

## ULB – Stage SIG libres – Fiche d'exercice pratique

Nom : Malongo Kokinambili

Prénom : yves

Année :2019

### Mise en place de la base de données spatiales sur vos bons plans belges

#### Description et objectifs de l'exercice

Afin d'intégrer vos bons plans insolites, vous allez transférer les tracés de vos balades et les différents points d'intérêts collectés par GPS dans la base de données commune bons\_plans sur le serveur de l'IGEAT. Dans le cadre d'un travail en 3 groupes de stagiaires (Groupe A/B & C) il vous est ainsi demandé de :

- Actualiser la base de données commune des bons plans belges des stagiaires en SIG libres
  - Dans votre schéma, implémenter les données utiles de votre groupe et regrouper les dans une table commune à votre groupe
  - Ajouter des données supplémentaires (les nouvelles données non encore reprises dans le schéma common\_data)
  - Visualiser la localisation de l'ensemble des bons plans ainsi que les balades effectuées par les membres de votre groupe
- Élaborer des requêtes
  - Requêtes imposées ci-dessous pour exercice
  - Requêtes personnelles en fonction des besoins
- Construire des vues affichables dans QGIS et créer un projet QGIS des bons plans
  - Vues imposées ci-dessous pour exercice
  - Vues personnelles en fonction des besoins
- Création d'un formulaire QGIS pour l'interrogation et l'ajout / mise à jour de nouveaux points d'intérêt et balades

**Données disponibles :** [Données GPS et shapefiles des balades et pigns](#), [photos des pigns](#), [données non-spatiales \(auteurs et tables glossaires\)](#), fonds de cartes (de services web et autres données vectorielles disponibles en ligne voir e.a : <https://data.gov.be/en> ou <https://www.atlas-belgique.be>)

#### Création de la connexion avec le serveur de bases de données (tomahawk)

Nom : tomahawk-bons\_plans\_group\_a/b/c (*à personnaliser*)

Hôte : 172.17.150.243 (vs pour autoriser une connexion extérieure à l'IGEAT : 164.15.12.243)

Base maintenance : bons\_plans

Nom utilisateur : group\_a/b/c & Mot de passe : (*donnés en fonction du groupe*)

**Lien vers le cours théorique** si besoin et pour une vue d'ensemble de l'exercice , vous pouvez vous référer au document **projet transversal dans la section WebGIS du site du stage.**

## Résultats attendus :

Il vous est demandé de consigner sur cette fiche, l'ensemble des résultats obtenus, une description succincte des procédures ainsi que les difficultés rencontrées pour arriver à :

