

Requêtes SQL- Travail avec la base de données *mondialNP*

Vous avez à votre disposition la base de données *mondialNP*, qui est composée de 3 tables :

Schéma de la table:

country (Name,Code,Capital,Province,Area,Population)

city (Name,Country,Province,Population,Longitude, Latitude)

borders (Country1,Country2,Length)

La clé primaire de chaque table est soulignée.

Dans la table **city**, l'attribut **Country** est une clé étrangère faisant référence à la clé primaire **Code** de la table **country**.

Dans la table **borders**, les attributs **Country1** et **Country2** sont deux clés étrangères faisant référence à la clé primaire **Code** de la table **country**.

Dans ce TP, vous allez écrire des requêtes en SQL suivantes et les tester au fur et à mesure.

Commencez toujours par vous poser la question suivante :

De quelle(s) table(s) ai-je besoin ?

Vous commencerez ainsi par remplir la partie FROM

I) Requêtes sans jointure (i.e. portant sur une seule table)

Travail dans la table city.

1) Faire afficher toutes les informations contenues dans cette table.

SELECT * FROM city

2) Combien de villes sont répertoriées dans cette table ?

SELECT COUNT(*) FROM city

Réponse: 3 111

3) Faire afficher sans doublons (i.e. on ne répète pas des villes qui auraient le même nom) le nom de toutes les villes répertoriées dans cette table.

SELECT DISTINCT Name FROM city

4) Ordonner cette liste par ordre alphabétique.

SELECT DISTINCT Name FROM city ORDER BY Name

5) Combien y a-t-il de noms de villes distincts ?

SELECT COUNT(DISTINCT Name) FROM city ORDER BY Name

Réponse: 3 044

Pour les questions suivantes, on suppose que l'on sait que le code utilisé pour la France est 'F'.

6) Faire afficher le nom de toutes les villes françaises répertoriées dans cette table.

`SELECT Name FROM city WHERE Country='F'`

7) Faire afficher le nom de toutes les villes françaises répertoriées dans cette table et leur population, par ordre décroissant de population.

`SELECT Name,Population FROM city WHERE Country='F' ORDER BY population DESC`

8) Combien de villes françaises sont répertoriées dans cette table ?

`SELECT COUNT(*) FROM city WHERE Country='F'`

Réponse: 39

9) Quelle est la population totale des villes françaises répertoriées dans cette table ?

`SELECT SUM(Population) FROM city WHERE Country='F'`

Réponse: 8 833 289

10) Quelle est la ville la plus peuplée répertoriée dans cette base ?

`SELECT Max(Population),Name FROM city`

Réponse: Seoul

11) Donner le nom de toutes les villes dont le nom commence par 'C' et finit par 'd'.

`SELECT Name FROM city WHERE Name LIKE 'C%d'`

12) Faire afficher le nom de toutes les villes françaises répertoriées dans cette table, qui sont situées au sud du 45ème parallèle.

`SELECT Name FROM city WHERE Country='F' AND Latitude<=45`

13) Faire afficher la population totale des villes regroupées par pays, en faisant afficher le code du pays à côté du total de la population.

`SELECT SUM(Population) AS Total , Country FROM city GROUP BY Country`

14) Ordonner cette liste par population totale décroissante.

`SELECT SUM(Population) AS Total , Country FROM city GROUP BY Country ORDER BY Total DESC`

15) Faire afficher la même chose qu'en question a), mais on ne veut que deux lignes : celle de la France et celle de la Belgique (code 'B')

`SELECT SUM(Population) AS Total , Country FROM city WHERE Country='F' OR Country='B' GROUP BY Country`

16) Faire afficher la même chose qu'en question a), mais vous ne gardez que les pays où la population totale des villes est supérieure à 10 millions.

