',
      pathto = "~/Exemple",
      type = "out")

adezip('750712184',2016,2, path = '~/Documents/R/sources/2016',
      liste = c('med','rapss', 'ano'),
      pathto = "~/Exemple",
      type = "in")

adezip('750712184',2016,2, path = '~/Documents/R/sources/2016',
      liste = c('rss', 'ano'),
      pathto = "~/Exemple",
      type = "in",
      recent = F)
```

```
## End(Not run)
```

```
adezip2 ~ *.zip - Dezippe des fichiers de l'archive PMSI, avec en parametre le  
nom de l'archive
```

Description

Alternative à la fonction [adezip](#), si on connaît précisément l'archive que l'on veut utiliser.

Usage

```
adezip2(path, file, liste = "", pathto = "")
```

Arguments

path	Chemin d'accès à l'archive
file	Nom de l'archive zip (ex: 750712184.2016.2.05042016093044.in.zip)
liste	Liste des fichiers à dézipper parmi l'archive ; si liste = "", dézippe la totalité de l'archive
pathto	Chemin où déposer les fichiers dézippés, par défaut à "", les fichiers sont mis là où se trouve l'archive

Author(s)

G. Pressiat

See Also

[adezip](#), [astat](#), [adelete](#)

Examples

```
## Not run:  
# Fichier ano  
adezip2(path = '~/Documents/R/sources/2011/',  
        file = '750712184.2011.12.27012012141857.in.zip',  
        liste = 'ano')  
  
# Totalité de l'archive  
adezip2(path = '~/Documents/R/sources/2011/',  
        file = '750712184.2011.12.27012012141857.in.zip',  
        liste = '')  
  
## End(Not run)
```

adezip3 ~ *.zip - Dezippe des fichiers de l'archive PMSI en provenance de l'Intranet AP-HP, avec en parametre le nom de l'archive

Description

Version de la fonction [adezip2](#) pour des archives au format Intranet du DIM Siège de l'AP-HP, <http://dime.aphp.fr/>.

Usage

```
adezip3(finess, path, file, liste = "", pathto = "")
```

Arguments

finess	Finess du fichier a dezipper
path	Chemin d'accès au fichier
file	Nom de l'archive zip (ex: 'MCO_IN_00000_201603.zip')
liste	des fichiers a dezipper ex: ano, rss, rsa, dmi, ... ; si liste = "", dezippe la totalite de l'archive
pathto	Chemin ou déposer les fichiers dezippes, par défaut à "", les fichiers sont mis la ou se trouve l'archive

Author(s)

G. Pressiat

See Also

[adezip2](#), [adezip](#), [astat](#), [adelete](#)

Examples

```
## Not run:
# Fichier ano
adezip3(path = '~/Downloads',
        file = 'MCO_IN_00000_201603.zip',
        liste = 'ano')

# Totalité de l'archive
adezip3(path = '~/Downloads',
        file = 'MCO_IN_00000_201603.zip',
        liste = '')

## End(Not run)
```

astat ~ *.zip - Liste et volume des fichiers d'une archive PMSI

Description

Pour lister sans dezipper les fichiers d'une archive

Usage

```
astat(path, file, view = T)
```

Arguments

path	Chemin d'accès à l'archive
file	Nom du fichier archive
view	par défaut à T : affiche la liste avec View(), à F retourne la table affichée à T

Author(s)

G. Pressiat

See Also

[adezip](#), [adezip2](#)

Examples

```
## Not run:
liste <- astat(path = '~/Documents/R/sources/2016/',
               file = "750712184.2016.2.05042016093044.in.zip",
               view = F)

## End(Not run)
```

collect_rsa_from_db ~ req : collecter les rsa présents dans une db

Description

On sélectionne certaines variables et on en crée d'autres utiles lors de l'exécution de la requête

Usage

```
collect_rsa_from_db(con, an, n = Inf)
```

Value

un liste rsa similaire à un import de pmeasyr

Author(s)

G. Pressiat

Examples

```
## Not run:  
rsa <- collect_rsa_from_db(con, 16, n = 1e5)  
  
## End(Not run)
```

creer_json

~ req : creer un json pour partager une liste

Description

On selectionne certaines variables et on en cree d'autres utiles lors de l'execution de la requete

Usage

```
creer_json(requete, chemin)
```

Value

un objet de classe rsa de pmeasyr

Author(s)

G. Pressiat

Examples

```
## Not run:  
liste = list(nom_abrege = "pac",  
            thematique = "test",  
            actes = c('EBLA003', 'EQLF002'),  
            dureemax = 0,  
            auteur = "John Doe",  
            date_saisie = "1960-01-01")  
creer_json(liste, paste0('~/Documents/listes/', liste$nom_abrege, '.json'))  
  
## End(Not run)
```

db_generique ~ db - Copier un tibble dans une db

Description

Copier une table R dans une db

Usage

```
db_generique(con, an, table, prefix, suffix, indexes = list(), remove = T)
```

Arguments

con	la connexion a la base de donnees (src_..)
an	l'annee pmsi
table	La table R (tibble) a copier dans la db
prefix	prefixe de la table dans la db (ex : mco, rsf, ssr, ...)
suffix	suffixe de la table dans la db (ex : rum_rum, rha_actes, rapss_rapss, ...)
indexes	index a ajouter a la table dans la base (voir copy_to)

Details

La tables déjà importée dans R est copiée dans la db

Value

nothing

Examples

```
## Not run:  
purrr::quietly(db_generique)(con, 16, ma_table, 'had', 'rapss_ano') -> statuts ; gc(); #  
# Result in db : had_16_rapss_ano  
  
## End(Not run)
```

db_had_out ~ db - Copier les rapss dans une db

Description

Copier les rapss, les actes, les diagnostics et la table ano des rapss dans une db

Usage

