

Description

inesss v.1.0.0.9000

Statistiques d'un ou de plusieurs codes de médicaments selon certains critères.
Vue utilisée : V_DEM_PAIMT_MED_CM¹.

Usage

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

Arguments

<code>conn</code>	Variable contenant la connexion entre R et Teradata. Voir <code>SQL.connexion</code> .
<code>debut</code>	Vecteur contenant la ou les dates de début des périodes d'étude au format AAAA-MM-JJ.
<code>fin</code>	Vecteur contenant la ou les dates de fin des périodes d'étude au format AAAA-MM-JJ.
<code>type_Rx</code>	Type de code à analyser. Une valeur parmi : <ul style="list-style-type: none">• 'AHFS' : Code identifiant la classe de médicaments telle que déterminée par l'<i>American Hospital Formulary Service</i>.• 'DENOM' : Code de dénomination commune.• 'DIN' : Code d'identification du médicament.
<code>codes</code>	Le ou les codes à analyser. Voir <i>Details</i> .
<code>group_by</code>	Équivalent du <i>group by</i> SQL. Regrouper (aggréger) les résultats par : <ul style="list-style-type: none">• 'AHFS' : Code identifiant la classe de médicaments telle que déterminée par l'<i>American Hospital Formulary Service</i>.

¹http://intranet/eci/ECI2/ASP/ECI2P04_DescVue.asp?Envir=PROD&NoVue=1823&NomVue=V\%5FDEM\%5FPAIMT\%5FMED\%5FCM+\%28DEMANDES+DE+PAIEMENT+\%2D+PROGRAMME+\%ABMEDICAMENT\%BB\%29

	<ul style="list-style-type: none"> • 'DENOM' : Code de dénomination commune. • 'DIN' : Code d'identification du médicament. • 'CodeList' : Code de catégorie de liste de médicament. • 'CodeServ' : Code de service. • 'Teneur' : Teneur du médicament. • 'Format' : Format d'acquisition du médicament. • 'Age' : Âge à une date précise. Combiner avec l'argument <code>age_date</code>.
<code>code_serv</code>	Le ou les codes de services à exclure ou inclure, sinon inscrire NULL. <code>character</code> .
<code>code_serv_filtre</code>	'Exclusion' ou 'Inclusion' des codes de services.
<code>code_list</code>	Le ou les codes de catégorie de listes de médicaments à exclure ou inclure, sinon inscrire NULL. <code>character</code> .
<code>code_list_filtre</code>	'Exclusion' ou 'Inclusion' des codes de catégories de listes de médicaments.
<code>age_date</code>	Date à laquelle on calcul l'âge des individus. À utiliser seulement si <code>group_by</code> contient 'Age'.

Details

debut, fin :

`debut` et `fin` doivent contenir le même nombre de valeurs.

codes :

Si `type_Rx='AHFS'` : codes sous la forme de 6 caractères où les deux premiers caractères représente la classe AHFS, les deux du milieu la sous-classe AHFS et les deux derniers la sous-sous-classe AHFS. Il est possible de remplacer une paire de caractères (`{1, 2}`, `{3, 4}` ou `{5, 6}`) par `--` pour rechercher toutes les types de classes. Par exemple, `'04--12'` indique qu'on recherche la classe AHFS 04, toutes les sous-classes AHFS et la sous-sous-classe 12.

Sinon : inscrire les codes sous la forme d'un nombre entier.

code_serv_filtre, code_list_filtre :

'Exclusion' inclus les NULL

'Inclusion' exclus les NULL.

Nom des médicaments :

Que ce soit pour les codes AHFS (`NOM_AHFS`), les DENOM (`NOM_DENOM`) ou les DIN (`NOM_MARQ_COMRC`), le nom inscrit est toujours celui le plus récent.

Value

`data.table`

Examples

```
conn <- SQL_connexion()
```

```

### group_by
# Aucun
ex01 <- SQL_stat_gen1(
  conn, debut = c('2018-01-01', '2019-01-01'), fin = c('2018-12-31', '2019-12-31'),
  type_Rx = 'DENOM', codes = c(39, 47092, 47135), group_by = NULL
)
# Tous les group_by
ex02 <- SQL_stat_gen1(
  conn, debut = c('2018-01-01', '2019-01-01'), fin = c('2018-12-31', '2019-12-31'),
  type_Rx = 'DENOM', codes = c(39, 47092, 47135),
  group_by = c('AHFS', 'DENOM', 'DIN', 'CodeList', 'CodeServ', 'Teneur', 'Format', 'Age')
)

### AHFS
ex03 <- SQL_stat_gen1(
  conn, debut = c('2018-01-01', '2019-01-01'), fin = c('2018-12-31', '2019-12-31'),
  type_Rx = 'AHFS', codes = c('04----', '08--12', '122426'), group_by = 'AHFS'
)

### DENOM
ex04 <- SQL_stat_gen1(
  conn, debut = c('2018-01-01', '2019-01-01'), fin = c('2018-12-31', '2019-12-31'),
  type_Rx = 'DENOM', codes = c(39, 47092, 47135), group_by = c('DENOM', 'DIN')
)

### DIN
ex05 <- SQL_stat_gen1(
  conn, debut = c('2018-01-01', '2019-01-01'), fin = c('2018-12-31', '2019-12-31'),
  type_Rx = 'DIN', codes = c(30848, 585092), group_by = 'DIN'
)

### Age
ex06 <- SQL_stat_gen1(
  conn, debut = c('2018-01-01', '2019-01-01'), fin = c('2018-12-31', '2019-12-31'),
  type_Rx = 'DIN', codes = c(30848, 585092), group_by = c('DIN', 'Age'), age_date = '2018-01-01'
)

### Exclusion et Inclusion code_serv et code_list
ex07 <- SQL_stat_gen1(
  conn, debut = c('2018-01-01', '2019-01-01'), fin = c('2018-12-31', '2019-12-31'),
  type_Rx = 'DENOM', codes = c(39, 47092, 47135), group_by = 'DENOM',
  code_serv = c('1', 'AD'), code_serv_filtre = 'Exclusion',
  code_list = c('40', '41'), code_list_filtre = 'Inclusion'
)

```