- Réaliser un schéma conceptuel final de la mise en place de la base de données et un schéma entité-association & schéma opérationnel (modélisation finale de la structure des données sous Dia)
- Charger et compléter vos données dans le serveur de bases de données du labo (tomahawk) :
  - *schéma « commun\_data » doit contenir les tables suivantes : auteurs, type\_pig, type\_balade, photos & difficultés avec données de l'ensemble des stagiaires : [https://framacalc.org/=bons\\_plans\\_Belg](https://framacalc.org/=bons_plans_Belg) + communes\_belges & reseaux\_belges (avec les données de <http://portal.openbelgium.be/dataset>)*
  - *schémas « group\_a/b/c » : pig, balades et balades\_pig (+ en fonction du modèle choisi auteurs\_pig et auteurs\_balades) avec respectivement les données des groupes A/B/C + table avec données spatiales belges supplémentaires trouvées par le groupe cf. <http://cafesig.ulb.ac.be/mod/page/view.php?id=1002>*
- Élaborer des requêtes afin de répondre aux questions suivantes:
  - a) Quels sont les noms et la durée des balades effectuées à pied ?
  - b) Quel est le nom du stagiaire ayant la balade la plus longue ?
  - c) Quelle est la longueur (en kilomètres) des balades de votre groupe ?
  - d) Quelles sont les longueurs maximum et minimum (en mètre) entre deux points d'intérêt géographique ?
  - e) Quels sont les noms des communes traversées par les balades de votre groupe ?
  - f) Quels sont les noms des 3 communes ayant le plus d'intérêts ?
  - i) Quels sont les noms des stations de métro à moins de 500m des balades de votre groupe ?
  - j) Créer la colonne distance de la table balades sur base de la longueur calculée
  - k) Requêtes personnelles en fonction des besoins
- Affichage de couches PostGIS pour la réalisation de cartes suivantes :
  - a) Localisation des bons plans sélectionnés par les stagiaires du groupe (une carte permettant la localisation et valorisation des points d'intérêt ainsi que les balades effectuées par les stagiaires)
  - b) Mise en évidence de l'ensemble des communes non attractives pour les stagiaires (une carte via création de vue/couche virtuelle permettant la localisation de toutes les communes n'ayant pas de bons plans renseignés)
  - c) Identification des communes par attrait thématiques (une carte permettant d'afficher la quantité de bons plans par commune et par thématique majoritaire)
  - d) Au moins une autre carte issue d'une ou de plusieurs autres vues ou couches virtuelles au choix
- Création d'un formulaire QGIS pour l'ajout ou la mise à jour de nouveaux points d'intérêts et balades
- Projet QGIS avec les couches des balades, des pig et de vos différentes vues/couches virtuelles et fonds de plan belge au choix.
- RAPPEL : Il vous est également demandé de bien compléter (avec schéma E-A et logique description méthodologique + fichier requête de votre groupe & projet QGIS (avec formulaire) + illustration de votre formulaire QGIS) la section « Base de données » de la VUE « Bons plans » de votre e-portfolio (via ajout d'éléments fichiers de la sous-sous catégorie « Base de données » de la sous catégorie « Bons plans » de la catégorie « Visites ») : [http://cafesig.ulb.ac.be/blocks/exaport/shared\\_view.php?access=hash/49-f75070b7](http://cafesig.ulb.ac.be/blocks/exaport/shared_view.php?access=hash/49-f75070b7)

**Astuces :** *Quelques rappels d'opérateurs et fonctions SQL pouvant vous être utiles :*

- Opérateurs de comparaisons

Opérateur	Description
=	Égale
<>	Pas égale
!=	Pas égale
>	Supérieur à
<	Inférieur à
>=	Supérieur ou égale à
<=	Inférieur ou égale à
IN	Liste de plusieurs valeurs possibles
BETWEEN	Valeur comprise dans un intervalle donnée (utile pour les nombres ou dates)
LIKE	Recherche en spécifiant le début, milieu ou fin d'un mot.
IS NULL	Valeur est nulle
IS NOT NULL	Valeur n'est pas nulle
NOT	N'est pas

