

EXERCICES SIG :QGIS/POSTGIS

Ce document présente les exercices pratiques réalisés avec QGIS et PostGIS. Il contient les requêtes SQL, les explications détaillées et la méthodologie. Les captures d'écran des résultats seront ajoutées ultérieurement.

Exercice 6 : Requêtes SQL avec DBManager

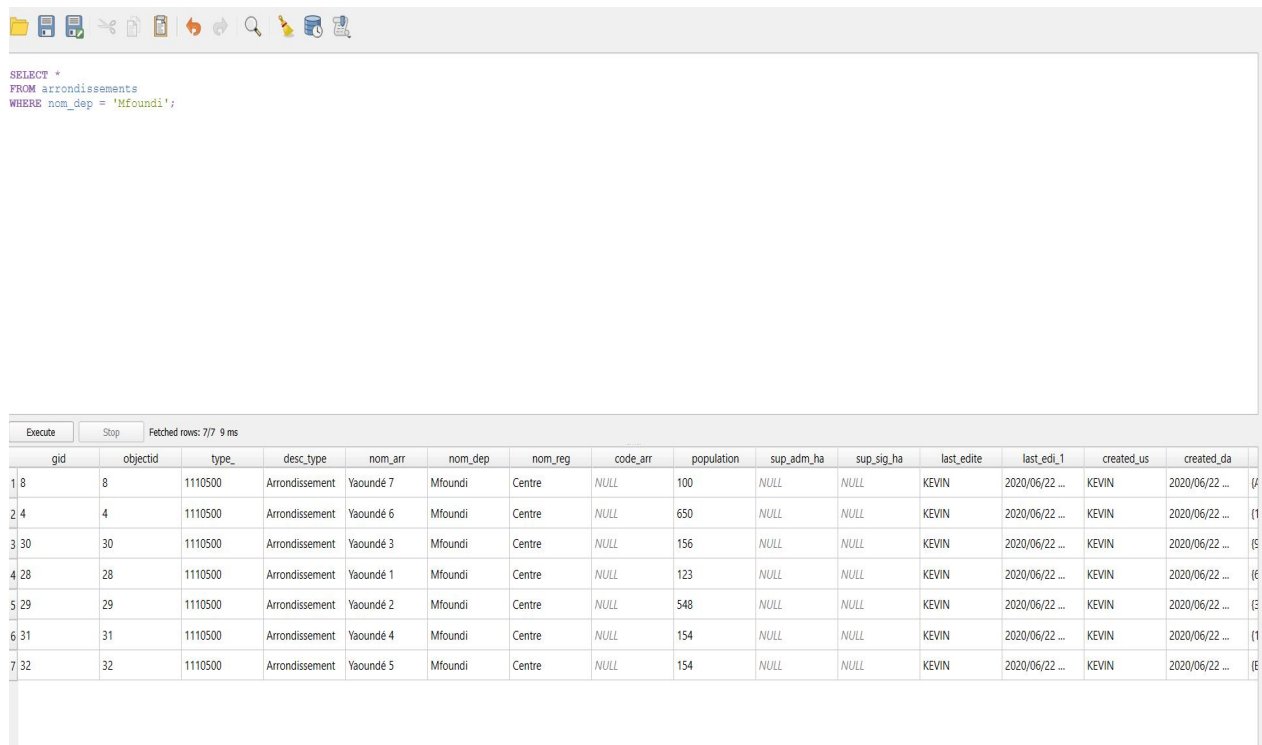
Question 1

Objectif : Sélectionner tous les arrondissements du département du Mfoundi.

Requête SQL :

```
SELECT *  
FROM arrondissements  
WHERE nom_dep = 'Mfoundi';
```

Explication : Cette requête permet de filtrer les arrondissements appartenant uniquement au département du Mfoundi.



