

NOM : ..... PRENOM : ..... n° ETU : .....

La base de données utilisée contient 2 tables, `city` et `country`.

country			city		
Column	Type		Column	Type	
countrycode	character(3)		id	integer	
name_country	text		name_city	text	
continent	text		countrycode	character(3)	
region	text		district	text	
population_country	integer		population_city	integer	
lifeexpectancy	real				
capital	integer				
governmentform	text				
gnp	integer				
surface	integer				

La base de données satisfait également les contraintes de clefs étrangères suivantes :

- `city.countrycode`  $\subseteq$  `country.countrycode`,
- `country.capital`  $\subseteq$  `city.id`.

1. Écrire en SQL une requête dont le résultat contient pour chaque pays la ville la plus peuplée de ce pays, par ordre alphabétique sur `countrycode` (tableau résultat : `name_city`, `population_city`).

```
SELECT C1.name_city, C1.population_city
FROM City C1
WHERE C1.population_city >= ALL
      (SELECT C2.population_city
       FROM City C2
       WHERE C1.countrycode=C2.countrycode)
ORDER BY C1.countrycode;
```

ou

```
SELECT C1.name_city, C1.population_city
FROM City C1
WHERE C1.population_city =
      (SELECT max(C2.population_city)
       FROM City C2
       WHERE C1.countrycode=C2.countrycode)
ORDER BY C1.countrycode;
```

ou

```

SELECT C1.name_city, C1.population_city
FROM City C1
WHERE NOT EXISTS
    (SELECT *
     FROM City C2
     WHERE C1.countrycode=C2.countrycode
     AND C1.population_city < C2.population_city)
ORDER BY C1.countrycode

```

2. Écrire en SQL une requête dont le résultat contient le nom des pays dont la surface (qu'on suppose par exemple donnée en nombre de kilomètres carrés) est strictement supérieure à la moitié du nombre d'habitants (tableau résultat : `name_country`)

```

SELECT name_country
FROM Country
WHERE surface > 0,5 * population_country ;

```

3. Écrire en SQL une requête dont le résultat contient pour chaque continent, la région dont la superficie est la plus grande (tableau résultat : `continent, region, total_surface`).

```

WITH size_region as (SELECT region, sum(surface) as total_surface
                     FROM country
                     GROUP BY region)
SELECT continent, region, total_surface
FROM (country NATURAL JOIN size_region) as J
WHERE J.total_surface =
    (Select max(S.total_surface)
     FROM size_region S
     WHERE S.continent=J.continent);

```

ou

```

SELECT C1.continent, C1.region, sum(C1.surface) as total_surface
FROM country C1
GROUP BY C1.continent, C1.region
HAVING sum(C1.surface) >= ALL
    (Select sum(C2.surface)
     FROM Country C2
     WHERE C1.continent=C2.continent
     GROUP BY C2.region);

```

4. Écrire en SQL une requête dont le résultat contient les régions dans lesquelles tous les pays ont la même espérance de vie. Donnez deux versions : une utilisant l'agrégation et l'autre pas.

```

SELECT C1.region
FROM Country C1
WHERE NOT EXISTS (SELECT *
                  FROM COUNTRY C2
                  WHERE C1.region=C2.region
                  AND C1.lifeexpectancy<>C2.lifeexpectancy);

```

ou

```

SELECT C1.region
FROM Country C1
WHERE C1.lifeexpectancy = ALL (SELECT C2.lifeexpectancy
                              FROM COUNTRY C2
                              WHERE C1.region=C2.region);

```

ou (en utilisant l'agrégation)

```
SELECT region
FROM Country
GROUP BY region
HAVING COUNT(DISTINCT lifeexpectancy)=1;
```