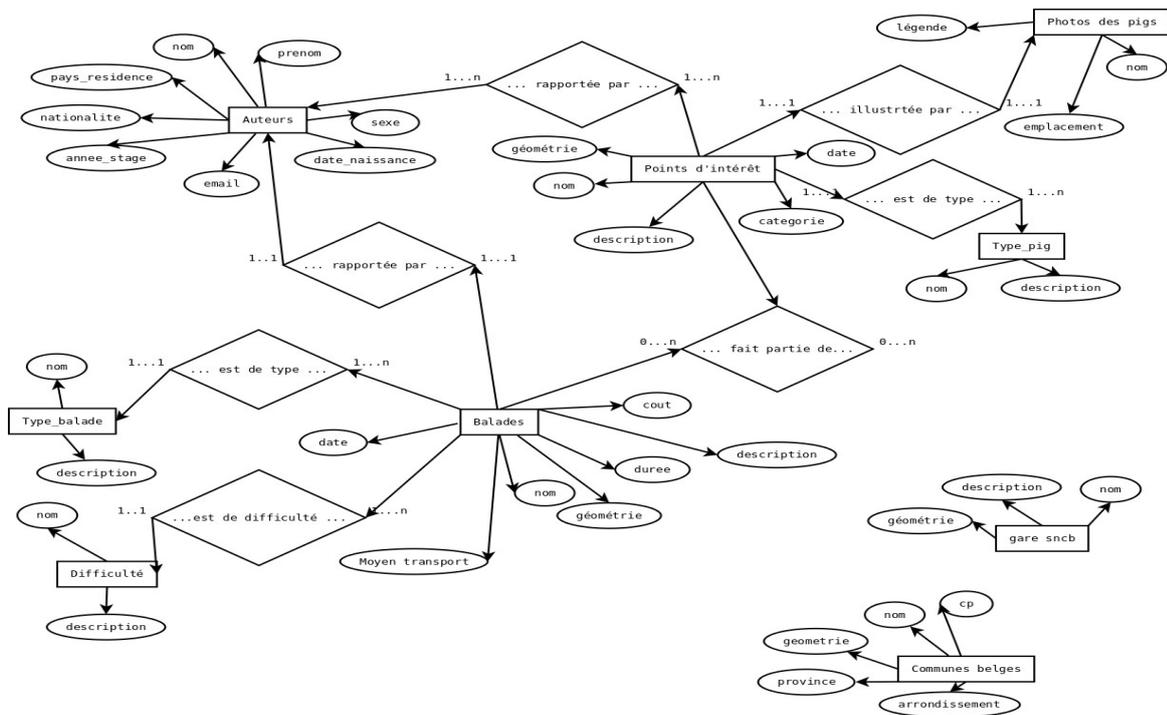
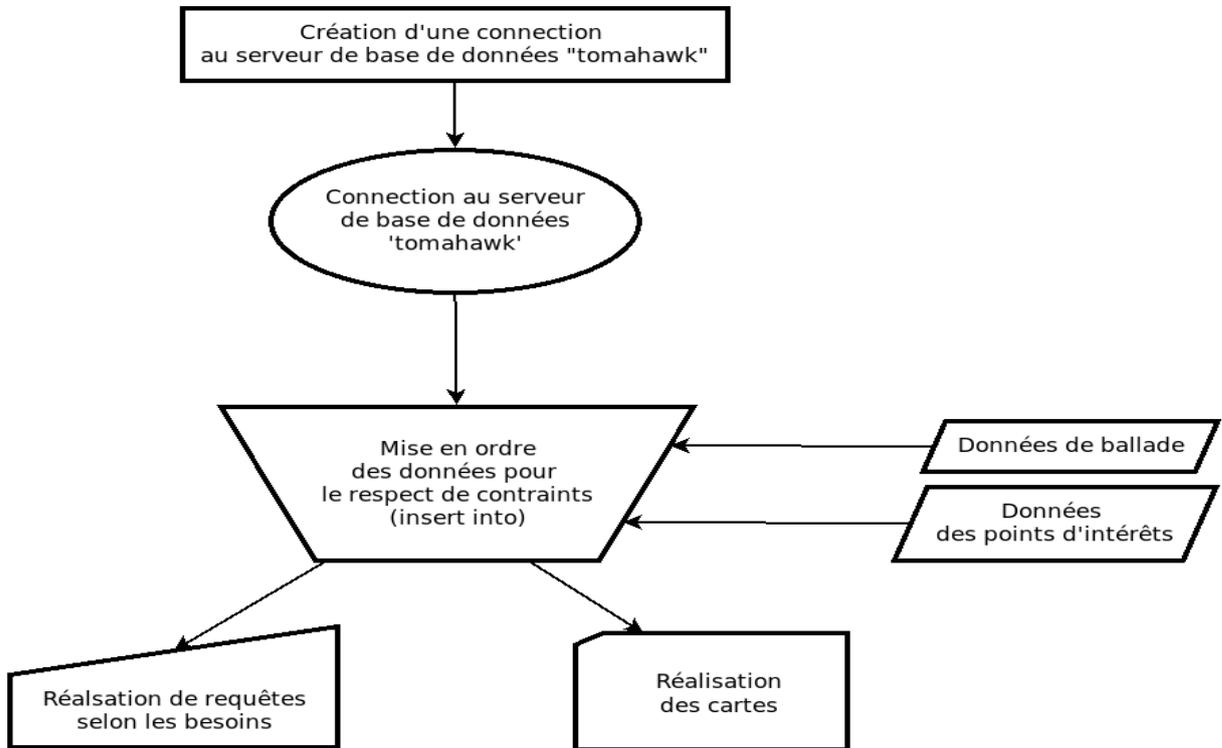
- Fonctions SQL

