

# Rapport TP OpenStreetMap

## Table des matières

I.Requêtes SQL :	2
Question 1 :	2
Question 2.a :	2
Question 2.b :	2
Question 3 :	3
Question 4 :	3
Question 5.a :	4
Question 5.b :	6
Question 5.c :	6
Question 6 (bonus) :	7
Question 7 :	8
II.Documentation de l'application :	9
Question 8 :	9
Question 9 :	9
Question 10 (alternatives a et b) :	10
Question 11 (suggestion a) :	11
III.Manuel Utilisateur.....	13

Charlotte TROTTA

Nathalie MIBORD

Emilie LIBESSART

Nicolas BERILLON

# I. Requêtes SQL :

## **Question 1 :**

*Compter le nombre de contributeurs pour les données de la région Rhône-Alpes (= nombre de tuples de la relation users).*

```
SELECT count(*) AS nb_distributeurs FROM users;
```

nb_distributeurs
4576

-----

4576

(1 ligne)

## **Question 2.a :**

*Quelles sont les coordonnées géographiques du domaine universitaire de Grenoble ?*

```
SELECT ST_X(geom) AS longitude, ST_Y(geom) AS latitude FROM nodes WHERE  
id=1787038609;
```

longitude   latitude
5.7680106   45.192893

-----+-----

5.7680106 | 45.192893

(1 ligne)

## **Question 2.b :**

*Dans quel système de référence ces coordonnées sont-elles exprimées ?*

```
SELECT auth_srid AS srid, auth_name AS name FROM spatial_ref_sys, nodes WHERE  
id=1787038609 AND ST_SRID(geom) = srid;
```

srid   name
4326   EPSG

-----+-----

4326 | EPSG

(1 ligne)

**Question 3 :**

Quelles sont les coordonnées géographiques du centroïde de la mairie de Grenoble (la mairie est un chemin qui contient l'attribut "amenity"=>"townhall" et dont le nom contient "Grenoble").

```
SELECT ST_X(ST_Centroid(bbox)) AS longitude, ST_Y(ST_Centroid(bbox)) AS latitude FROM
ways WHERE tags->'amenity'='townhall' AND (tags->'name' LIKE '%Grenoble%');
```

longitude	latitude
5.73644115	45.18644215

(1 ligne)

**Question 4 :**

Compter le nombre de routes (chemins contenant la clef "highway") par type (par valeur de l'attribut "highway"), ordonné par ordre décroissant.

```
SELECT tags->'highway' AS highway, COUNT(*) AS nb_routes FROM ways WHERE
tags?'highway' GROUP BY tags->'highway' ORDER BY COUNT(*) DESC;
```

highway	nb_routes
residential	94356
unclassified	77678
service	64461
track	58276
tertiary	21923
footway	21742
path	21413
secondary	18444
primary	11970
pedestrian	4002
steps	3355
cycleway	2505
motorway	2210
road	1976
motorway_link	1956
living_street	1907
platform	1821
trunk	1045
trunk_link	1017
primary_link	763
bridleway	320
tertiary_link	295
secondary_link	270
construction	195

raceway		48
proposed		40
rest_area		33
services		26
via_ferrata		18
bus_stop		16
conveyor		8
private		7
elevator		7
corridor		5
bus_guideway		5
crossing		4
abandoned		3
proposal		2
turning_circle		2
emergency_access_point		2
unclassified;service		1
cycleway;path		1
unclassified;track		1
tertiary;secondary		1
residential;unclassified		1
pathway		1
yes		1
passing_place		1
ford		1
alley		1
residential;track		1
(51 lignes)		

**Question 5.a :**

Même question que précédemment (question 4), mais au lieu de compter les routes, affichez leur longueur.

```
SELECT tags->'highway' AS highway, SUM(ST_Length(linestring)) AS length FROM ways
WHERE tags?'highway' GROUP BY tags->'highway' ORDER BY SUM(ST_Length(linestring))
DESC;
```

highway		length
-----+-----		
unclassified		435.285712879273
track		343.366696791793
residential		210.396982404606
tertiary		208.294200085326
path		134.367018999509
secondary		120.622408690095
service		62.5645575726099
primary		53.3422148486077