The screenshot shows a database management interface. At the top, there is a toolbar with various icons. Below the toolbar, a SQL query is entered in a text area:

```
SELECT *  
FROM arrondissements  
WHERE nom_dep = 'Mfoundi';
```

Below the query, there is a table with 16 columns and 7 rows of data. The columns are: gid, objectid, type_, desc_type, nom_arr, nom_dep, nom_reg, code_arr, population, sup_adm_ha, sup_sig_ha, last_edite, last_edt_1, created_us, and created_da. The rows represent different arrondissements in the Mfoundi department.

	gid	objectid	type_	desc_type	nom_arr	nom_dep	nom_reg	code_arr	population	sup_adm_ha	sup_sig_ha	last_edite	last_edt_1	created_us	created_da
1	8	8	1110500	Arrondissement	Yaoundé 7	Mfoundi	Centre	NULL	100	NULL	NULL	KEVIN	2020/06/22 ...	KEVIN	2020/06/22 ...
2	4	4	1110500	Arrondissement	Yaoundé 6	Mfoundi	Centre	NULL	650	NULL	NULL	KEVIN	2020/06/22 ...	KEVIN	2020/06/22 ...
3	30	30	1110500	Arrondissement	Yaoundé 3	Mfoundi	Centre	NULL	156	NULL	NULL	KEVIN	2020/06/22 ...	KEVIN	2020/06/22 ...
4	28	28	1110500	Arrondissement	Yaoundé 1	Mfoundi	Centre	NULL	123	NULL	NULL	KEVIN	2020/06/22 ...	KEVIN	2020/06/22 ...
5	29	29	1110500	Arrondissement	Yaoundé 2	Mfoundi	Centre	NULL	548	NULL	NULL	KEVIN	2020/06/22 ...	KEVIN	2020/06/22 ...
6	31	31	1110500	Arrondissement	Yaoundé 4	Mfoundi	Centre	NULL	154	NULL	NULL	KEVIN	2020/06/22 ...	KEVIN	2020/06/22 ...
7	32	32	1110500	Arrondissement	Yaoundé 5	Mfoundi	Centre	NULL	154	NULL	NULL	KEVIN	2020/06/22 ...	KEVIN	2020/06/22 ...

Question 2

Objectif : Sélectionner les arrondissements du Mfoundi ayant plus de 4000 habitants.

Requête SQL :

```
SELECT nom_arr, population  
FROM arrondissements  
WHERE nom_dep = 'Mfoundi'  
AND population > 4000;
```

Explication : On applique deux conditions avec AND : le département et la population.

```
SELECT nom_arr, population  
FROM arrondissements  
WHERE nom_dep = 'Mfoundi'  
AND population > 4000;
```

Execute

Stop

Query executed successfully (0 rows, 1 ms)

Question 3

Objectif : Sélectionner les arrondissements dont le statut n'est pas 'arrondissement simple'.

Requête SQL :

```
SELECT nom_arr AS NOM, statut, population, superficie  
FROM arrondissements  
WHERE statut <> 'arrondissement simple';
```

Explication : L'opérateur <> signifie différent de.

```
SELECT nom_arr AS NOM, statut, population, superficie
FROM arrondissements
WHERE statut <> 'arrondissement simple';
```

Execute	Stop	Fetched rows: 17/17 11 ms		
	nom	statut	population	superficie
1	Yaoundé 7	Chef lieu de ...	100	0,...
2	Nkongsamba 2	Chef lieu arrondis	150	0,...
3	Mayo-Baléo	Chef lieu arrondis	156	0,...
4	Tignère	Chef lieu de ...	214	0,...
5	Yaoundé 6	Chef lieu arrondis	650	0,...
6	Kai-Kai	Arrondis simple	200	0,...
7	Galim-Tignère	Arrondis simple	100	0,...
8	Kontcha	Arrondis simple	300	0,...
9	Bankim	Chef lieu de ...	125	0,...
10	Ngaoundal	Chef lieu arrondis	123	0,...
11	Tibati	Chef lieu de ...	125	0,...
12	Banyo	Chef lieu de ...	332	0,...
13	Ngaoundéré 2	Chef lieu arrondis	125	0,...
14	Mayo-Darley	Arrondis simple	157	0,...
15	Dir	Arrondis simple	154	0,...
16	Nkongsamba 3	Chef lieu de ...	125	0,...
17	Djohong	Arrondis simple	654	0,...