- *SUM(colonne A)* retourne la somme totale de la colonne A
- *MAX(colonne A)* retourne le maximum de la colonne A
- *MIN(colonne A)* retourne le minimum de la colonne A
- *COUNT(colonne A)* retourne le nombre de lignes dans la colonne A

- Fonctions SQL spatiales :

- *ST\_X(point)* retourne la coordonnée X
- *ST\_Y(point)* retourne la coordonnée Y
- *ST\_Point(float X, float Y)* retourne la géométrie d'un point avec les coordonnées X, Y
- *ST\_SetSRID(geometry A, integer srid)* affecte le système de coordonnées srid (correspondant à un code EPSG) à la géométrie A
- *ST\_Transform(geometry A, integer srid)* retourne la géométrie résultant de la reprojction de la géométrie A (qui doit avoir un système de projection affecté) vers le système de coordonnées srid (correspondant à un code EPSG)
- *ST\_Length(linestring)* retourne la longueur d'une ligne
- *ST\_Area(geometry)* retourne l'aire d'un polygone
- *ST\_Contains(geometry A, geometry B)* retourne TRUE si la géométrie A contient la géométrie B
- *ST\_Buffer(geometry A, float d)* retourne la géométrie d'un tampon de rayon d autour de la géométrie A
- *ST\_Distance(geometry A, geometry B)* retourne la distance minimum entre deux géométries
- *ST\_DWithin(geometry A, geometry B, radius)* retourne TRUE si la A est distante d'au plus radius de B
- *ST\_Intersects(geometry A, geometry B)* retourne TRUE si la géométrie A intersecte la géométrie B
- *ST\_Touches(geometry A, geometry B)* retourne TRUE si le contour extérieur de A touche B
- *ST\_Within(geometry A, geometry B)* retourne TRUE si A est hors de B
- *ST\_Intersection (geometry A, geometry B)* retourne une géométrie qui représente l'intersection de l'ensembles de points des géométries A et B
- *ST\_Difference (geometry geomA, geometry geomB)* retourne une géométrie représentant la partie de la géométrie A qui ne recoupe pas la géométrie B.

➔ Schéma théorique de la solution (indépendamment d'un logiciel)



## **Choix du/des logiciel(s) et des outils à utiliser dans ce(s) logiciel(s)**

– QGIS

– Post Gis

– Gps (garmin)

– Serveur IGEAT

– BDD bons\_plans

– SQL

## Résultats (y compris résultats intermédiaires) sous forme de cartes et/ou chiffres (chaque carte doit contenir le nom du groupe)

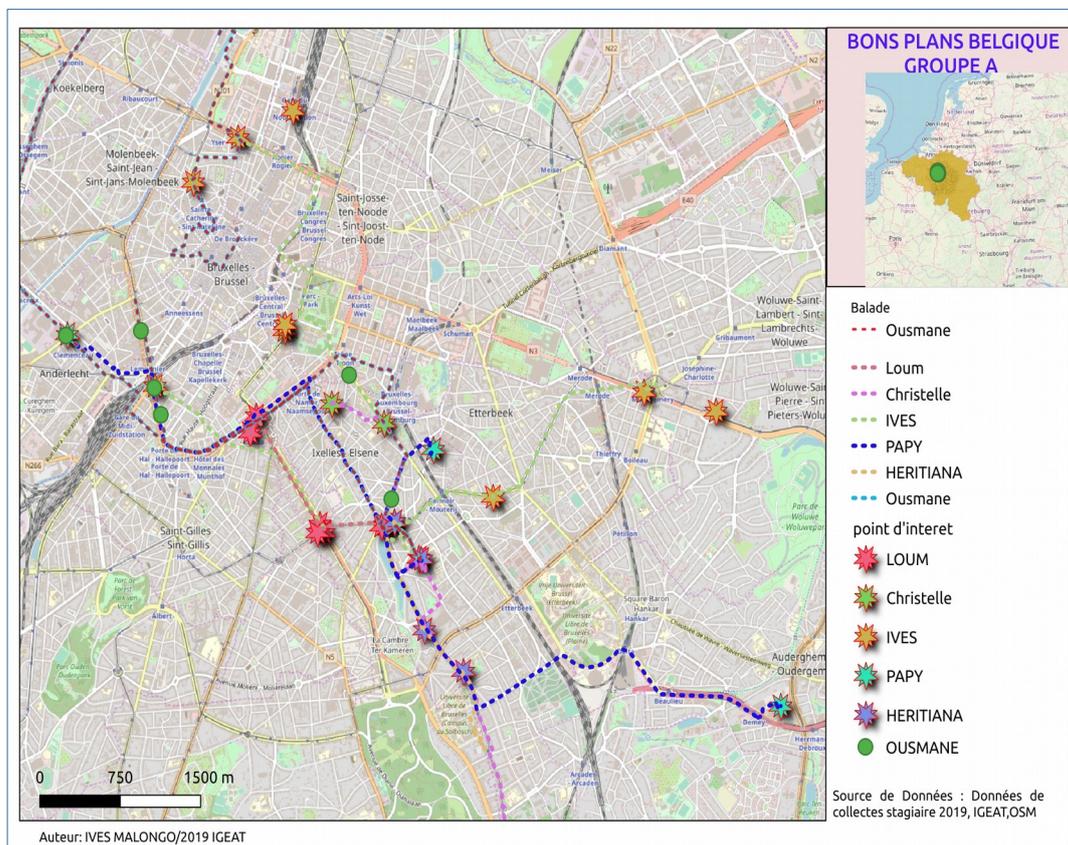
Dans cette partie, il vous est également demandé de fournir l'explication détaillée sur au moins l'élaboration de 3 requêtes spatiales imposées et une non imposée en SQL