motorway		28.6715936154116
footway		26.9145189180964
cycleway		10.1213381544952
road		8.37733244501455
trunk		7.09821321050042
motorway_link		5.51177937675859
pedestrian		4.73331637143003
bridleway		2.85057359795041
living_street		2.49893965509429
trunk_link		2.12683675252511
construction		1.27360036717929
primary_link		1.1340883449525
steps		0.790884454795759
tertiary_link		0.412146721856399
proposed		0.405155153438685
corridor		0.384494201037925
services		0.342651295516932
platform		0.323979669146777
raceway		0.302529637202219
rest_area		0.257651954958723
secondary_link		0.229150230092211
via_ferrata		0.0354335168638014
residential;unclassified		0.0221886942773748
proposal		0.0209007795790456
unclassified;track		0.0111607302543396
bus_stop		0.00972746337138419
tertiary;secondary		0.00678690105795661
private		0.0065058917846346
pathway		0.00578500038720849
abandoned		0.00555497862918651
cycleway;path		0.00541457054121882
residential;track		0.00457510050326216
bus_guideway		0.00382350615509007
elevator		0.00286261406357175
unclassified;service		0.00207177328855967
turning_circle		0.00152729118701463
conveyor		0.00137332309839854
yes		0.000967514906313659
alley		0.000928515144842681
passing_place		0.00088163470610578
ford		0.000780944588306162
emergency_access_point		0.000780703162049341
crossing		0.00035376927717115
(51 lignes)		

**Question 5.b :**

En quelle unité cette longueur est-elle exprimée ?

```
SELECT srid, srtext AS unite FROM spatial_ref_sys, ways WHERE ST_SRID(line
string)=srid LIMIT 1;
```

srid	unite
4326	GEOGCS["WGS 84",DATUM["WGS_1984",SPHEROID["WGS 84",6378137,298.257223563,AUTHORITY["EPSG","7030"]],AUTHORITY["EPSG","6326"],PRIMEM["Greenwich",0,AUTHORITY["EPSG","8901"]],UNIT["degree",0.0174532925199433,AUTHORITY["EPSG","9122"]],AUTHORITY["EPSG","4326"]]

(1 ligne)

L'unité de cette longueur est le **degré**.

**Question 5.c :**

Même question, mais avec toutes les longueurs converties dans le système métrique (km si possible).

```
SELECT tags->'highway' AS highway,
SUM(ST_Length(ST_Transform(linestring,4326)::geography))/1000 AS length FROM ways
WHERE tags?'highway' GROUP BY tags->'highway' ORDER BY
sum(ST_length(ST_Transform(linestring,4326)::geography)) DESC;
```

highway	length
unclassified	39840.3847465004
track	31548.5308675041
residential	19217.248646164
tertiary	19092.5143218903
path	12385.5371652032
secondary	11054.7544485223
service	5721.88752214514
primary	4905.59039039862
motorway	2645.76412625689
footway	2452.9424121304
cycleway	927.600411902203
road	769.272454468655
trunk	647.685572912772
motorway_link	505.818144374077
pedestrian	430.80963640836
bridleway	257.050912255378
living_street	229.069247325745
trunk_link	193.487275288256
construction	117.639306249841
primary_link	103.318919551506

steps		71.86176885469
tertiary_link		36.9880580178088
proposed		35.369600898467
corridor		34.7364695145201
services		32.3508890077913
platform		30.0639354207091
raceway		27.4755623767283
rest_area		24.1065931803798
secondary_link		21.1688219702772
via_ferrata		3.15112400231986
residential;unclassified		2.05310559011799
proposal		1.86149018896854
bus_stop		1.03321353461398
unclassified;track		1.02255260611962
private		0.609209323081245
tertiary;secondary		0.540034761122943
pathway		0.458675475737785
cycleway;path		0.45578564221882
abandoned		0.454916822085196
bus_guideway		0.381385712138663
residential;track		0.381302829031696
elevator		0.293071330312212
unclassified;service		0.175376399581823
turning_circle		0.138258611412091
conveyor		0.128737535725735
alley		0.0934480974760791
ford		0.0856883509068673
passing_place		0.0802450829762093
emergency_access_point		0.0789614870720364
yes		0.0753841162546393
crossing		0.0335306240181216
(51 lignes)		

**Question 6 (bonus) :**

Quelle est l'aire totale de l'Ensimag en m² ?

```
SELECT SUM( ST_AREA(ST_Transform(bbox,4326)::geography)) AS aire_total_ensimag
FROM ways WHERE tags->'amenity'='university' AND tags->'name' like '%Ensimag%';
```

aire\_total\_ensimag

-----

3829.31086280942

(1 ligne)

