

CC1

Nom des pays dont l'espérance de vie est supérieure à 70 ans

```
SELECT nom_pays FROM country
WHERE lifeexpectancy>70;
```

$\Pi_{nom_pays}(\sigma_{lifeexpectancy>70}(country))$

Nom des capitales des pays du continent 'Europe' ^[1]

```
SELECT name_city FROM city
INNER JOIN country ON
city.id=country.capital
AND country.continent='Europe';
```

$\Pi_{name_city}(city_{city.id=country.capital \wedge country.continent='Europe'})$

Régions dans lesquelles tous les pays ont une espérance de vie supérieure à 80 ans

```
SELECT region FROM country AS c1
WHERE NOT IN(
SELECT DISTINCT region FROM country AS c2 WHERE
c2.lifeexpectancy<=80
);
```

Nom des pays dans lesquels il existe deux villes différentes ayant exactement le même nombre d'habitants

```
SELECT countrynome FROM country
INNER JOIN city AS c1 ON
c1.countrycode=country.countrycode
INNER JOIN city AS c2 ON
c2.countrycode=c1.countrycode
AND c2.population_city=c1.population_city
AND NOT c2.id=c1.id;
```

Nom des pays dont le nombre d'habitants est supérieur à 20 millions ^[2]

```
SELECT name_country FROM country
WHERE population_country>200000000;
```

$\Pi_{name_country}(\sigma_{population_country>200000000}(country))$

Nom des capitales des pays dont l'espérance de vie est inférieure à 60 ans

```
SELECT name_city FROM city
INNER JOIN country ON
city.id=country.capital
AND country.lifeexpectancy<60;
```

$\Pi_{name_city}(city_{city.id=country.capital \wedge lifeexpectancy<60})$

Districts de France dans lesquels toutes les villes ont une population supérieure à 500 habitants

```
SELECT district FROM city AS c1
WHERE NOT IN(
SELECT DISTINCT district FROM city AS c2
INNER JOIN country ON
c2.population_city<=500
AND name_country='France'
AND country.countrycode=c2.countrycode
);
```

Nom des pays dans lesquels la capitale est la ville la plus peuplée

```
SELECT name_country FROM country
WHERE NOT EXISTS(
SELECT * FROM city AS c1
WHERE c1.id=capital
INNER JOIN city AS c2 ON
c2.population_city>c1.population_city
AND NOT c2.id=c1.id
);
```

Nom des capitales des pays du continent 'Asia' avec moins de 10 millions d'habitants ^[3]

```
SELECT name_city FROM city
INNER JOIN country ON
continent='Asia'
AND population_city<100000000;
AND id=capital
```

$\Pi_{name_city}(city_{continent='Asia' \wedge population_city>10000000 \wedge id=capital})$

Régions dans lesquelles aucun pays n'a un nom qui commence par la lettre 'P'

```
SELECT region FROM country AS c1
WHERE NOT EXISTS(
  SELECT DISTINCT region FROM country AS c2
  WHERE LEFT(c2.name_country, 1)='P'
);
```

Régions dans lesquels tous les pays ont une capitale avec plus de 100 000 habitants

```
SELECT region FROM country AS c1
WHERE NOT IN(
  SELECT DISTINCT region FROM country AS c2
  INNER JOIN city ON
    id=capital
    AND population_city<=100000
);
```

Nom des villes qui sont capitale d'un pays et aussi ville d'un autre pays sur le même continent [4]

```
SELECT name_city FROM city AS cit1
INNER JOIN city AS cit2 ON
  cit2.name_city=cit1.name_city
AND NOT cit2.countrycode=cit1.countrycode
INNER JOIN country AS cou1 ON
  cit1.id=cou1.capital
INNER JOIN country AS cou2 ON
  cit2.countrycode=cou2.countrycode
AND cou2.continent=cou1.continent;
```

Quelle(s) requête(s) calcule(nt) le nom des villes françaises de plus de 200 000 habitants ? [5]

```
SELECT name_city FROM city WHERE
  population_city>200000
AND countrycode='FRA';
```

⇒ OK

```
SELECT A.name_city FROM city AS A, country AS B WHERE
  population_city>200000
AND B.countrycode='FRA';
```

⇒ On va avoir des problèmes à cause du produit cartésien : certaines lignes pourront avoir B.countrycode='FRA' et A.countrycode='ENG'

- Dans la requête, pas besoin de country.countrycode=city.countrycode car id est une clé primaire.
- La méthode adoptée pour la question demandant « le nom des pays dont la population totale est au moins 50 millions » pourrait s'assimiler à du plagiat vis-à-vis de cette question.
- Syntaxe digne d'un collégien... On supposera ici que ce sont les villes qui ont moins de 10 millions d'habitants. L'adaptation de la requête pour l'autre sens possible de la question est néanmoins trivial.
- Une fois de plus, syntaxe vomitive. On supposera qu'on recherche les noms tels qu'ils puissent faire référence à la fois à la capitale d'un pays P1 et à une ville d'un pays P2 (avec P1 ≠ P2 et P1.continent=P2.continent)
- On supposera qu'on a countrycode='FRA' ⇒ name_country='France'