

Package ‘inesss’

January 19, 2021

Title Institut National Excellence Sante Services Sociaux

Version 0.4.0

Description Cette librairie fournit des fonctionnalités pour une variété de tâches propices au domaine de la santé et des outils pour visualiser les résultats.

License GPL-3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

Imports askpass (>= 1.1),
data.table (>= 1.13.4),
DBI (>= 1.1.0),
fs (>= 1.5.0),
lubridate (>= 1.7.9.2),
miniUI (>= 0.1.1.1),
odbc (>= 1.3.0),
readxl (>= 1.3.1),
rstudioapi (>= 0.13),
shiny (>= 1.5.0),
shinydashboard (>= 0.7.1),
shinyFiles (>= 0.9.0),
stringr (>= 1.4.0),
testthat (>= 3.0.1),
writexl (>= 1.3.1)

VignetteBuilder knitr

Depends R (>= 2.10)

Suggests knitr,
rmarkdown

R topics documented:

formulaire	2
Pop_QC	2
query_stat_gen1	3
RLS_convert	5
RLS_list	6

SQL_connexion	6
SQL_stat_gen1	7
Index	10

formulaire	<i>Formulaire Interactif</i>
------------	------------------------------

Description

Permet d'exécuter des requêtes Excel à partir d'un formulaire interactif. La documentation complète du formulaire, *AIDE_FORMULAIRE_DATE.pdf*, est disponible [ici](#).

Usage

```
formulaire()
```

Details

Requêtes via Excel :

Il est conseillé d'utiliser les gabarits Excel pour éviter des erreurs de structures dans les tableaux d'arguments. Les fichiers Excel sont disponibles [ici](#).

Pop_QC	<i>Estimations et projections de population comparables (1996-2041)</i>
--------	---

Description

Tableau de la population québécoise par niveau géographique.

Ce fichier présente une série continue de données populationnelles comparables composée de la série des estimations (1996-2019) et de la série des projections (2020-2041) de population. Ces données tiennent compte de l'évolution de la population selon les plus récentes données observées de naissances, décès et mouvements migratoires.

Il est à noter que ces données de population sont présentées sur la base du découpage territorial du réseau de la santé et des services sociaux, soit pour les territoires suivants : le Québec, les réseaux universitaires intégrés de santé et de services sociaux (RUISSS), les régions sociosanitaires (RSS), les réseaux territoriaux de services (RTS), les réseaux locaux de services (RLS) et les centres locaux de services communautaires (CLSC).

Usage

```
data("Pop_QC") / MaVariable <- Pop_QC.
```

Format

Tableau de 8 variables et 2 595 320 observations :

GEO Niveau géographique : Québec, RUISSS, RSS, RTS, RLS, CLSC.

CODE Code du territoire.

AN Année.

TYPE Type de données : Estimations ou Projections.

STATUT Donnée révisée ou provisoire. NA indique que la donnée n'a pas été changée depuis la dernière publication.

SEXE

AGE

POP Population

Details

La classe des colonnes est character lorsque c'est du texte ou integer lorsque c'est un nombre.

Mise en ligne : 25 février 2016.

Dernière modification : 24 avril 2020.

Publication no : EstimProjComp-ISQ.

La fiche d'information et technique de cette base de données est disponible avec le fichier Excel (voir *Source*).

Source

MSSS Données de population.

Fichier Excel utilisé.

query_stat_gen1	<i>Code SQL</i>
-----------------	-----------------

Description

Générateur de code SQL pour la méthode stat_gen1.

Usage

```
query_stat_gen1(
  debut,
  fin,
  type_Rx = "DENOM",
  codes,
  group_by = "Codes",
  code_serv = c("1", "AD"),
  code_serv_filtre = "Exclusion",
  code_list = NULL,
  code_list_filtre = "Inclusion",
  ...
)
```

Arguments

debut	Date de début de la période d'étude au format 'AAAA-MM-JJ' (une seule valeur).
fin	Date de fin de la période d'étude au format 'AAAA-MM-JJ' (une seule valeur).
type_Rx	Indique le type de code analysé : <ul style="list-style-type: none"> 'DENOM' : Code de dénomination commune (SMED_COD_DENOM_COMNE). 'DIN' : Code d'identification du médicament (SMED_COD_DIN).
codes	Vecteur comprenant le ou les codes d'analyse au format numérique, sans zéros.
group_by	Regrouper (aggréger) les résultats par : <ul style="list-style-type: none"> 'Codes' : Résultats par code analysé. 'Teneur' : Résultats par teneur de médicament (SMED_COD_TENR_MED) incluant les valeurs absentes. 'Format' : Résultats par format d'acquisition du médicament (SMED_COD_FORMA_ACQ_MED) incluant les valeurs absentes.
code_serv	Vecteur de type character comprenant le ou les codes de service (SMED_COD_SERV_1) à exclure ou à inclure, sinon inscrire NULL.
code_serv_filtre	'Inclusion' ou 'Exclusion' des codes de service code_serv. Incrire code_serv = NULL s'il n'y a pas de filtre à appliquer.
code_list	Vecteur de type character comprenant le ou les codes de catégories de liste de médicaments (SMED_COD_CATG_LISTE_MED) à exclure ou à inclure, sinon inscrire NULL.
code_list_filtre	'Inclusion' ou 'Exclusion' des codes de catégories de liste de médicaments code_list. Incrire code_list = NULL s'il n'y a pas de filtre à appliquer.