**Question 7 :**

*Affichez l'ensemble des quartiers de Grenoble, avec, pour chaque quartier, le nombre d'écoles ("amenity"=>"school") qu'il contient, le tout ordonné par nombre d'écoles décroissant.*

```
SELECT q.quartier, COUNT(w) AS nb_ecoles FROM quartier q, ways w WHERE  
w.tags->'amenity'='school' AND ST_Intersects(ST_Transform(q.the_geom, 4326), w.bbox)  
GROUP BY q.quartier ORDER BY COUNT(w) DESC;
```

quartier	nb_ecoles
-----+-----	
BERRIAT ST BRUNO	13
CENTRE VILLE	12
EXPOSITION-BAJATIERE	11
RONDEAU-LIBERATION	8
MALHERBE	7
ALPINS-ALLIERS	7
ABBAYE-JOUHAUX	6
VILLENEUVE1	6
EAUX CLAIRES	6
JEAN MACE	5
VILLENEUVE2	5
CHAMPIONNET	4
CAPUCHE GR	4
NOTRE DAME	3
FOCH AIGLE	3
VILLAGE-OLYMPIQUE	3
CENTRE GARE	3
TEISSEIRE	2
POLYGONNE	1
ILE VERTE	1
MUTUALITE	1
SAINT-LAURENT	1
MISTRAL-DRAC	1
(23 lignes)	



## II. Documentation de l'application :

### **Question 8 :**

Créez un petit programme de test permettant simplement de vous connecter à la base de données OSM via Java / JDBC, et vérifiez que tout fonctionne bien.

Pour tester cette question, lancez l'application avec comme argument le chiffre 8. Dans la console s'affichera alors les coordonnées du centroïde de la mairie de Grenoble ainsi que sa bbox.

Voici un aperçu du résultat :

```
janv. 29, 2014 9:38:33 PM database.Utills createConnection
Infos: Loading PostgreSQL driver...
janv. 29, 2014 9:38:33 PM database.Utills createConnection
Infos: Trying to connect to Grenoble database...
janv. 29, 2014 9:38:34 PM database.Utills createConnection
Infos: Connected.
Voici les coordonnées (en WGS84) du centroïde et la bbox de la mairie de Grenoble :
    longitude centroide = 5.73644115
    latitude centroide = 45.18644215
    bbox = SRID=4326;POLYGON((5.7357904 45.1860426,5.7357904 45.1868417,5.7370919
45.1868417,5.7370919 45.1860426,5.7357904 45.1860426))
janv. 29, 2014 9:38:47 PM database.Utills closeConnection
Infos: Connection closed.
```

### **Question 9 :**

Écrivez un petit programme de test prenant un argument en ligne de commande, et affichant tous les noms et coordonnées géographiques des points dont le nom ressemble à (au sens du LIKE SQL) l'argument.

Requête utilisée pour cette question :

```
SELECT tags->'name' AS nom, ST_X(geom) AS longitude, ST_Y(geom) AS latitude FROM
nodes WHERE tags->'name' LIKE 'Dom__ne _niversit%';
```

nom	longitude	latitude
-----	+-----	-----
Domaine Universitaire	5.758102	45.1874865
Domaine Universitaire	5.7569486	45.187046
Domaine Universitaire	5.7680106	45.192893

(3 lignes)

Pour tester cette question, lancez l'application avec comme premier argument le chiffre 9 et comme second argument le nom du lieu pour lequel vous cherchez des coordonnées. Dans la console s'affichera les lieux accompagnés de leurs coordonnées, correspondant au nom du lieu passé en paramètre.

Voici un aperçu du résultat lors d'un test avec l'argument "Dom\_\_ne \_niversit%" :

```
janv. 29, 2014 9:52:52 PM database.Utills createConnection
Infos: Loading PostgreSQL driver...
janv. 29, 2014 9:52:52 PM database.Utills createConnection
Infos: Trying to connect to Grenoble database...
janv. 29, 2014 9:52:53 PM database.Utills createConnection
Infos: Connected.
nom = Domaine Universitaire
longitude = 5.758102
latitude = 45.187486500000006
nom = Domaine Universitaire
longitude = 5.7569486
latitude = 45.187046
nom = Domaine Universitaire
longitude = 5.7680106
latitude = 45.192893000000005
janv. 29, 2014 9:53:14 PM database.Utills closeConnection
Infos: Connection closed.
```

**Question 10 (alternatives a et b) :**

Pour tester cette question, lancez l'application avec comme argument le nombre 10.

**(alternative a) :** L'ensemble des routes autour de Grenoble (longitudes comprises entre 5.7 et 5.8, et latitudes comprises entre 45.1 et 45.2).