1 – Réalisation d'une base des données des balades et pigs du groupe

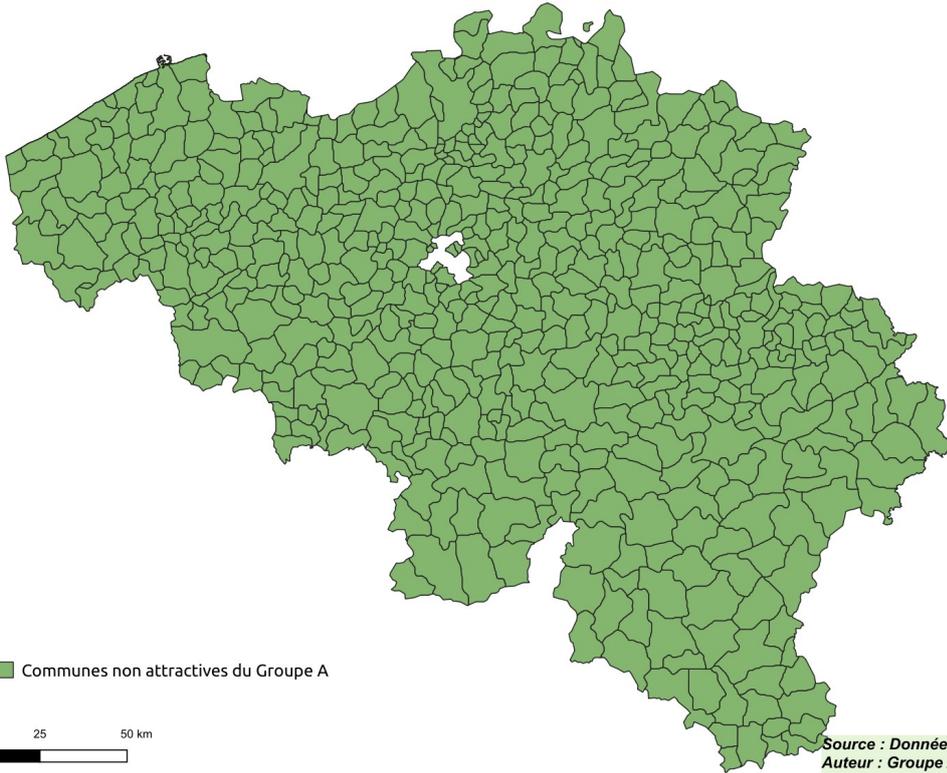
2 – Réalisation de la carte des balades et pigs

3 – Réalisation de la vue « Commune non attractive »