```
db_had_out(con, p, remove = T, zip = T, indexes = list(), ...)
```

Arguments

con	la connexion a la base de donnees (src_..)
p	le noyau pmeasyr
remove	a TRUE, les tables precedentes rapss de l'annee sont effacees avant
zip	a TRUE les fichiers des archives sont dezippes et effaces apres integration dans la db
indexes	index a ajouter a la table dans la base (voir copy_to)

Details

Les tables sont importées dans R puis copiées dans la db. Le tra est ajouté aux tables.

Value

nothing

Examples

```
## Not run:
purrr::quietly(db_had_out)(con, p) -> statuts ; gc(); #ok
purrr::quietly(db_had_out)(con, p, annee = 2017, mois = 7) -> statuts ; gc(); #..
## End(Not run)
```

db_liste_tables ~ db - Lister les tables d'une db en tableau

Description

~ db - Lister les tables d'une db en tableau

Usage

```
db_liste_table(con, nb = 15)
```

Arguments

con	la connexion a la base de donnees (src_..)
nb	le nombre de lignes du tableau

Value

nothing

Examples

```
## Not run:
db_liste_tables(con)
## End(Not run)
```

db_mco_in	~ db - Copier les rum dans une db
-----------	-----------------------------------

Description

Copier les rum, les actes et les diagnostics des rums dans une db

Usage

```
db_mco_in(con, p, remove = T, zip = T, indexes = list(), ...)
```

Arguments

con	la connexion a la base de donnees (src_..)
p	le noyau pmeasyr
remove	a TRUE, les tables precedentes rum de l'annee sont effacees avant
zip	a TRUE les fichiers des archives sont dezippees et effaces apres integration dans la db
indexes	index a ajouter a la table dans la base (voir copy_to)

Details

Les tables sont importées dans R puis copiées dans la db La table diag est créée et la durée des rum est calculée (DUREESEJPART)

Value

nothing

Examples

```
## Not run:  
purrr::quietly(db_mco_in)(con, p) -> statuts ; gc(); #ok  
purrr::quietly(db_mco_in)(con, p, annee = 2015) -> statuts ; gc(); #..  
  
## End(Not run)
```

db_mco_out	~ db - Copier les rsa dans une db
------------	-----------------------------------

Description

Copier les rsa, les passages um, les actes et les diagnostics des rsa, et ano dans une db

Usage

```
db_mco_out(con, p, remove = T, zip = T, indexes = list(), ...)
```

Arguments

con	la connexion a la base de donnees (src_..)
p	le noyau pmeasyr
remove	a TRUE, les tables precedentes rsa de l'annee sont effacees avant
zip	a TRUE les fichiers des archives sont dezippes et effaces apres integration dans la db
indexes	index a ajouter a la table dans la base (voir copy_to)

Details

Les tables sont importées dans R puis copiées dans la db. La table diag est créée, les variables ghm, année séquentielle des tarifs et un champ caractère diagnostics sont ajoutés à la table rsa. Le tra est ajouté aux tables.

Value

nothing

Examples

```
## Not run:
purrr::quietly(db_mco_out)(con, p) -> statuts ; gc(); #ok
purrr::quietly(db_mco_out)(con, p, annee = 2017, mois = 7) -> statuts ; gc(); #..

## End(Not run)
```

db_psy_out ~ db - Copier les rpsa dans une db

Description

Copier les rpsa, les actes, les diagnostics et la table ano des rpsa dans une db

Usage

```
db_psy_out(con, p, remove = T, zip = T, indexes = list(), ...)
```

Arguments

con	la connexion a la base de donnees (src_..)
p	le noyau pmeasyr
remove	a TRUE, les tables precedentes rpsa de l'annee sont effacees avant
zip	a TRUE les fichiers des archives sont dezippes et effaces apres integration dans la db
indexes	index a ajouter a la table dans la base (voir copy_to)

Details

Les tables sont importées dans R puis copiées dans la db. Le tra est ajouté aux tables.

Value

nothing

Examples

```
## Not run:
purrr::quietly(db_psy_out)(con, p) -> statuts ; gc(); #ok
purrr::quietly(db_psy_out)(con, p, annee = 2017, mois = 6) -> statuts ; gc(); #..

## End(Not run)
```

db_rsf_out ~ db - Copier les rsf dans une db

Description

Copier tous les rsf (lettre par lettre, A, B, C, ...), les ano-ace dans une db

Usage

```
db_rsf_out(con, p, remove = T, zip = T, indexes = list(), ...)
```

Arguments

con	la connexion a la base de donnees (src_..)
p	le noyau pmeasyr
remove	a TRUE, les tables precedentes rafael de l'annee sont effacees avant
zip	a TRUE les fichiers des archives sont dezippes et effaces apres integration dans la db
indexes	index a ajouter a la table dans la base (voir copy_to)

Details

Les tables sont importées dans R puis copiées dans la db

Value

nothing

Examples

```
## Not run:
purrr::quietly(db_rsf_out)(con, p) -> statuts ; gc(); #ok
purrr::quietly(db_rsf_out)(con, p, annee = 2014) -> statuts ; gc(); #ok

## End(Not run)
```

db_ssr_out ~ db - Copier les rha dans une db

Description

Copier les rha, les actes, les diagnostics des rha, ssrha et ano dans une db

Usage

```
db_ssr_out(con, p, remove = T, zip = T, indexes = list(), ...)
```

Arguments

con	la connexion a la base de donnees (src_..)
p	le noyau pmeasyr
remove	a TRUE, les tables precedentes rha de l'annee sont effacees avant
zip	a TRUE les fichiers des archives sont dezippes et effaces apres integration dans la db
indexes	index a ajouter a la table dans la base (voir copy_to)

Details

Les tables sont importées dans R puis copiées dans la db. Le tra est ajouté aux tables.

Value

nothing

Examples