Requête utilisée pour cette question :

```
SELECT ST_Transform(linestring, 2154) AS linestring, tags->'highway' AS highway FROM ways
WHERE tags->'highway' LIKE '%' AND ST_Intersects(linestring,
ST_SetSRID(ST_GeomFromText('Polygon((5.8 45.1, 5.8 45.2, 5.7 45.2, 5.7 45.1, 5.8
45.1))'),4326));
```

Pour faciliter l'observation des routes, nous avons décidé de colorer les principaux types de routes selon ce code de couleur :

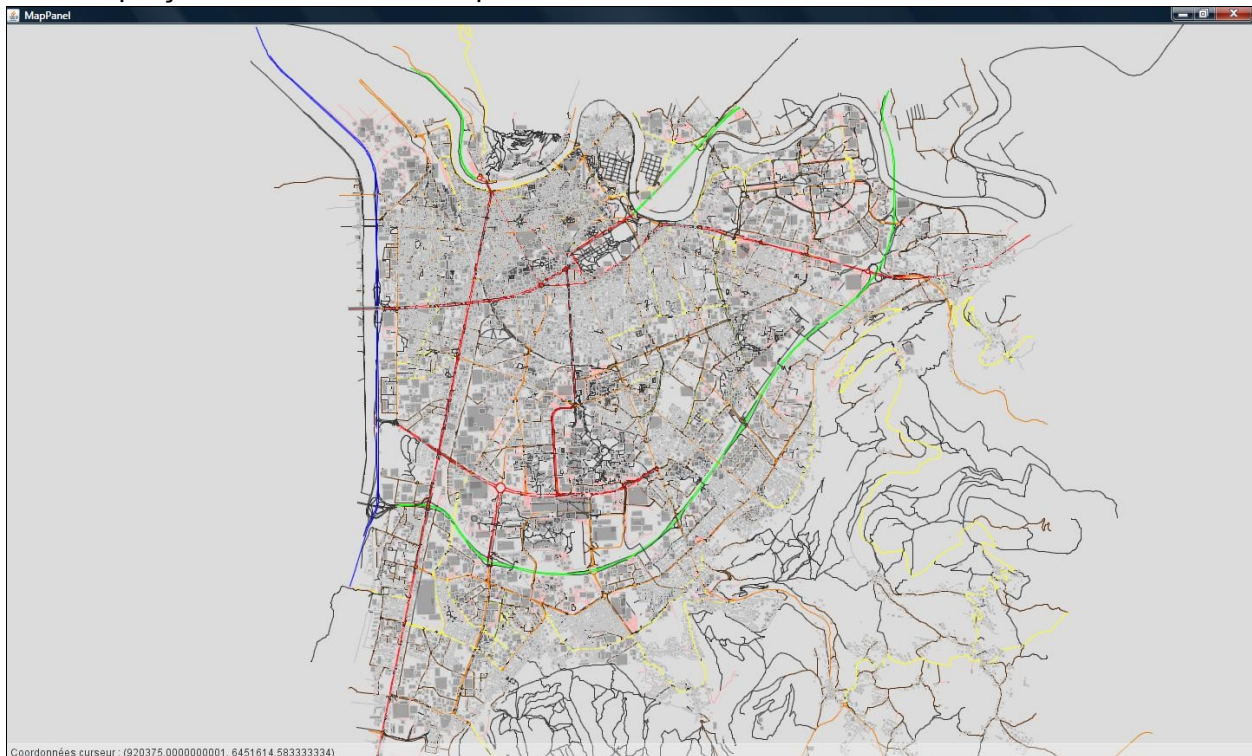
Motorway	Bleu
Trunk	Vert
Primary	Red
Secondary	Orange
Tertiary	Jaune
Unclassified	Marron
Residential	Gris clair
Service	Pink

**(alternative b)** : L'ensemble des bâtiments autour de Grenoble (longitudes comprises entre 5.7 et 5.8, et latitudes comprises entre 45.1 et 45.2).

Requête utilisée pour cette question :

```
SELECT ST_Transform(bbox, 2154) AS bbox FROM ways WHERE tags->'building' LIKE '%'
AND ST_Intersects(linestring, ST_SetSRID(ST_GeomFromText('Polygon((5.8 45.1, 5.8 45.2, 5.7
45.2, 5.7 45.1, 5.8 45.1))'),4326));
```

Voici un aperçu du résultat de cette question :



### **Question 11 (suggestion a) :**

Même question que la question 7 (nombre de boulangeries par quartier de Grenoble), mais en représentant graphiquement sur la carte les valeurs (utilisez des attributs graphiques tels que la couleur, ou autre).

