

Proposition de corrigé du TP5 : Bases de données

1 Présentation de la base de données

1. Afficher toutes les entrées de la table `country` puis celle de `city` (une requête pour chaque, l'une après l'autre).

```
SELECT * FROM country
SELECT * FROM city
```

2. Afficher toutes les entrées de la table `city` qui concerne le pays France dont le code est F.

```
SELECT * FROM city WHERE country='F'
```

3. Afficher toutes les entrées de la table `city` qui concerne le pays France et dont la population est d'au moins 150 milliers d'habitants. Le faire également avec une intersection de table (syntaxe : `SELECT * FROM INTERSECT SELECT ...`).

```
SELECT * FROM city WHERE country='F' AND population >= 150000

(SELECT * FROM city WHERE country='F')
INTERSECT
(SELECT * FROM city WHERE population >= 150000)
```

4. Afficher toutes les entrées de la table `city` qui concerne le pays France et dont la population est d'au moins 150 milliers d'habitants, classées par ordre alphabétique du nom de la ville.

```
SELECT * FROM city WHERE country='F' AND population >= 150000 ORDER BY name
```

5. Afficher toutes les entrées de la table `city` qui concerne le pays France et dont la population est d'au moins 150 milliers d'habitants, classées par nombre d'habitants dans l'ordre décroissant. (Rappel : ASC signifie ordre ascendant, et DESC ordre descendant).

```
SELECT * FROM city WHERE country='F' AND population >= 150000 ORDER BY population DESC
```

2 Agrégation

On rappelle les principales fonctions d'agrégation : COUNT, MAX, MIN, SUM, AVG qui donnent respectivement le nombre, le maximum, le minimum, la somme et la moyenne.

6. Donner le nombre d'entrées de la table `city`, de même que pour la table `country` (une requête à chaque fois).

```
SELECT COUNT(*) FROM city
SELECT COUNT(*) FROM country
```

7. Quelle est la moyenne du nombre d'habitants par ville dans le pays France (code F) ?

```
SELECT AVG(population) FROM city WHERE country='F'
```

8. Donner le nombre d'habitants du pays France vivant dans les villes enregistrées dans la base de données `city`. On affichera le résultat en milliers d'habitants.

```
SELECT SUM(population)/1000 FROM city WHERE country='F'
```

9. Faire une requête donnant simultanément l'altitude maximale et l'altitude minimale des villes mondiales.

```
(SELECT min(elevation) FROM city) UNION (SELECT max(elevation) FROM city)
```

10. La requête précédente a montré (normalement) que l'altitude d'une ville est comprise entre -28 et 4330 mètres. Quelle est le nombre d'altitudes atteintes dans l'intervalle $[-28, 4330]$? (Réponse : 929)

```
SELECT COUNT(elevation) FROM (SELECT DISTINCT elevation FROM city)
```

11. Donner pour chaque pays le nombre d'habitants (c'est la somme du nombre d'habitants de chaque ville : on se contentera d'interroger la table `city` et on donnera simplement le code du pays et le nombre total d'habitants).

```
SELECT SUM(population), country FROM city GROUP BY country
```

12. Quelle est la liste des pays (on se contentera là aussi d'interroger la table `city` et de donner le code du pays) ayant au moins 500000 habitants ? Afficher le résultat par nombre d'habitants décroissant (vous devez trouver 144 résultats).

```
SELECT SUM(population), country FROM city
GROUP BY country HAVING SUM(population) >= 500000
```

3 Jointures

On va maintenant s'intéresser à des résultats concernant les deux tables `city` et `country`. On rappelle que pour faire une jointure, on utilise `JOIN nom_autre_table ON attribut`. On précise le nom des attributs à garder dans une sélection par `nom_table.nom_attribut` (nécessaire si le nom de l'attribut figure dans les deux tables, facultatif sinon). Il est courant et pratique de procéder à un renommage de la table (avec `AS` ou simplement en juxtaposant le nouveau nom à l'ancien) pour écourter les requêtes.

13. Écrire une requête permettant d'obtenir la capitale la moins peuplée du monde (réponse : Plymouth, Montserrat) et une autre pour la plus peuplée (réponse : Lilongwe, Malawi).

```
SELECT name, country, pop name
FROM (SELECT city.name AS name, country.name AS country, city.population AS pop
FROM city JOIN country ON city.country=country.code AND city.name=country.capital)
WHERE pop=(SELECT MAX(pop) FROM
(SELECT city.name AS name, country.name AS country, city.population AS pop
FROM city JOIN country ON city.country=country.code AND city.name=country.capital))
```

14. Quelle est la population moyenne des villes nord-américaines (États-Unis et Canada réunis) ? (réponse : environ 340 174)

```
SELECT AVG(population)
FROM (SELECT city.population FROM city JOIN country ON country.code=city.country
WHERE country.name='United_States' OR country.name='Canada')
```

15. En considérant que la population d'un pays est égale à la somme des populations des villes qu'il abrite, donner la liste des pays par population décroissante. Quel est le pays le plus peuplé ? (Réponse : la Chine)

```
SELECT SUM(city.population), country.name
FROM city JOIN country ON city.country=country.code
GROUP BY country.name HAVING SUM(city.population) IS NOT null
ORDER BY SUM(city.population) DESC
```

4 Sous-requêtes et requêtes complexes

16. Quel est le nom de la (ou les) ville(s) ayant l'altitude maximale ? (Réponse : 'Cerro de Pasco', 4330)

```
SELECT name, elevation FROM city WHERE elevation= (SELECT MAX(elevation) FROM city)
```

17. Quelle est le nombre de villes situées au dessus de l'altitude moyenne des villes mondiales ? On n'écrira qu'une seule requête. (Réponse : 786).

```
SELECT COUNT(*) FROM city WHERE elevation >= (SELECT AVG(elevation) FROM city)
```

18. La fonction `ROUND()` permet d'arrondir à l'entier le plus proche. Quelle est le nombre de villes situées à l'altitude moyenne des villes mondiales, arrondie à l'entier le plus proche ?

```
SELECT COUNT(*) elevation FROM city
WHERE elevation = (SELECT ROUND(AVG(elevation)) FROM city)
```

19. En considérant que la population totale d'un pays est égale à la somme des populations des villes qu'il abrite, calculer la population moyenne d'un pays. (Réponse : environ 8262882)

```
SELECT AVG(pop) FROM (SELECT SUM(city.population) AS pop FROM city GROUP BY country)
```

20. Quel est la capitale du pays le moins peuplé ?

(Réponse : Adamstown, capitale de Pitcairn.)

```
SELECT capital, name, population FROM country  
WHERE population= (SELECT MIN(population) FROM country)
```