

LIFBDW2 : PL/pgSQL – Fonctions

L'objectif de ce TP est de se familiariser avec la programmation procédurale sous Postgresql, notamment via la langage PL/pgSQL.

1. Consulter la documentation en ligne sur PL/pgSQL :

- <https://www.postgresqltutorial.com/postgresql-plpgsql/> (ou)
- <https://docs.postgresql.fr/12/plpgsql.html>
- <https://www.w3resource.com/PostgreSQL/pl-pgsql-tutorial.php>

2. Premiers pas :

- Écrire une fonction **hello_pl** qui renvoie la chaîne de caractère « Hello DB World! » en PL/pgSQL.

```
CREATE OR REPLACE FUNCTION hello_pl()  
RETURNS text  
AS $BODY$  
BEGIN  
    - Votre instruction  
END  
$BODY$  
LANGUAGE plpgsql;
```

- Écrire en PL/pgSQL une fonction de division appelée **division**. Elle acceptera en entrée deux arguments de type entier et renverra un nombre réel (numeric).
- Comment corriger le problème de la division par zéro ? Écrire cette nouvelle fonction dans les deux langages. (*Conseil : dans ce genre de calcul impossible, il est possible d'utiliser la constante NaN (Not A Number)*).
- Écrire une fonction de résolution d'une équation du second degré dans **R** ($ax^2 + bx + c = 0$)
- Écrire une fonction inverser qui inverse une chaîne (pour « lifbdw2 » en entrée, afficher « 2wdbfil » en sortie), à l'aide d'une boucle WHILE et des fonctions **char_length** et **substring**.

3. Sélection sur des tables.

On s'appuiera sur le script de création des tables :

https://perso.liris.cnrs.fr/marc.plantevit/ENS/LIFBDW2/TP/script_tp_table_pgsql.sql

- Créer une fonction qui retourne l'estimation du nombre de ligne d'une table d'un schéma passés en paramètres (il faudra interroger la table **pg_stat_user_tables**) :

```
CREATE OR REPLACE FUNCTION nb_lignes_table (sch text, tbl text)  
RETURNS bigint  
...
```

- Similairement, définir une fonction qui retourne le nombre total des tuples estimés sur un schéma donné.
- Créer une fonction qui ramène le nombre d'employés embauchés une année donnée (à partir du champ **employes.date_embauche**).

- d. Utiliser la fonction **generate_series()** pour lister le nombre d'embauches pour chaque année entre 2000 et 2020.
- e. Créer une fonction qui fait la même chose avec deux années en paramètres une boucle FOR ... LOOP, RETURNS TABLE et RETURN NEXT.
- f. Écrire une fonction **departement** qui admet un numéro d'employé en paramètre et qui renvoie comme résultat le nom du département de l'employé.
- g. Écrire une fonction **collegues** qui à partir d'un numéro d'employé renvoie comme résultat le nom et le prénom de ses collègues (membres du même département), lui-même ne devant pas faire partie de la liste de ses collègues.