Pour tester cette question, lancez l'application avec comme argument le nombre 11.

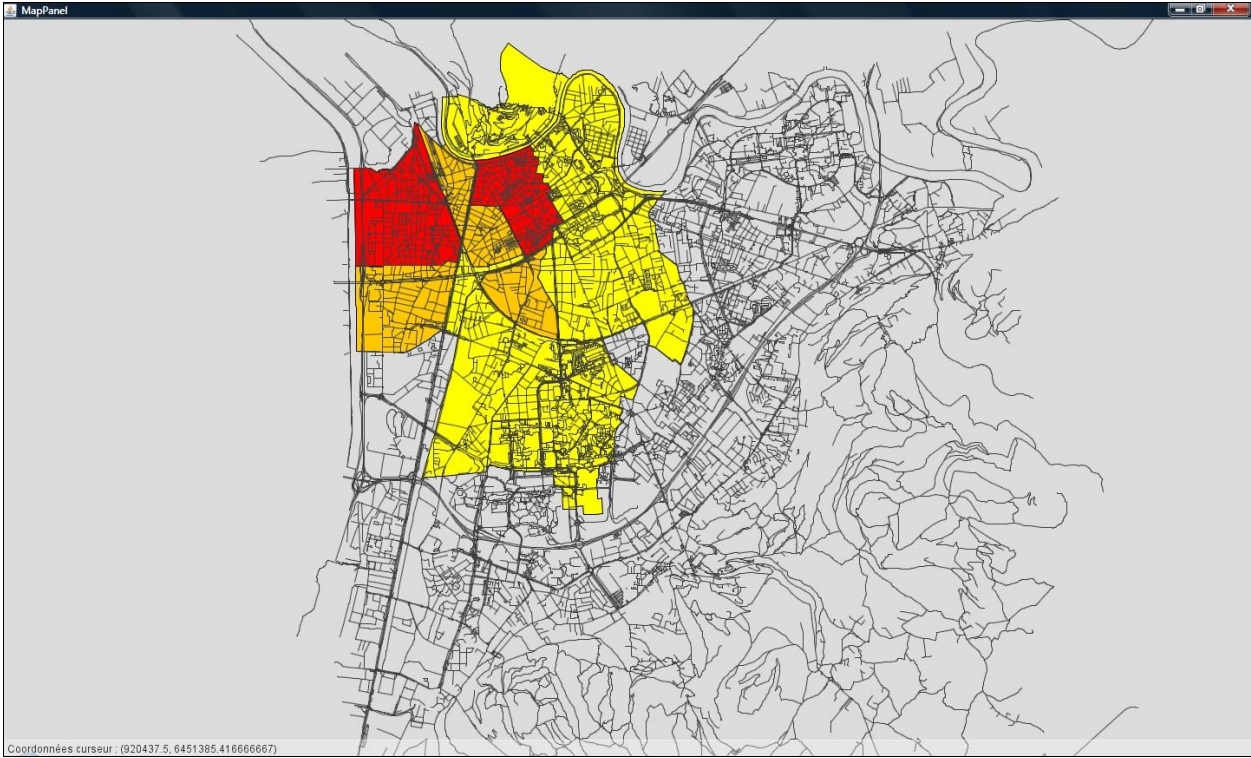
Requête utilisée pour cette question :

```
SELECT q.quartier AS quartier, q.the_geom AS contour, COUNT(n) AS nb_boulangeries FROM
quartier q, nodes n WHERE n.tags->'shop'='bakery' AND
ST_Intersects(ST_Transform(q.the_geom, 4326), n.geom) GROUP BY q.quartier, q.the_geom
ORDER BY COUNT(n) DESC;
```

Pour afficher les différentes valeurs, nous avons décidé de colorer en jaune les quartiers contenant entre 1 et 4 boulangeries, en orange les quartiers contenant entre 5 et 9 boulangeries et en rouge ceux qui contiennent au moins 10 boulangeries.

De plus, pour faciliter la localisation des différents quartiers, nous avons affiché les routes aux alentours de Grenoble.

Voici un aperçu du résultat de cette question :



### **III. Manuel Utilisateur**

Pour tester notre application, veuillez passer en argument le numéro de la question que vous souhaitez tester :

- 8 => programme simple pour tester la connection à la base de données.
- 9 \$name => affiche tous les noms et coordonnées géographiques des points dont le nom ressemble à \$name.
- 10 => affiche toutes les routes et tous les bâtiments autour de Grenoble.
- 11 => affiche le nombre de boulangeries par quartier à Grenoble.