Activité : les Bases de données relationnelles - partie 2

Présentation

Dans cette activité je vais **interroger** les données sur les villes du site http://sql.sh en utilisant le langage de requêtes SQL :

- à partir de site Web http://icn.boonum.fr, je vais pouvoir exécuter des requêtes SQL pour faire des recherches dans les données de la table **icn_ville**.

Niveau 1

Teste la requête suivante

```
SELECT dep, nom
FROM icn_ville
```

Question 1 : Ecrire la requête SQL pour avoir le **nom**, la **latitude** et la **longitude** (dans cet ordre) des villes.

Résultat à obtenir :

nom	latitude	longitude
Ozan	46.3833	4.91667
Cormoranche-sur-Saône	46.233	4.83333

...

```
#Note ta requête SQL ici
#
#
#
#
```

Niveau 2

Teste la requête suivante

```
SELECT nom AS ville, cp AS "code postal"
FROM icn_ville
```

Question 2 : Ecrire la requête SQL pour avoir le **nom**, le nombre d'habitants en 2010 des villes.

Résultat à obtenir :

nom	Nombre d'habitants en 2010
Ozan	618
Cormoranche-sur-Saône	1058

...

```
#Note ta requête SQL ici
#
#
#
#
```

Question 3 : Ecrire la requête SQL pour avoir le **nom**, la superficie de la commune des villes.

Résultat à obtenir :

nom de la commune	Superficie de la commune

...

```
#Note ta requête SQL ici
#
#
#
#
```

Niveau 3

Teste la requête suivante

```
SELECT nom AS commune, dens AS "Habitants par Km2"
FROM icn_ville
WHERE dep="87"
```

Question 4 : Ecrire la requête SQL pour avoir le **nom** et l'altitude minimum des villes de la Corrèze (département 19).

Résultat à obtenir :

Commune	Altitude minimum

...

```
#Note ta requête SQL ici
#
#
#
#
```

Question 5 : Ecrire la requête SQL pour avoir le **nom** des villes ayant **la même altitude minimum** que limoges.

Résultat à obtenir :

- **Cherche l'altitude minimum de Limoges**, *note la requête SQL permettant d'obtenir ce résultat et la valeur obtenue :*

```
#Requête SQL :
#
#
#
# valeur obtenue :
```

- Affiche ensuite le résultat comme suit :

nom de la commune	Altitude minimum

...

```
#Note ta requête SQL ici
#
#
#
#
```

Niveau 4

Teste la requête suivante

```
SELECT nom AS Commune, nb_hab_1999 AS "Nombre d'habitants en 1999"
FROM icn_ville
WHERE nb_hab_1999 > 100000
```

Question 6 : Ecrire la requête SQL pour connaitre les **communes** dont le nombre d'habitants en 2010 est inférieur à 300 .

Résultat à obtenir :

Commune	Nombre d'habitants en 2010

...

```
#Note ta requête SQL ici
#
#
#
#
```

Question 7 : Ecrire la requête SQL pour avoir le nom des **villes** dont la densité est supérieure à la densité de Paris.

(densité = Nombre d'habitants au km2) Résultat à obtenir :

Ville	Nombre d'habitants au km2

...

```
#Note ta requête SQL ici
#
#
#
#
```

Niveau 5

Teste la requête suivante

```
SELECT nom AS Commune, cp AS "Code postal", nb_hab_2010 AS "Nombre d'habitants en 2010"
FROM icn_ville
WHERE dep = "23" AND nb_hab_2010 > 2000
ORDER BY nb_hab_2010
```

Question 8 : Ecrire la requête SQL pour connaitre les **communes** dont le nombre d'habitants en 2010 est supérieur à 10000 pour les villes ayant une latitude négative (<0). Le résultat sera trié sur la latitude.

Résultat à obtenir :

Département	Commune	Nombre d'habitants en 2010	Latitude

...

```
#Note ta requête SQL ici
#
#
#
```

```
#
```

Question 9 : Ecrire la requête SQL pour afficher la liste des communes dont l'altitude est comprise entre 1500 et 2500 mètres. Le résultat sera trié sur l'altitude minimum.

Résultat à obtenir :

Département	Ville	Altitude minimum	Altitude maximum	Nombre d'habitants au km2

...

```
#Note ta requête SQL ici
#
#
#
#
```

Niveau 6

Teste la requête suivante

```
SELECT nom AS Commune, cp AS "Code postal", nb_hab_2010 AS "Nombre d'habitants en 2010"
FROM icn_ville
WHERE dep = "23" AND nb_hab_2010 > 2000
ORDER BY nb_hab_2010
```

Question 10 : Ecrire la requête SQL pour connaitre les **communes** dont le nombre d'habitants en 2010 est supérieur à 10000 pour les villes ayant une latitude négative (<0). Le résultat sera trié sur la latitude.

Résultat à obtenir :

Département	Commune	Nombre d'habitants en 2010	Latitude

...

```
#Note ta requête SQL ici
#
#
#
#
```

Question 11 : Ecrire la requête SQL pour afficher la liste des communes dont l'altitude est comprise entre 1500 et 2500 mètres. Le résultat sera trié sur l'altitude minimum.

Résultat à obtenir :

Département	Ville	Altitude minimum	Altitude maximum	Nombre d'habitants au km2

...

```
#Note ta requête SQL ici
#
#
#
#
```

Question 12 : Ecrire la requête SQL donnant la liste des villes situées entre la longitude 2.34 et 2.35. Le résultat sera trié sur le nom de la ville.

Résultat à obtenir :

Département	Ville	Latitude	Longitude

...

```
#Note ta requête SQL ici
#
#
#
#
```

Question 13 : Ecrire la requête SQL donnant la liste des villes dont la densité est supérieure à 2800 habitants au km2 et dont la surface est inférieure à 10km2. Le résultat sera trié selon la surface.

Résultat à obtenir :

Département	Commune	Nombre d'habitant au km2	Surface en km2	Nombre d'habitants en 2010

...

```
#Note ta requête SQL ici
#
#
#
#
```

Niveau 7

Teste la requête suivante

```
SELECT nom AS Commune, dep AS "Département", latitude, alt_max AS "Altitude maximum"
FROM icn_ville
WHERE nom LIKE "mont%" AND alt_max < 100
ORDER BY dep
```

Question 14 : Ecrire la requête SQL donnant la liste des villes des départements d'outremer. Le résultat sera trié sur le département.

Résultat à obtenir :

Département	Ville	Latitude	Longitude

...

```
#Note ta requête SQL ici
#
#
#
#
```

From:
/ - **Les cours du BTS SIO**

Permanent link:
/doku.php/icn/donneesbdd_th

Last update: **2019/04/01 15:30**