```
## Not run:
purrr::quietly(db_ssr_out)(con, p) -> statuts ; gc(); #ok
purrr::quietly(db_ssr_out)(con, p, annee = 2017, mois = 7) -> statuts ; gc(); #..
## End(Not run)
```

dico ~ Dico - Dictionnaire des tables

Description

Obtenir le dictionnaire d'une table

Usage

```
dico(table)
```

Arguments

table	Table dont on veut le dictionnaire de variables
-------	---

Author(s)

G. Pressiat

See Also[irsa](#), [irum](#)**Examples**

```
## Not run:
# N'importer qu'une ligne du fichier :
  irsa('750712184', 2016, 8, '~/path/path', typi= 1, n_max = 1) -> import
  dico(import$rsa)

## End(Not run)
```

 enrobeur

 ~ req : mise en forme d'une liste de codes

Description

~ req : mise en forme d'une liste de codes

Usage

```
enrobeur(a, robe = "", colonne = F, interstice = ", ", symetrique = F)
```

Author(s)

G. Pressiat voir <https://guillaumepressiat.shinyapps.io/transcodeur/> pour son utilisation interactive hors AP-HP

Examples

```
## Not run:
li <- c('QEFA003', 'QEFA005', 'QEFA010', 'QEFA013', 'QEFA015', 'QEFA019', 'QEFA020')

enrobeur(li, robe="", interstice="|") %>% cat()

enrobeur(li, robe="\'", interstice=",") %>% cat()

enrobeur(li, robe="\'%", interstice="\n", symetrique = T) %>% cat()

## End(Not run)
```

formats	<i>Table des formats</i>
---------	--------------------------

Description

Table des formats

Author(s)

G. Pressiat

iano_had	<i>~ HAD - Import des Anohosp</i>
----------	-----------------------------------

Description

Imports du fichier Ano Out

Usage

```
iano_had(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~ ...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu (sortie de Paprica) : *finess.annee.moisc.ano*

750712184.2016.2.ano

Value

Une table (data.frame, tbl_df) contenant les données Anohosp HAD du Out.

Author(s)

G. Pressiat

See Also

[irapss](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  anoh <- iano_had('750712184',2015,12,"~/Documents/data/had")

## End(Not run)
```

 iano_mco

 ~ MCO - Import des Anohosp

Description

Import du fichier ANO In ou Out.

Usage

```
iano_mco(finess, annee, mois, path, lib = T, tolower_names = F, typano = "out")
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des données sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
typano	Type de donnees In / Out
lib	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~ . . .	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu : *finess.annee.moisc.ano*
finess.annee.moisc.ano.txt

750712184.2016.2.ano 750712184.2016.2.ano.txt

Value

Une table (data.frame ou tbl_df) qui contient les données Anohosp in / out

Author(s)

G. Pressiat

See Also

[irum](#), [irsa](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
iano_mco('750712184',2015,12,'~/Documents/data/mco') -> ano_out15
iano_mco('750712184',2015,12,'~/Documents/data/mco', typano = "in") -> ano_in15

## End(Not run)
```

iano_psy

~ PSY - Import des Anohosp

Description

Import du fichier Ano Out

Usage

```
iano_psy(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2012 pris en charge Structure du nom du fichier attendu (sortie de Genrha) : *finess.annee.moisc.ano*

750712184.2016.2.ano

Value

Une table (data.frame, tbl_df) contenant les données Anohosp SSR du Out.

Author(s)

G. Pressiat

See Also

[irpsa](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  anoh <- iano_psy('750712184',2015,12,"~/Documents/data/psy")

## End(Not run)
```

iano_rafael ~ RSF - Import des Anohosp RSFA

Description

Import du fichier ANO-ACE RSF Out ou le ano-ace-maj (reprise)

Usage

```
iano_rafael(finess, annee, mois, path, lib = T, lamda = F, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lamda	a TRUE, importe le fichier ano-ace-maj
lib	Ajout des libelles de colonnes aux tables, par defaut a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2012 pris en charge pour les ano-ace Formats depuis 2014 pris en charge pour les ano-ace-maj (reprise 2013)

Structure du nom du fichier attendu : *finess.annee.moisc.ano*

750712184.2016.2.ano

Value

Une table (data.frame ou tbl_df) qui contient les données Anohosp in / out

Author(s)

G. Pressiat

See Also

[irafael](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
iano_rafael('750712184', 2015, 12, '~/Documents/data/rsf') -> ano_out15
iano_rafael('750712184', 2015, 12, '~/Documents/data/rsf', lamda = T) -> lamda_maj_ano_out14

## End(Not run)
```

iano_ssr

~ SSR - Import des Anohosp

Description

Import du fichier Ano Out

Usage

```
iano_ssr(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	paramètres supplémentaires à passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premières lignes, progress = F, skip =...

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu (sortie de Genrha) : *finess.annee.moisc.ano*

750712184.2016.2.ano

Value

Une table (data.frame, tbl_df) contenant les données Anohosp SSR du Out.

Author(s)

G. Pressiat

See Also

[irha](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  idiap <- idiap('750712184',2015,12,"~/Documents/data/mco")

## End(Not run)
```

idmi_mco

~ MCO - Import des DMI

Description

Import des fichiers DMI In ou Out.

Usage

```
idmi_mco(...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
typdmi	Type de donnees In / Out
lib	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge

Value

Une table (data.frame, tbl_df) contenant les dispositifs médicaux implantables In ou Out (T2A, ATU et thrombo selon l'existence des fichiers : si le fichier n'existe pas, pas de donnée importée). Pour discriminer le type de prestation, la colonne TYPEPREST donne l'information : T2A 06 - ATU 09 - THROMBO 10

Author(s)

G. Pressiat

See Also

[irum](#), [irsa](#)