Details

Méthode stat_gen1 :

Statistiques descriptives tirées de la vue V_DEM_PAINT_MED_CM.

group_by : Revient à utiliser la commande group by dans le code SQL.

code_serv_filtre, code_list_filtre :

"Exclusion" : Inclus les NULL, "Inclusion" exclus les NULL.

Value

Chaîne de caractères à utiliser dans une requête SQL.

Examples

```
### Avantages d'utiliser cat() si c'est pour afficher du code dans la console.
# Avec cat()
cat(query_stat_gen1(
  debut = "2020-01-01", fin = "2020-12-31",
  type_Rx = "DENOM", codes = c(47092, 47135, 48222), group_by = "Codes",
  code_serv = c("1", "AD"), code_serv_filtre = "Exclusion"
))
# Sans cat()
query_stat_gen1(
```

```

    debut = "2020-01-01", fin = "2020-12-31",
    type_Rx = "DENOM", codes = c(47092, 47135, 48222), group_by = "Codes",
    code_serv = c("1", "AD"), code_serv_filtre = "Exclusion"
  )

  ### GROUP_BY
  # group_by = NULL : resultats par periode d'etude.
  cat(query_stat_gen1(
    debut = "2020-01-01", fin = "2020-12-31",
    type_Rx = "DENOM", codes = c(47092, 47135, 48222),
    group_by = NULL,
    code_serv = c("1", "AD"), code_serv_filtre = "Exclusion"
  ))
  # group_by = "Codes" : resultats par code d'analyse.
  cat(query_stat_gen1(
    debut = "2020-01-01", fin = "2020-12-31",
    type_Rx = "DENOM", codes = c(47092, 47135, 48222),
    group_by = "Codes",
    code_serv = c("1", "AD"), code_serv_filtre = "Exclusion"
  ))
  # group_by = c("Teneur", "Format") : resultats par teneur et format.
  cat(query_stat_gen1(
    debut = "2020-01-01", fin = "2020-12-31",
    type_Rx = "DENOM", codes = c(47092, 47135, 48222),
    group_by = c("Teneur", "Format"),
    code_serv = c("1", "AD"), code_serv_filtre = "Exclusion"
  ))

  ### Exclusion VS Inclusion
  cat(query_stat_gen1(
    debut = "2020-01-01", fin = "2020-12-31",
    type_Rx = "DENOM", codes = c(47092, 47135, 48222),
    group_by = NULL,
    code_serv = c("1", "AD"), code_serv_filtre = "Exclusion",
    code_list = c("40", "41"), code_list_filtre = "Inclusion"
  ))

```

RLS_convert

Correspondance RLS Loi 10

Description

Établir la correspondance des RLS avant et après l'adoption de la loi 10.

Usage

```
data("RLS_convert") / MaVariable <- RLS_convert.
```

Format

Tableau de 2 variables et 84 observations :

Details

Certains RLS ne peuvent être convertis, car leur valeur se retrouve avant et après l'adoption de la loi 10.

`attr(RLS_convert, "RLS_exclus")` indique les quatre (4) RLS exclus : 611, 612, 1611, 1612.

`attr(RLS_convert, "RLS_exclus_value")` renvoie un tableau indiquant les valeurs avant et après l'adoption de la loi 10 pour ces quatre (4) RLS.

Source

Correspondance Etablissement Public Loi 10.
Fichier Excel utilisé.

RLS_list	Liste des RLS
----------	---------------

Description

Vecteur contenant la liste des 93 RLS plus 3 valeurs utiles lors d'analyse : 1001, 1701, 1801.

Usage

```
data("RLS_list") / MaVariable <- RLS_list.
```

Format

Vecteur integer de 96 nombres.

SQL_connexion	Connexion Teradata
---------------	--------------------

Description

Connexion entre R et SQL Teradata.

Usage

```
SQL_connexion(uid, pwd = NULL, dsn = "PEI_PRD", encoding = "latin1")
```

Arguments

<code>uid</code>	Identifiant.
<code>pwd</code>	Mot de passe. Si NULL, le mot de passe est demandé lors de l'exécution.
<code>dsn</code>	Data Source Name. Par défaut "PEI_PRD".
<code>encoding</code>	"latin1" ou "UTF-8". Encodage de la base de données. Par défaut "latin1".

Details

Voir `?odbc::dbConnect`.

Value

Connexion Teradata, sinon NULL.