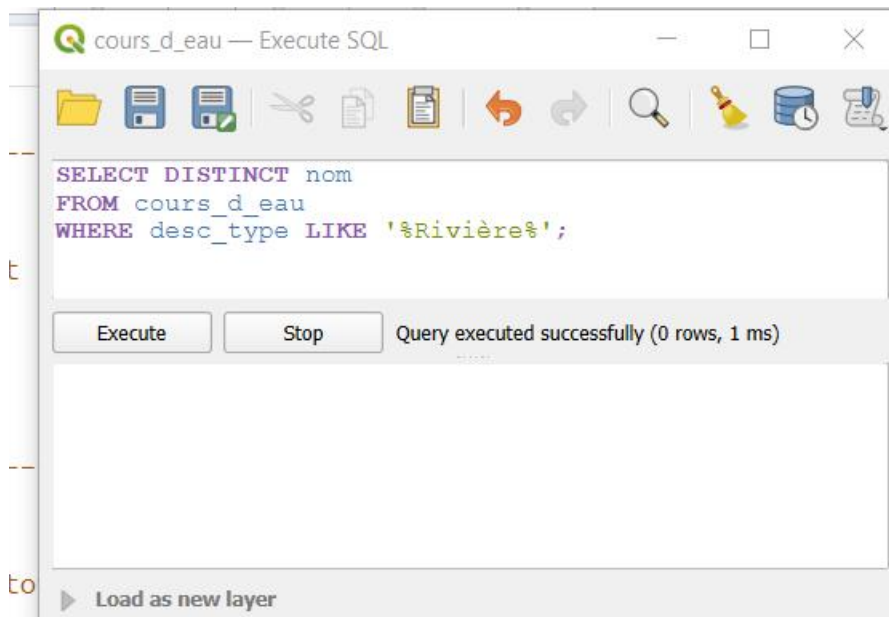
Question 4

Objectif : Lister les cours d'eau contenant le mot 'Rivière'.

Requête SQL :

```
SELECT DISTINCT nom
FROM cours_d_eau
WHERE desc_type LIKE '%Rivière%';
```

Explication : LIKE permet de rechercher une chaîne de caractères partielle.



Question 5

Objectif : Calculer les statistiques démographiques par département.

Requête SQL :

```
SELECT nom_dep,
SUM(population) AS population_totale,
ROUND(AVG(population/superficie),2) AS densite_moyenne,
MAX(population) AS population_max,
MIN(population) AS population_min,
AVG(superficie) AS superficie_moyenne
FROM arrondissements
GROUP BY nom_dep;
```

Explication : Les fonctions d'agrégation sont utilisées avec GROUP BY.

Explication : Les fonctions ST_Area et ST_Perimeter travaillent en mètres, d'où la conversion.

```
SELECT nom_arr,
       ROUND((ST_Area(geom)/1000000)::numeric, 2) AS surface_km2,
       ROUND((ST_Perimeter(geom)/1000)::numeric, 2) AS perimetre_km
FROM arrondissements;
```

Execute Stop Fetched rows: 360/360 23 ms

	nom_arr	surface_km2	perimetre_km
1	Ngaoui	0	0
2	Nsem	0	0
3	Makenene	0	0
4	Yoko	0	0,01
5	Ndikinimeki	0	0
6	Soa	0	0
7	Yaoundé 7	0	0
8	Nkongsamba 2	0	0
9	Mayo-Baléo	0	0
10	Tignère	0	0
11	Kikki	0	0
12	Ndobian	0	0
13	Ndéléélé	0	0
14	Lomié	0	0,01
15	Mindourou	0	0
16	Tchatibali	0	0
17	Wina	0	0
18	Makary	0	0