Examples

```
## Not run:
idmi_mco('750712184',2015,12,'~/Documents/data/mco') -> dmi_out15
idmi_mco('750712184',2015,12,'~/Documents/data/mco', typdmi = "in") -> dmi_in15

## End(Not run)
```

iium

~ MCO - Import des donnees UM du Out

Description

Imports du fichier IUM MCO

Usage

```
iium(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par defaut a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires à passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge

Value

Une table (data.frame, tbl_df) contenant les informations structures du Out.

Author(s)

G. Pressiat

See Also
[irsa](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  um <- iium('750712184',2015,12,"~/Documents/data/mco")

## End(Not run)
```

```
iium_ssr          ~ SSR - Import des donnees UM du Out
```

Description

Imports du fichier IUM SSR

Usage

```
iium_ssr(finess, annee, mois, path, lib = T, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires à passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2013 pris en charge

Value

Une table (data.frame, tbl_df) contenant les informations structures du Out.

Author(s)

G. Pressiat

See Also

[irsa](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  um <- iium_ssr('750712184',2015,12,"~/Documents/data/ssr")

## End(Not run)
```

ileg_had ~ HAD - Import des erreurs Leg

Description

Import de la liste d'erreurs de génération Paprica

Usage

```
ileg_had(finess, annee, mois, path, reshape = F, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
reshape	booleen TRUE/FALSE : la donnee doit-elle etre restructuree ? une ligne = une erreur, sinon, une ligne = un sejour. par default a F
tolower_names	a TRUE les noms de colonnes sont tous en minuscules

Value

Une table (data.frame, tbl_df) contenant les erreurs Out.

Author(s)

G. Pressiat

See Also

[irapss](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:  
  ileg_had('750712184', 2015, 12, '~/Documents/data/had') -> leg15  
  
## End(Not run)
```

ileg_mco ~ MCO - Import des erreurs Leg

Description

Import de la liste d'erreurs de génération Gensra

Usage

```
ileg_mco(finess, annee, mois, path, reshape = F, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
reshape	booleen TRUE/FALSE : la donnee doit-elle etre restructuree ? une ligne = une erreur, sinon, une ligne = un sejour. par defaut a F
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge

Value

Une table (data.frame, tbl_df) contenant les erreurs Out.

Author(s)

G. Pressiat

See Also

[irum](#), [irsa](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  ileg_mco('750712184', 2015, 12, '~/Documents/data/mco') -> leg15

## End(Not run)
```

ileg_ssr ~ SSR - Import des erreurs Leg

Description

Import de la liste d'erreurs de génération Genrha

Usage

```
ileg_ssr(finess, annee, mois, path, reshape = F, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
reshape	booleen TRUE/FALSE : la donnee doit-elle etre restructuree ? une ligne = une erreur, sinon, une ligne = un sejour. par default a F
tolower_names	a TRUE les noms de colonnes sont tous en minuscules

Value

Une table (data.frame, tbl_df) contenant les erreurs Out.

Author(s)

G. Pressiat

See Also

[irha](#), [issrha](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:  
ileg_had('750712184', 2015, 12, '~/Documents/data/ssr') -> leg15  
  
## End(Not run)
```

imed_had ~ HAD - Import des Med

Description

Imports du fichier Med Out

Usage

```
imed_had(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par defaut a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu (sortie de Paprica) : *finess.annee.moisc.med*

750712184.2016.2.med

Value

Une table (data.frame, tbl_df) contenant les données médicaments HAD du Out.

Author(s)

G. Pressiat

See Also

[irapss](#) utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:  
medh <- imed_had('750712184', 2015, 12, "~/Documents/data/had")  
  
## End(Not run)
```

imed_mco ~ MCO - Import des Med

Description

Import des fichiers MED In ou Out.

Usage

```
imed_mco(finess, annee, mois, path, lib = T, tolower_names = F, typmed = c('out', 'in'))
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
typmed	Type de donnees In / Out
lib	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge

Value

Une table (data.frame, tibble) contenant les médicaments In ou Out (T2A, ATU et thrombo selon l'existence des fichiers : si le fichier n'existe pas, pas de donnée importée). Pour discriminer le type de prestation, la colonne TYPEPREST donne l'information : T2A 06 - ATU 09 - THROMBO 10

Author(s)

G. Pressiat

See Also

[irum](#), [irsa](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
imed_mco('750712184',2015,12,'~/Documents/data/mco') -> med_out15
imed_mco('750712184',2015,12,'~/Documents/data/mco', typmed = "in") -> med_in15

## End(Not run)
```

imed_ssr ~ SSR - Import des Med

Description

Imports du fichier Med Out

Usage

```
imed_ssr(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par defaut a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu (sortie de Genrha) : *finess.annee.moisc.med*

750712184.2017.2.med

Value

Une table (data.frame, tbl_df) contenant les données médicaments SSR du Out.

Author(s)

G. Pressiat

See Also

[irapss](#) utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
meds <- imed_ssr('750712184', 2015, 12, "~/Documents/data/ssr")

## End(Not run)
```

inner_tra ~ TRA - Ajout du TRA aux donnees Out

Description

Ajout du TRA par dplyr::inner_join

Usage

```
inner_tra(table, tra, sel = 1, champ = "mco")
```

Arguments

table	Table a laquelle rajouter le tra
tra	tra a rajouter
sel	Variable a garder du tra ; sel = 1 : numero de sejour, sel = 2 : toutes les variables
champ	Champ PMSI : mco, had, sss, psy : deux tra en psy : psy_rpsa, psy_r3a

Value

Une table contenant le inner_join entre table et tra

Author(s)

G. Pressiat

See Also

[irum](#), [irsa](#), [imed_mco](#), [irpsa](#), [irha](#), [irapss](#)

