**Exo3**

SELECT libelle\_region FROM t\_pays INNER JOIN t\_regions ON (t\_pays.region\_id=t\_regions.id\_region) WHERE libelle\_region LIKE 'Europe%' GROUP BY libelle\_region;

* Sélectionne le nom des régions en partant de pays et en liant les régions
* Pour toutes les régions qui ont pour continent Europe, afficher la liste

SELECTIONNE/AFFICHE « nom region » A PARTIR DE « table pays » EN LIANT « table région » SUR (« table pays » . « id\_region » = « table region » . « id region » ALORS QUE « nom région » RESSEMBLE A ‘Europe%’ REGROUPE PAR « nom region »

SELECT libelle\_pays FROM t\_pays INNER JOIN t\_continents ON (t\_pays.continent\_id=t\_continents.id\_continent) WHERE libelle\_continent LIKE 'Europe%';

* Sélectionne le nom des pays en partant de pays et en liant les continents
* Pour tous les pays dans Europe, afficher la liste

SELECT libelle\_pays FROM t\_pays INNER JOIN t\_regions ON (t\_pays.region\_id=t\_regions.id\_region) WHERE libelle\_region LIKE '%Afrique Centrale%' GROUP BY libelle\_pays;

* Sélectionne le nom des pays en partant de pays et en liant les régions
* Pour tous les pays dans la région Afrique Centrale, afficher la liste

SELECT libelle\_continent,COUNT(libelle\_pays) FROM t\_pays INNER JOIN t\_continents ON (t\_pays.continent\_id=t\_continents.id\_continent) WHERE libelle\_continent LIKE 'Oceanie%' GROUP BY libelle\_continent;

* Sélectionne le nom des Continents, compte le nombre de pays en liant aux continents
* Affiche le nombre de pays du continent Océanie

SELECT libelle\_continent,COUNT(libelle\_region) FROM t\_regions INNER JOIN t\_continents ON (t\_regions.continent\_id=t\_continents.id\_continent) WHERE libelle\_continent LIKE '%Asie%';

* Sélectionne le nom des Continents, compte le nombre de régions en liant aux continents
* Affiche le nombre de régions du continent Asie

SELECT libelle\_continent,COUNT(libelle\_region) FROM t\_regions INNER JOIN t\_continents ON (t\_regions.continent\_id=t\_continents.id\_continent) WHERE id\_continent=3;

* Sélectionne le nom des Continents, compte le nombre de régions en liant aux continents
* Affiche le nombre de région du continent Amérique Septentrionale (id\_3) = NULL/0

**Exo4**

SELECT libelle\_continent, SUM(population\_pays) FROM t\_pays INNER JOIN t\_continents ON (t\_pays.continent\_id=t\_continents.id\_continent) GROUP BY libelle\_continent;

* Sélectionne le nom des Continents, fait la somme de la population totale
* Affiche le total de population par Continent

SELECT libelle\_region, SUM(population\_pays) FROM t\_pays INNER JOIN t\_regions ON (t\_pays.region\_id=t\_regions.id\_region) GROUP BY libelle\_region;

* Sélectionne le nom des Régions, fait la somme de la population totale
* Affiche le total de population par Région

SELECT libelle\_continent AS 'Continent', SUM(population\_pays) AS 'Population Totale' FROM t\_pays INNER JOIN t\_continents ON (t\_pays.continent\_id=t\_continents.id\_continent) GROUP BY libelle\_continent ORDER BY SUM(population\_pays) DESC LIMIT 1;

* Sélectionne le nom des Continents, fait la somme de la population totale
* Affiche Continent ayant la plus grande population

SELECT libelle\_region, SUM(population\_pays) FROM t\_pays INNER JOIN t\_regions ON (t\_pays.region\_id=t\_regions.id\_region) GROUP BY libelle\_region ORDER BY SUM(population\_pays) DESC LIMIT 1;

* Sélectionne le nom des Régions, fait la somme de la population totale
* Affiche Région ayant la plus grande population

SELECT libelle\_continent AS 'Continent', SUM(population\_pays) AS 'Population Totale' FROM t\_pays INNER JOIN t\_continents ON (t\_pays.continent\_id=t\_continents.id\_continent) GROUP BY libelle\_continent ORDER BY SUM(population\_pays) ASC LIMIT 1;

* Sélectionne le nom des Continents, fait la somme de la population totale
* Affiche Continent ayant la plus petite population

SELECT libelle\_region, SUM(population\_pays) FROM t\_pays INNER JOIN t\_regions ON (t\_pays.region\_id=t\_regions.id\_region) GROUP BY libelle\_region ORDER BY SUM(population\_pays) ASC LIMIT 1;

* Sélectionne le nom des Régions, fait la somme de la population totale
* Affiche Région ayant la plus petite population

SELECT libelle\_pays AS 'Pays', MAX(esperance\_vie\_pays) AS 'Espérance de vie' FROM t\_pays;

* Sélectionne le nom des Pays et affiche celui qui a la plus grande espérance de vie

Correction :

SELECT libelle\_pays, esperance\_vie\_pays FROM t\_pays WHERE esperance\_vie\_pays = (SELECT MAX(esperance\_vie\_pays) FROM t\_pays) ;

SELECT libelle\_pays AS 'Pays', MIN(taux\_mortalite\_pays) AS 'Taux de Mortalité' FROM t\_pays;

* Sélectionne le nom des Pays et affiche celui qui a le plus petit taux de mortalité

SELECT libelle\_pays AS 'Pays', MAX(taux\_natalite\_pays) AS 'Taux de Natalité' FROM t\_pays;

* Sélectionne le nom des Pays et affiche celui qui a le plus grand taux de natalité

SELECT libelle\_pays AS 'