

1 Création des tables de comparaisons

Les programmes sont destinés à être utilisés avec PostgreSQL. Ils sont compatibles au moins avec les versions 11.10 et 12 (liste non exhaustive).

Pour créer les opérateurs d'égalités, vous devez charger le fichier `create-table_comp.sql` en entier. Vous devez également charger vos tables ainsi que vos fonctions de comparaisons et vos fonctions d'interprétations pour chaque attribut dont vous souhaitez relaxer l'égalité.

Note : pour un attribut `attrib`, le prototype de sa fonction de comparaison doit se présenter comme : `fattrib(type_attrib, type_attrib)`, et le prototype de sa fonction d'interprétation se présente comme : `interprattrib(bit)`. La fonction de comparaison renvoie des bits et la fonction d'interprétation renvoie un booléen.

Exemple :

```
CREATE OR REPLACE FUNCTION flevel(x numeric, y numeric)
RETURNS bit AS $$
SELECT
  CASE
    WHEN (x=y) or (0<=x and x<2) THEN b'11'
    WHEN ((2<=x and x<5) AND (2<=y and y<5)) THEN b'10'
    WHEN ((0<=x and x<2) AND (2<=y and y<5)) THEN b'01'
    ELSE b'00'
  END
$$ LANGUAGE sql;

CREATE OR REPLACE FUNCTION interprlevel(b bit)
RETURNS boolean AS $$
SELECT
  CASE
    WHEN b='11' or b='10' THEN true
    ELSE false
  END
$$ LANGUAGE sql;
```

Il faut lancer la procédure `createtable_comp` avec comme paramètre une chaîne de caractères avec les tables à utiliser, délimitées par une virgule si il y en a plusieurs. Par exemple : `call createtable_comp('r, iris')` génère les tables `r_comp` et `iris_comp` ainsi que les nouveaux opérateurs pour ces tables.

Pour chaque attribut, si il existe une fonction de comparaison ET une fonction d'interprétation, alors un nouveau type composite est créé pour cet attribut avec ces opérateurs. Les types des attributs de la table de comparaison pourront être des types de bases ou des types créés.

Vous pouvez maintenant faire vos requêtes :

```
select * from iris_comp where sepal_l=5 renverra la même chose que :  
select * from iris_comp where interprsepal_l(fsepal_l(sepal_l, 5)).
```

```
select * from iris_comp where sepal_l!=5 renverra la même chose que :  
select * from iris_comp where not interprsepal_l(fsepal_l(sepal_l,  
5)).
```

Note : L'opérateur d'inégalité peut s'écrire != ou <>.

Les opérateurs créés entre deux mêmes types composites sont == et != et
<>= :

```
select *  
from iris1_comp as a, iris2_comp as b  
where a.sepal_l==b.sepal_l;
```

```
select *  
from iris1_comp as a, iris2_comp as b  
where a.sepal_l!=b.sepal_l;
```

```
select *  
from iris1_comp as a, iris2_comp as b  
where a.sepal_l<=>b.sepal_l;
```

Ces 2 dernières requêtes renvoient exactement la même chose.

Mais si vous oubliez de mettre le dernier égal, la requête vous renverra quand même une table avec l'opérateur de base du SGBD.