Examples

```
## Not run:
med <- imed_mco('750712184',2015,12,"~/Documents/data/mco","out")
tra <- itra('750712184',2015,12,"~/Documents/data/mco")
med <- inner_tra(med,tra)

## End(Not run)
```

ipo ~ MCO - Import des PO

Description

Imports des fichiers PO In / Out

Usage

```
ipo(finess, annee, mois, path, typpo = c("out", "in"), lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
typpo	Type de donnees In / Out
lib	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge

Value

Une table (data.frame, tibble) contenant les prélèvements d'organes In ou Out.

Author(s)

G. Pressiat

See Also

[irum](#), [irsa](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  po <- ipo('750712184', 2015, 12, "~/Documents/data/mco")

## End(Not run)
```

ir3a ~ PSY - Import des R3A

Description

Import du fichier R3A

Usage

```
ir3a(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des données (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par defaut a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2012 pris en charge Structure du nom du fichier attendu (sortie de Pivoine) : *finess.annee.moisc.r3a*

750712184.2016.3.r3a

Value

Une table (data.frame, tbl_df) contenant les données R3A.

Author(s)

G. Pressiat

See Also

[irpsa](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:  
r3a <- ir3a('750712184', 2015, 12, "~/Documents/data/psy")  
  
## End(Not run)
```

irafael ~ RSF - Import des Rafael

Description

Import des Rafael et des Rafael reprises

Usage

```
irafael(finess, annee, mois, path, lib = T, tolower_names = F, stat = T, lister = c("A", "B",
"C", "H", "L", "M", "P"), lamda = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par défaut a TRUE ; necessite le package sjlabelled
stat	avec stat = T, un tableau synthetise le nombre de lignes par type de rafael
lister	Liste des types d'enregistrements a importer
lamda	a TRUE, importe les fichiers rsfa-maj de reprise de l'annee passee
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	Autres parametres a specifier n_max = 1e3, ...

Details

Formats depuis 2012 pour les rsfa Formats depuis 2014 pour les rsfa-maj (reprise 2013)

Value

Une classe S3 contenant les tables (data.frame, tbl_df ou tbl) importées (rafaels)

Author(s)

G. Pressiat

See Also

[iano_rafael](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  irafael('750712184',2015,12,'~/Documents/data/rsf') -> rsfa15
  irafael('750712184',2015,12,'~/Documents/data/rsf', lister = 'C', lamda = T) -> rsfa14_lamda

## End(Not run)
```

irapss ~ HAD - Import des RAPSS

Description

Imports du fichier RAPSS

Usage

```
irapss(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des données sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu (sortie de Paprica) : *finess.annee.moisc.rapss*

750712184.2016.2.rapss

Value

Une classe S3 contenant les tables (data.frame, tbl_df ou tbl) importées (rapss, acdi, ght).

Author(s)

G. Pressiat

See Also

[iano_had](#), [ileg_had](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:  
um <- iium('750712184',2015,12,"~/Documents/data/had")  
  
## End(Not run)
```

irha ~ SSR - Import des RHA

Description

Import des RHA

Usage

```
irha(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du fichier Out de GENRHA a integrer
annee	Annee de la periode (du fichier Out)
mois	Mois de la periode (du fichier Out)
path	Chemin d'accès au fichier .rha
lib	Attribution de libelles aux colonnes
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple <code>n_max=10e3</code> pour lire les 1000 premieres lignes

Details

Formats depuis 2011 pris en charge

Author(s)

G. Pressiat

See Also

[iano_ssr](#), [ileg_ssr](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:  
irha('750712184',2015,12,'pathpath/') -> rha15  
  
## End(Not run)
```

irpsa ~ PSY - Import des RPSA

Description

Import du fichier RPSA

Usage

```
irpsa(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```

Arguments

finess	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par defaut a TRUE ; necessite le package sjlabelled
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction read_fwf , par exemple n_max = 1e3 pour lire les 1000 premieres lignes, progress = F, skip = 1e3

Details

Formats depuis 2012 pris en charge Structure du nom du fichier attendu (sortie de Pivoine) : *finess.annee.moisc.rpsa*

750712184.2016.2.rpsa

Value

Une table (data.frame, tbl_df) contenant les données RPSA.

Author(s)

G. Pressiat

See Also

[ir3a](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:  
rpsa <- irpsa('750712184',2015,12,"~/Documents/data/psy")  
  
## End(Not run)
```

 irsa

 ~ MCO - Import des RSA

Description

Import des RSA. 6 types d'imports possibles.

Usage

```
irsa(finness, annee, mois, path, lib = T, typi = 4, tolower_names = F, ...)
```

Arguments

finness	Finess du Out a importer : dans le nom du fichier
annee	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
mois	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
path	Localisation du fichier de donnees
lib	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package sjlabelled
typi	Type d'import, par default a 4, a 0 : propose a l'utilisateur de choisir au lancement
tolower_names	a TRUE les noms de colonnes sont tous en minuscules
~...	parametres supplementaires a passer dans la fonction <code>read_fwf</code> , par exemple <code>n_max = 1e3</code> pour lire les 1000 premieres lignes, <code>progress = F</code> , <code>skip = 1e3</code>

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu (sortie de Genrsa) : *finness.annee.moisc.rsa*

750712184.2016.2.rsa

Types d'imports :

1 Light :	partie fixe (très rapide)
2 Light+ :	Partie fixe + stream en ligne (+) actes et das
3 Light++ :	Partie fixe + stream en ligne (++) actes, das, typaut um et dpdr des um
4 Standard :	Partie fixe + création des tables acdi et rsa_um
5 Standard+ :	Partie fixe + création des tables acdi et rsa_um + stream (+)
6 Standard++ :	Partie fixe + création des tables acdi et rsa_um + stream (++)

Principe du streaming : Mise en chaîne de caractères de la succession d'actes CCAM au cours du RUM, par exemple, pour un RUM : "ACQK001, LFQK002, MCQK001, NAQK015, PAQK002, PAQK900, YYYY600, ZZQP004"

La recherche d'un (ou d'une liste d') acte(s) sur un RUM est largement accélérée, comparée à une requête sur la large table acdi par une requête du type :

```
grep1("ZZQP004", rsa$actes) # toutes les lignes de RSA avec au moins un ZZQP004
e66 <- grep1('E66', das) | grep1('E66', dpdrum) # toutes les lignes de RSA avec un diagnostic E66
```

Cela permet de n'utiliser que la seule table `rsa` avec `stream` et d'avoir les infos sur les séjours directement : nb séjours, journées, entrée / sortie (...) plutôt que d'avoir à utiliser et croiser les tables `acdi`, `rsa_um` avec `rsa`.