Examples

```
## Not run:
conn <- SQL_connexion("abc007")
conn <- SQL_connexion(uid = "abc007", pwd = "MonMotDePasse", dsn = "PEI_PRD")

## End(Not run)
```

SQL_stat_gen1

*Statistiques générales***Description**

Tableau indiquant les statistiques générales d'un ou de plusieurs codes de médicaments selon certains critères.

Usage

```
SQL_stat_gen1(
  conn = NULL,
  uid = NULL,
  pwd = NULL,
  debut,
  fin,
  type_Rx = "DENOM",
  codes,
  group_by = "Codes",
  code_serv = c("1", "AD"),
  code_serv_filtre = "Exclusion",
  code_list = NULL,
  code_list_filtre = "Inclusion",
  ...
)
```

Arguments

conn	Variable contenant la connexion entre R et Teradata. Voir SQL_connexion .
uid	Nom de l'identifiant pour la connexion SQL Teradata.
pwd	Mot de passe associé à l'identifiant. Si NULL, le programme demande le mot passe. Cela permet de ne pas afficher le mot de passe dans un script.
debut	Date de début de la période d'étude au format 'AAAA-MM-JJ' (une seule valeur).
fin	Date de fin de la période d'étude au format 'AAAA-MM-JJ' (une seule valeur).
type_Rx	Indique le type de code analysé : <ul style="list-style-type: none"> 'DENOM' : Code de dénomination commune (SMED_COD_DENOM_COMNE). 'DIN' : Code d'identification du médicament (SMED_COD_DIN).
codes	Vecteur comprenant le ou les codes d'analyse au format numérique, sans zéros.

group_by	Regrouper (aggréger) les résultats par : <ul style="list-style-type: none"> • 'Codes' : Résultats par code analysé. • 'Teneur' : Résultats par teneur de médicament (SMED_COD_TENR_MED) incluant les valeurs absentes. • 'Format' : Résultats par format d'acquisition du médicament (SMED_COD_FORMA_ACQ_MED) incluant les valeurs absentes.
code_serv	Vecteur de type character comprenant le ou les codes de service (SMED_COD_SERV_1) à exclure ou à inclure, sinon inscrire NULL.
code_serv_filtre	'Inclusion' ou 'Exclusion' des codes de service code_serv. Inscrire code_serv = NULL s'il n'y a pas de filtre à appliquer.
code_list	Vecteur de type character comprenant le ou les codes de catégories de liste de médicaments (SMED_COD_CATG_LISTE_MED) à exclure ou à inclure, sinon inscrire NULL.
code_list_filtre	'Inclusion' ou 'Exclusion' des codes de catégories de liste de médicaments code_list. Inscrire code_list = NULL s'il n'y a pas de filtre à appliquer.

Examples

```
## Not run:
### GROUP_BY
# Aucun group by
dt1 <- SQL_stat_gen1(
  conn = conn,
  debut = c("2017-01-01", "2018-01-01"), fin = c("2017-12-31", "2018-12-31"),
  type_Rx = "DENOM", codes = c(39, 47092, 48222), group_by = NULL
)
# Codes - DENOM
dt2 <- SQL_stat_gen1(
  conn = conn,
  debut = c("2017-01-01", "2018-01-01"), fin = c("2017-12-31", "2018-12-31"),
  type_Rx = "DENOM", codes = c(39, 47092, 48222), group_by = "Codes"
)
# Codes - DIN
dt3 <- SQL_stat_gen1(
  conn = conn,
  debut = c("2017-01-01", "2018-01-01"), fin = c("2017-12-31", "2018-12-31"),
  type_Rx = "DIN", codes = c(30848, 585092), group_by = "Codes"
)
# Codes et Format
dt4 <- SQL_stat_gen1(
  conn = conn,
  debut = c("2017-01-01", "2018-01-01"), fin = c("2017-12-31", "2018-12-31"),
  type_Rx = "DENOM", codes = c(39, 47092, 48222), group_by = c("Codes", "Format")
)
# Teneur
dt5 <- SQL_stat_gen1(
  conn = conn,
  debut = c("2017-01-01", "2018-01-01"), fin = c("2017-12-31", "2018-12-31"),
  type_Rx = "DENOM", codes = c(39, 47092, 48222), group_by = "Teneur"
)

### Exclusion & Inclusion
```



```
dt6 <- SQL_stat_gen1(  
  conn = conn,  
  debut = c("2017-01-01", "2018-01-01"), fin = c("2017-12-31", "2018-12-31"),  
  type_Rx = "DENOM", codes = c(39, 47092, 48222), group_by = "Codes",  
  code_serv = c("1", "AD"), code_serv_filtre = "Exclusion",  
  code_list = c("03", "40", "41"), code_list_filtre = "Inclusion"  
)  
  
## End(Not run)
```

Index

* datasets

Pop_QC, [2](#)

RLS_convert, [5](#)

RLS_list, [6](#)

formulaire, [2](#)

Pop_QC, [2](#)

query_stat_gen1, [3](#)

RLS_convert, [5](#)

RLS_list, [6](#)

SQL_connexion, [6](#), [7](#)

SQL_stat_gen1, [7](#)