► Load as new layer

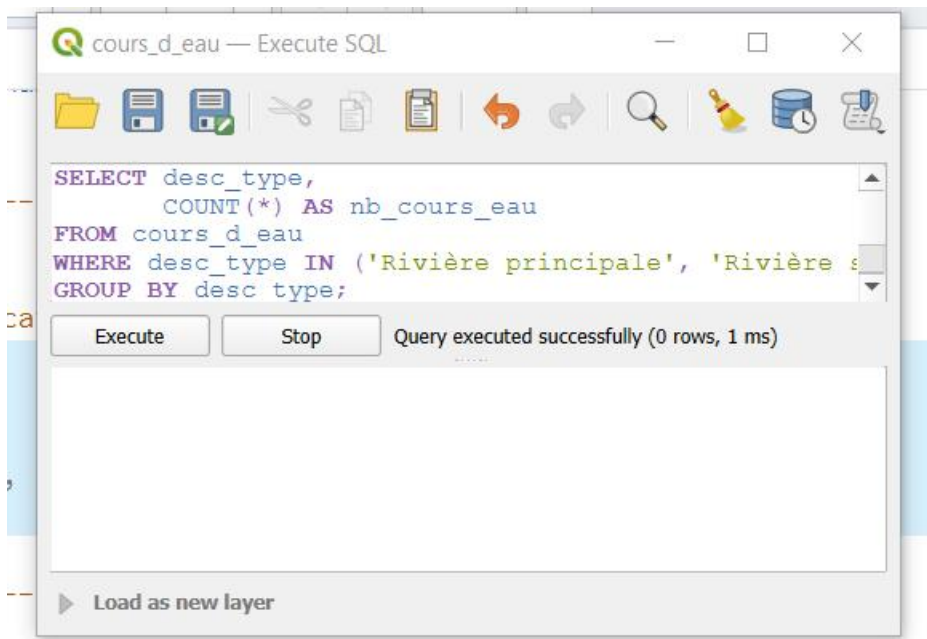
Question 7

Objectif : Compter le nombre de cours d'eau par catégorie.

Requête SQL :

```
SELECT desc_type, COUNT(*) AS nombre
FROM cours_d_eau
GROUP BY desc_type;
```

Explication : COUNT permet de compter les entités par type.



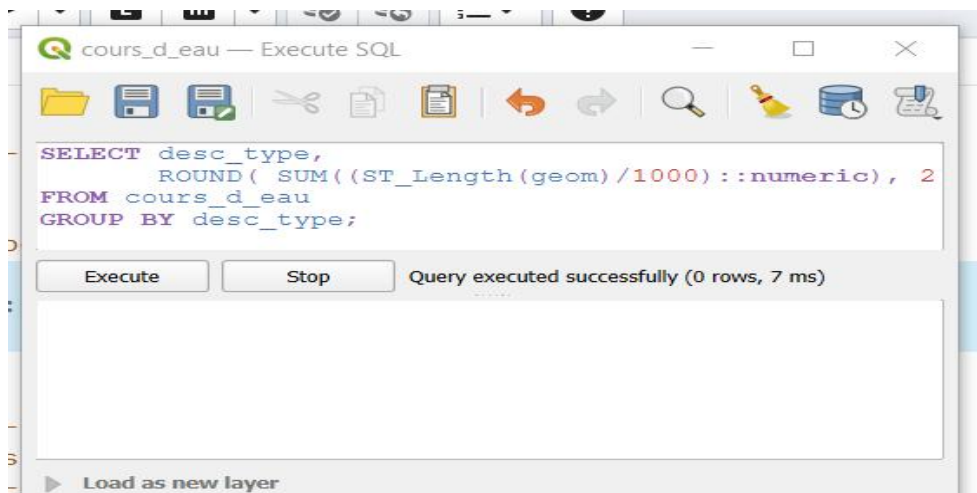
Question 8

Objectif : Calculer la longueur totale des cours d'eau par type.

Requête SQL :

```
SELECT desc_type,
       SUM(ST_Length(geom)/1000) AS longueur_km
FROM cours_d_eau
GROUP BY desc_type;
```

Explication : ST_Length calcule la longueur des objets linéaires.



Conclusion

Ces exercices ont permis de maîtriser les bases de l'utilisation de QGIS avec PostGIS, notamment les requêtes SQL spatiales, les agrégations et l'analyse géographique.