Value

Une classe S3 contenant les tables (`data.frame`, `tbl_df` ou `tbl`) importées (`rsa`, `rsa_um`, `actes` et `das` si `import > 3`)

Author(s)

G. Pressiat

See Also

[irum](#), [ileg_mco](#), [iano_mco](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  irsa('750712184',2015,12,'~/Documents/data/mco') -> rsa15

## End(Not run)
```

irum ~ MCO - Import des RUM

Description

Import des RUM. 4 types d'imports possibles.

Usage

```
irum(finess, annee, mois, path, lib = T, typi = 3, tolower_names = F, ...)
```

Arguments

- `finess` Finess du In a importer : dans le nom du fichier
- `annee` Annee PMSI (nb) des donnees sur 4 caracteres (2016)
- `mois` Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
- `path` Localisation du fichier de donnees
- `lib` Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package `sjlabelled`
- `typi` Type d'import, par default a 3, a 0 : propose a l'utilisateur de choisir au lancement
- `tolower_names` a TRUE les noms de colonnes sont tous en minuscules
- `~...` parametres supplementaires a passer dans la fonction [read_fwf](#), par exemple `n_max = 1e3` pour lire les 1000 premieres lignes, `progress = F`, `skip = 1e3`

Details

Formats depuis 2011 pris en charge

Structure du nom du fichier attendu (entrée pour Genrsa) : *finess.annee.moisc.rum*

750712184.2016.2.rum

Types d'imports :

- 1 XLight : partie fixe
- 2 Light : partie fixe + streaming des actes, dad et das
- 3 Standard : partie fixe + table acdi
- 4 Standard+ : Import standard (3) + stream

Principe du streaming : Mise en chaîne de caractères de la succession d'actes CCAM au cours du RUM, par exemple, pour un RUM : "ACQK001, LFQK002, MCQK001, NAQK015, PAQK002, PAQK900, YYYY600, ZZQP004"

La recherche d'un (ou d'une liste d') acte(s) sur un RUM est largement accélérée, comparée à une requête sur la large table acdi par une requête du type :

```
grep1("ZZQP004", rum$actes) # toutes les lignes de RUM avec au moins un ZZQP004
grep1("ZZQP004|EBLA003", rum$actes) # toutes les lignes de RUM avec au moins un ZZQP004 ou un EBLA003
```

Value

Une classe S3 contenant les tables (data.frame, tibble) importées (rum, actes, das et dad si import 3 et 4)

Author(s)

G. Pressiat

See Also

[irsa](#), [ileg_mco](#), [iano_mco](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
  irum('750712184', 2015, 12, '~/Documents/data/mco', typi = 1) -> rum15

## End(Not run)
```

issrha

~ SSR - Import des SSRHA

Description

Import du fichier SHA

Usage

```
issrha(finess, annee, mois, path, lib = T, tolower_names = F, ...)
```


Arguments

<code>finess</code>	Finess du Out a importer : dans le nom du fichier
<code>annee</code>	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
<code>mois</code>	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
<code>path</code>	Localisation du fichier de donnees
<code>lib</code>	Ajout des libelles a la table : T
<code>tolower_names</code>	a TRUE les noms de colonnes sont tous en minuscules
<code>~...</code>	parametres supplementaires a passer dans la fonction <code>read_fwf</code> , par exemple <code>n_max = 1e3</code> pour lire les 1000 premieres lignes, <code>progress = F</code> , <code>skip = 1e3</code>

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu (sortie de Genrha) : *finess.annee.moisc.sha*

750712184.2016.2.sha

Value

Une table (data.frame, tbl_df) contenant les données SHA, et a partir de 2017 une liste de deux tables (sha et gme)

Author(s)

G. Pressiat

See Also

[irha](#), [ileg_ssr](#), [iano_ssr](#), utiliser un noyau de parametres avec [noyau_pmeasyr](#)

Examples

```
## Not run:
sha <- issrha('750712184',2015,12,"~/Documents/data/ssr")

## End(Not run)
```

itra

~ TRA - Import du TRA

Description

Import du fichier TRA, 4 champs PMSI couverts.

Usage

```
itra(finness, annee, mois, path, lib = T, tolower_names = F, champ = "mco")
```

Arguments

<code>finess</code>	Finess du Out a importer : dans le nom du fichier
<code>annee</code>	Annee PMSI (nb) des donnees sur 4 caracteres (2016)
<code>mois</code>	Mois PMSI (nb) des donnees (janvier : 1, decembre : 12)
<code>path</code>	Localisation du fichier de donnees
<code>lib</code>	Ajout des libelles de colonnes aux tables, par default a TRUE ; necessite le package <code>sjlabelled</code>
<code>champ</code>	Champ PMSI du TRA a integrer ("mco", "ssr", "had", "psy_rpsa", ", ", "psy_r3a"), par default "mco"
<code>tolower_names</code>	a TRUE les noms de colonnes sont tous en minuscules
<code>~...</code>	parametres supplementaires a passer dans la fonction <code>read_fwf</code> , par exemple <code>n_max = 1e3</code> pour lire les 1000 premières lignes, <code>progress = F</code> , <code>skip = 1e3</code>

Details

Formats depuis 2011 pris en charge Structure du nom du fichier attendu (sortie de Genrsa) : *finess.annee.moisc.tra*