## Balades et Pigs



## Communes non attractives pour le groupe a



## FORMULAIRE

**balades - Attributs d'entités**

Attribut	Valeur
id	NULL
nom	NULL
description	NULL
duree	NULL
mode_transport	NULL
cout	NULL
date	NULL
id_type_balade	NULL
id_difficulte	NULL
id_auteur	NULL
distance	NULL

**balade\_pig**

Attribut	Valeur
id	
nom	NULL
description	NULL
id_type_pig	NULL
id_photo	NULL
id_auteur	NULL
date	NULL
categorie	NULL

**Résultats de l'identification**

ID	Valeur
id	201903001
id	201903001
nom	montgomery
de...	espace public ouvert
du...	360,0000000000000000
m...	tram
cout	10
date	2019-08-11
id...	3
id...	2
id...	201903
di...	3528,88

## **1. Commentaires & Notes**

Évaluation de la base de données spatiales développée

→ avantages / désavantages

→ **difficultés éventuelles**

- Importation des données des balades et pigns
- Accès au serveur
- Incohérence des données (projection, format)
- Plantage du module Db manager de QGIS
- Difficulté à mettre en exécution le formulaire avec les relations

**Astuces / techniques intéressantes découvertes**

Mise à jour des tables (regrouper les attributs pour faciliter le traitement)

Temps mis pour résoudre l'exercice tout entier : 13h

## FICHE REQUÊTES GROUPE\_A

--a) Quels sont les noms et la durée des balades effectuées à pied ?

```
UPDATE group_a.balades
  SET mode_transport='mixte'
  WHERE group_a.balades.mode_transport = 'pied; bus; metro' or
group_a.balades.mode_transport = 'pied, tram' or group_a.balades.mode_transport = 'bus_metro' ;
```

```
UPDATE group_a.balades
  SET mode_transport='pied'
  WHERE group_a.balades.mode_transport = 'à pied';
```

```
Select id,nom, duree, mode_transport
from group_a.balades
where group_a.balades.mode_transport = 'pied';
```

--b) Quel est le nom du stagiaire ayant la balade la plus longue ?

```
select common_data.auteurs.nom, max(st_length(geom))
from group_a.balades join common_data.auteurs on group_a.balades.id_auteur =
common_data.auteurs.id
group by common_data.auteurs.nom;
```

```
select common_data.auteurs.nom
from common_data.auteurs join group_a.balades on group_a.balades.id_auteur =
common_data.auteurs.id
where st_length(geom) = (select max(st_length(geom))
from group_a.balades) ;
```

--c) Quelle est la longueur (en kilomètres) des balades de votre groupe ?

```
select round((sum(st_length(geom))/1000)::numeric,2) as dist_gpe_a_km
from group_a.balades;
```

--d) Quelles sont les longueurs maximum et minimum (en mètre) entre deux points d'intérêt géographique ?

```
select round(min(st_distance(p1.geom,p2.geom))::numeric,2) as dist_min,
round(max(st_distance(p1.geom,p2.geom))::numeric,2) as dit_max
from group_a.pig p1,group_a.pig p2
where not(p1.geom = p2.geom);
```

--e) Quels sont les noms des communes traversées par les balades de votre groupe ?

```
select distinct c.name_4
from common_data.communes_belges c join group_a.balades b on st_intersects(b.geom,c.geom);
```

--f) Quels sont les noms des 3 communes ayant le plus d'intérêts ?

```
select distinct c.name_4, count (p.id) as nbre_pig
from common_data.communes_belges c join group_a.pig p on st_within(p.geom,c.geom)
group by c.name_4
order by nbre_pig desc;
```

--i) Quels sont les noms des stations de métro à moins de 500m des balades de votre groupe ?

```
select distinct g.name
from common_data."gares-sncb" g join group_a.balades b on st_dwithin(g.geom_31370,
b.geom,500);
```

--j) Créer la colonne distance de la table balades sur base de la longueur calculée

```
alter table group_a.balades add column distance double precision;
update group_a.balades set distance = round(st_length(geom)::numeric,2);
```

## --k) Requêtes personnelles en fonction des besoins

Christelle

--Nombre de pigs par auteurs

```
select a.prenom,count(p.id)
from common_data.auteurs a join group_a.pig p on a.id = p.id_auteur
group by a.prenom;
```