`SELECT SUM(Population) AS Total , Country FROM city GROUP BY Country HAVING Total >=10000000 ORDER BY Total DESC`

17)(plus difficile, possibilité d'utiliser une sous-requête) Combien de pays sont affichés lors de cette dernière requête (pas de comptage à la main bien sûr:)

`SELECT COUNT(*) FROM (SELECT SUM(Population) AS Total , Country FROM city GROUP BY Country HAVING Total >=10000000)`

Travail dans la table borders.

Remarque sur la table borders: chaque frontière n'apparaît qu'une seule fois dans la table, mais par exemple quand on cherche la frontière France/Italie, on ne sait pas si la France sera Country1 ou Country2...

18) Faire afficher toutes les lignes correspondant aux frontières de la Suisse (code 'CH').

`SELECT * FROM borders WHERE Country1='CH' OR Country2='CH'`

19) Ecrire une requête permettant de compter combien la Suisse a de frontières.

```
SELECT COUNT(*) FROM borders WHERE Country1='CH' OR Country2='CH'
```

Réponse: 5

II) Requêtes avec jointures (i.e. portant sur deux tables ou plus)

20) Faire afficher le nom du pays où se trouve la ville qui s'appelle Ulm.

```
SELECT P.Name FROM city AS V JOIN country AS P ON V.Country=P.Code WHERE V.Name='Ulm'
```

Réponse: Germany

21) Dans quel(s) pays se trouve la ville de Cordoba ?

```
SELECT P.Name FROM country P JOIN city V ON V.Country=P.code WHERE V.Name='Cordoba'
```

Réponse: 3 pays: Spain, Mexico, Argentina

22) Faire afficher le nom de toutes les villes de Chine répertoriées dans cette base (on ne connaît pas le code utilisé pour la Chine, mais on sait qu'en anglais Chine se dit 'China'...)

```
SELECT V.Name , P.Name FROM country AS P JOIN city AS V ON P.Code=V.Country WHERE P.Name='China'
```

23) Faire afficher la population totale des villes regroupées par pays, en faisant afficher le nom du pays à côté du total de la population des villes. Vous ordonnerez cette liste par ordre décroissant de population.

```
SELECT P.Name, SUM(V.Population) AS TotalPopVille FROM city AS V Join country AS P ON V.Country=P.Code GROUP BY P.code ORDER BY TotalPopVille DESC
```

24) Donner la liste de tous les couples de noms de pays correspondant à une frontière avec la France ? (par exemple dans votre liste doit apparaître (France,Italy))

```
SELECT P1.Name, P2.Name FROM country P1 JOIN borders B ON P1.Code=B.Country1 JOIN country P2 ON P2.Code=B.Country2 WHERE P1.Name='France' OR P2.Name='France'
```

III) Requêtes diverses

25) Donner le nom des villes possédant au moins une ville homonyme (donner une liste qui si possible ne contient pas de doublons)

Les deux premières réponses utilisent une auto-jointure:

```
SELECT DISTINCT(V1.Name) FROM city V1 JOIN city V2 ON V1.Name=V2.Name AND (V1.Name,V1.Country,V1.Province)!=(V2.Name,V2.Country,V2.Province)
```

OU BIEN

```
SELECT DISTINCT(V1.Name) FROM city V1 CROSS JOIN city V2 WHERE (V1.Name=V2.Name AND (V1.Name,V1.Country,V1.Province)!=(V2.Name,V2.Country,V2.Province))
```

OU BIEN (plus simple)

```
SELECT V.name FROM City V GROUP BY V.name HAVING COUNT(*)>1
```

26)(difficile) Quel est le nom du pays possédant le plus de frontières ?

```
SELECT P.Name, SUM(Voisins) AS Total_voisins FROM  
(SELECT Country1 AS Pays, COUNT(*) AS Voisins FROM borders GROUP BY Country1  
UNION ALL  
SELECT Country2 AS Pays, COUNT(*) AS Voisins FROM borders GROUP BY Country2)  
AS matable  
JOIN country P ON P.code=matable.Pays  
GROUP BY Pays  
ORDER BY Total_voisins DESC LIMIT 1
```

Réponse: la Chine a 16 voisins