750712184.2016.2.tra

Value

Une table (`data.frame` ou `tbl_df`) qui contient : - Clé RSA - NORSS - Numéro de ligne du fichier RSS d'origine (`rss.ini`) - NAS - Date d'entrée du séjour - GHM groupage du RSS (origine) - Date de sortie du séjour

Author(s)

G. Pressiat

See Also

`irum`, `irsa`, `ileg_mco`, `iano_mco`, `irha`, `irapss`, `irpsa`, `ir3a`, utiliser un noyau de parametres avec `noyau_pmeasyr`

Examples

```
## Not run:
  itra('750712184', 2015, 12, '~/Documents/data/champ_pmsi') -> tra15

## End(Not run)
```

labelasier *~ Labels pour le PMSI*

Description

Attribuer des libelles aux colonnes PMSI

Usage

```
labelasier(col, Mode_entree = F, Mode_sortie = F, Provenance = F,
           Destination = F, Sexe = F, facteur = F)
```

Arguments

col	Colonne à laquelle attribuer le libellé
Mode_entree	'6' : 'Mutation'
Mode_sortie	'9' : 'Décès'
Provenance	'1' : 'MCO'
Destination	'6' : 'HAD'
Sexe	'2' : 'Femme'

Value

Un vecteur caractère ou facteur

Author(s)

G. Pressiat

Examples

```
## Not run:
  labelasier(rsa$rsa$SEXE, Sexe = T, F)
  labelasier(rsa$rsa$DEST, Destination = T, F)

## End(Not run)
```

lancer_requete *~ req : lancer une ou plusieurs requetes avec une ou des listes*

Description

~ req : lancer une ou plusieurs requetes avec une ou des listes

Usage

```
lancer_requete(tables, elements, vars = NULL)
```

Value

un tibble concatenant les resultats de toutes les requetes : les rsa qui ont un acte de la liste, un diag, une duree correspondante, etc.

Author(s)

G. Pressiat

Examples

```
## Not run:
prepare_rsa(rsa) -> rsa
get_all_listes('Recours Exceptionnel') -> listes_re
lancer_requete(rsa, listes_re)

get_liste('chir_bariatrique_total') -> liste_bari
lancer_requete(rsa, liste_bari)

## End(Not run)
```

`lancer_requete_db` *~ req : lancer une ou plusieurs requetes dans une db avec une ou des listes*

Description

~ req : lancer une ou plusieurs requetes dans une db avec une ou des listes

Usage

```
lancer_requete_db(con, an, elements, vars = NULL)
```

Value

un tibble concatenant les resultats de toutes les requetes : les rsa qui ont un acte de la liste, un diag, une duree correspondante, etc.

Author(s)

G. Pressiat

Examples

```
## Not run:
get_all_listes('Recours Exceptionnel') -> listes_re
lancer_requete_db(con, an, listes_re)

get_liste('chir_bariatrique_total') -> liste_bari
lancer_requete_db(con, an, liste_bari)

## End(Not run)
```

noyau_pmeasyr *~ par - Noyau de parametres*

Description

Définir un noyau de paramètres

Usage

```
noyau_pmeasyr(...)  
  
## S3 method for class 'pm_param'  
print(p)
```

Details

Voir exemple

Author(s)

G. Pressiat

See Also

[noyau_skeleton](#)

Examples

```
## Not run:  
library(magrittr)  
  
p <- noyau_pmeasyr(  
  fitness = '750712184',  
  annee = 2016,  
  mois = 12,  
  path = '~/Documents/data/mco',  
  progress = F  
)  
  
p %>% azip(type = "out", liste = "")  
  
p %>% irsa()        -> rsa  
p %>% iano_mco() -> ano  
p %>% ipo()        -> po  
  
azip(type = "in", liste = "")  
  
p %>% irum()       -> rum  
  
# Modifier le type d'import :  
irsa(p, typi = 6) -> rsa  
  
# Pour visualiser p :  
p
```

```
print(p)

## End(Not run)
```

```
noyau_skeleton      ~ par - Noyau de paramètres
```

Description

Générer un squelette de noyau de paramètres

Usage

```
noyau_skeleton(nom = "p", zip = F)
```

Details

Voir exemple

Author(s)

G. Pressiat

See Also

[noyau_pmeasyr](#)

Examples

```
## Not run:

noyau_skeleton()
## résultat :
## noyau_pmeasyr(
##   fitness = '.....',
##   annee = ....,
##   mois = ..,
##   path = ''
## ) -> p

noyau_skeleton("alpha_bravo", T)

## noyau_pmeasyr(
##   fitness = '.....',
##   annee = ....,
##   mois = ..,
##   path = ''
## ) -> alpha_bravo
##
## # adezip(alpha_bravo, type = 'out')
## # adezip(alpha_bravo, type = 'in')

## End(Not run)
```

prepare_rsa ~ req : *preparer les rsa pour la requete*

Description

On selectionne certaines variables et on en cree d'autres utiles lors de l'execution de la requete

Usage

```
prepare_rsa(rsa)
```

Value

un objet de classe rsa de pmeasyr

Author(s)

G. Pressiat

Examples

```
## Not run:  
tab <- irsa(typi = 6)  
prepare_rsa(tab)  
  
## End(Not run)
```

requete ~ req : *requeter les rsa avec une liste*

Description

~ req : requeter les rsa avec une liste

Usage

```
requete(tables, elements, vars = NULL)
```

Value

un tibble contenant les rsa respectant la requete : les rsa qui ont un acte de la liste, un diag, une duree correspondante, etc.