--Liste des communes visitées par auteur

```
select distinct a.prenom, c.name_4
      from common_data.auteurs a
      join group_a.balades b on a.id = b.id_auteur
      join common_data.communes_belges c on st_intersects(b.geom,c.geom)
```

--Nombre de communes visitées par les stagiaires

```
select x.prenom, count(*)
from
      (select distinct a.prenom, c.name_4
      from common_data.auteurs a
      join group_a.balades b on a.id = b.id_auteur
      join common_data.communes_belges c on st_intersects(b.geom,c.geom)) x
group by x.prenom;
```

Heritiana

--Quels sont les noms du patrimoine bâti dans la region?

```
Select id,nom
from group_a.pig
where id_type_pig = 6;
```

--Dans quelles régions se localisent les patrimoines bâtis?

```
Select pig.nom,common_data.communes_belges.name_4
from pig
join common_data.communes_belges on st_within(pig.geom,communes_belges.geom)
where id_type_pig = 6;
```

LOUM :

--Quels sont les noms des personnes et des balades effectués dans la commune d'Elsene

```
SELECT distinct auteurs.nom, balades.nom
FROM balades
join common_data.communes_belges on ST_intersects(balades.geom,communes_belges.geom)
join common_data.auteurs on auteurs.id =balades.id_auteur
WHERE name_4='Elsene';
```

Papy

--Quelle est la distance minimale de la balade dans le groupe ?

```
select min(long)
from
      (SELECT balades.nom, st_length(geom) as long
      FROM balades) x;
```

Diarra

--Quel est le nom du stagiaire ayant la balade la plus petite?

```
select common_data.auteurs.nom
from common_data.auteurs join group_a.balades on group_a.balades.id_auteur =
common_data.auteurs.id
where st_length(geom) = (select min(st_length(geom))
from group_a.balades) ;
```

--Quelle est la longueur (en kilomètres) des balades de Ousmane (id\_auteur= 201906)?

```
select round((sum(st_length(geom))/1000)::numeric,2) as dist_gpe_a_km
from group_a.balades where id_auteur=201906;
```

## IVES

sélectionnez les noms de tous les pigs et les noms de communes qu'ils traversent de l'auteur ives select p.nom, c.name\_4 from group\_a.pig as p join common\_data.communes\_belges as c on st\_within (p.geom,c.geom) where p.id\_auteur=201903

## Vues créées

--b) Ensemble des communes non attractives pour les stagiaires

--(une vue permettant la localisation de toutes les communes n'ayant pas de bons plans renseignés)

--Liste de toutes les communes

```
select distinct name_4 from common_data.communes_belges
```

--Listes des communes attractives (pigs et balades)

```
select distinct x.name_4, x.geom
from (select distinct c.name_4, c.geom
      from common_data.communes_belges c
      join group_a.pig p on st_within(p.geom,c.geom)
      union
      select distinct c.name_4, c.geom
      from common_data.communes_belges c
      join group_a.balades b on st_intersects(b.geom,c.geom)) x
```

--Listes des communes non attractives

```
select distinct c.name_4, c.geom
      from common_data.communes_belges c
      except
select distinct x.name_4, x.geom
from (select distinct c.name_4, c.geom
      from common_data.communes_belges c
      join group_a.pig p on st_within(p.geom,c.geom)
      union
      select distinct c.name_4, c.geom
      from common_data.communes_belges c
      join group_a.balades b on st_intersects(b.geom,c.geom)) x
;
```

--c) Identification des communes par attrait thématiques (une carte permettant d'afficher la quantité de bons plans par commune et par thématique majoritaire)

```
select name_4, nom, nb from
(select name_4, nom, nb, row_number() over(partition by name_4 order by nb desc) rn
      from
      (select c.name_4,tb.nom, count(tb.description) nb
      from common_data.communes_belges c join group_a.balades b on st_intersects
      (b.geom,c.geom)
      join common_data.type_balades tb on tb.id = b.id_type_balade
      group by c.name_4, tb.nom
      order by count (tb.description) desc) t
      ) x
```

where rn = 1  
order by nb desc, name\_4