

Manipulation d'une base de données PostgreSQL avec les triggers / procédures stockées

A partir de la base tp et des relations 'villes' et 'routes'.

1 Avec plpgsql

- Créez une relation villesTrigger qui contient le schéma de villes avec un attribut supplémentaire, portionRoutiere, de type GEOMETRY.
- Créez une procédure, constructionRoutiere, en plpgsql qui permet de stocker dans l'attribut portionRoutiere, la partie urbaine de la route qui traverse la ville considérée.
- Créez un trigger, trig_partieRoutiere qui sur une insertion dans villesTrigger vient remplir automatiquement l'attribut portionRoutiere à l'aide de la procédure constructionRoutiere.
- Chargez la relation villesTrigger à partir des tuples de la relation villes.
- Vérifiez que l'attribut portionRoutiere a bien été rempli.

```
id | nom |          portionroutiere
----+-----+-----
 1 | V1 | LINESTRING(1 2,4 3.63636363636364)
 2 | V2 | LINESTRING(6 4.72727272727273,8 5.81818181818182)
 3 | V3 |
(3 rows)
```

- Supprimez les éléments créés.

2 Avec le langage C

- Créez une relation routesTrigger qui contient le schéma de routes avec un attribut supplémentaire, longueur, de type réel.
- Créez un trigger, trig_longueurRoutiere, qui va lors de l'insertion d'un tuple dans routesTrigger va déclencher l'exécution de la fonction constructionlongueur.
- Créez une fonction, constructionlongueur, en C qui va venir remplir l'attribut longueur avec la longueur de la route associée au tuple.
- Chargez la relation routesTrigger à partir des tuples de la relation routes.
- Vérifiez que l'attribut longueur a bien été rempli.

```
id | nom | longueur
----+-----+-----
 1 | R1 | 15.529964
 2 | R2 |      2
 3 | R3 |      2
(3 rows)
```

- Supprimez les éléments créés.