Author(s)

G. Pressiat

Examples

```
## Not run:
liste = list(actes = c('EBLA003', 'EQLF002'))
requete(rsa, liste)

liste = list(actes = c('EBLA003', 'EQLF002'), dureemax = 0)
requete(rsa, liste)

## End(Not run)
```

requete_db

~ req : requeter les rsa dans une db avec une liste

Description

~ req : requeter les rsa dans une db avec une liste

Usage

```
requete_db(con, an, elements, vars = NULL)
```

Value

un tibble contenant les rsa respectant la requete : les rsa qui ont un acte de la liste, un diag, une duree correspondante, etc.

Author(s)

G. Pressiat

Examples

```
## Not run:
liste = list(actes = c('EBLA003', 'EQLF002'))
requete_db(con, an, liste)

liste = list(actes = c('EBLA003', 'EQLF002'), dureemax = 0)
requete_db(con, 16, liste)

## End(Not run)
```

tbl_had	<i>~ db - remote access aux tables had</i>
---------	--

Description

~ db - remote access aux tables had

Usage

```
tbl_had(con, an, table)
```

Value

tibble

Examples

```
## Not run:  
tbl_had(con, 16, 'rapss_rapss')  
  
## End(Not run)
```

tbl_mco	<i>~ db - remote access aux tables mco</i>
---------	--

Description

~ db - remote access aux tables mco

Usage

```
tbl_mco(con, an, table)
```

Value

tibble

Examples

```
## Not run:  
tbl_mco(con, 16, 'rsa_rsa')  
  
## End(Not run)
```

tbl_psy ~ db - remote access aux tables psy

Description

~ db - remote access aux tables psy

Usage

tbl_psy(con, an, table)

Value

tibble

Examples

```
## Not run:  
tbl_psy(con, 16, 'rpsa_rpsa')  
  
## End(Not run)
```

tbl_rsf ~ db - remote access aux tables rsf

Description

~ db - remote access aux tables rsf

Usage

tbl_rsf(con, an, table)

Value

tibble

Examples

```
## Not run:  
tbl_rsf(con, 16, 'rsf_rafael_ano')  
  
## End(Not run)
```

tbl_ssr ~ db - remote access aux tables ssr

Description

~ db - remote access aux tables ssr

Usage

tbl_ssr(con, an, table)

Value

tibble

Examples

```
## Not run:
tbl_ssr(con, 16, 'rha_rha')

## End(Not run)
```

tdiag ~ Tidy - Tidy Diagnostics

Description

Restructurer les diagnostics

Usage

tdiag(d, include = T)

Arguments

d Objet S3 resultat de l'import pmeasyr (irsa, irum, irha)
include booleen : default a T; T : restructure l'objet S3 (agglomerer dp, dr, das et dad, par exemple)

Details

On obtient une table contenant tous les diagnostics par séjour, sur le principe suivant : Une variable numérique indique la position des diagnostics - pour les rsa : 1 : DP du rsa, 2 : DR du rsa, 3 : DPUM, 4 : DRUM, 5 : DAS - pour les rum : 1 : DP du rum, 2 : DR du rum, 3 : DAS, 4 : DAD - pour les rha : 1 : MMP du rha, 2 : FPPC du rha, 3 : AE, 4 : DA

Author(s)

G. Pressiat

See Also

[irsa](#), [irum](#), [irha](#)

Examples

```
## Not run:
# avec include = T
irum('750712184', 2016, 8, '~/path/path', typi = 3) -> d1
tdiag(d1) -> d1
d1$diags
d1$actes
d1$dad
irsa('750712184', 2016, 8, '~/path/path', typi = 4) -> d1
tdiag(d1, include = F) -> alldiag

## End(Not run)
```

%+%

~ ... paste0 comme un pipe

Description

~ ... paste0 comme un pipe

Usage

x

Examples

```
## Not run:
'Hého' %+% ' ' %+% 'world !'

## End(Not run)
```

Index

- *Topic **data**
 - formats, 16
- %+%, 52
- adelete, 3, 4–6
- adezip, 3, 4, 5–7
- adezip2, 3, 4, 5, 6, 7
- adezip3, 6
- astat, 3–6, 7

- collect_rsa_from_db, 7
- copy_to, 9–14
- creer_json, 8

- db_generique, 9
- db_had_out, 9
- db_liste_tables, 10
- db_mco_in, 11
- db_mco_out, 11
- db_psy_out, 12
- db_rsf_out, 13
- db_ssr_out, 14
- dico, 14

- enrobeur, 15

- formats, 16

- iano_had, 16, 35
- iano_mco, 17, 39, 40, 42
- iano_psy, 18
- iano_rafael, 19, 34
- iano_ssr, 20, 36, 41
- idiap, 21
- idmi_mco, 22
- iium, 23
- iium_ssr, 24
- ileg_had, 25, 35
- ileg_mco, 26, 39, 40, 42
- ileg_ssr, 27, 36, 41
- imed_had, 28
- imed_mco, 29, 31
- imed_ssr, 30
- inner_tra, 31
- ipo, 32

- ir3a, 33, 37, 42
- irafael, 19, 34
- irapss, 17, 25, 28, 30, 31, 35, 42
- irha, 20, 27, 31, 36, 41, 42, 52
- irpsa, 18, 31, 33, 37, 42
- irsa, 15, 18, 21–24, 26, 29, 31, 32, 38, 40, 42, 52
- irum, 15, 18, 21, 22, 26, 29, 31, 32, 39, 39, 42, 52
- issrha, 27, 40
- itra, 41

- labeleasier, 43
- lancer_requete, 43
- lancer_requete_db, 44

- noyau_pmeasyr, 3, 4, 17–20, 23–30, 32–37, 39–42, 45, 46
- noyau_skeleton, 45, 46

- prepare_rsa, 47
- print.pm_param (noyau_pmeasyr), 45

- read_fwf, 16–24, 26, 28–30, 32, 33, 35–39, 41, 42
- requete, 47
- requete_db, 48

- tbl_had, 49
- tbl_mco, 49
- tbl_psy, 50
- tbl_rsf, 50
- tbl_ssr, 51
- tdiag, 51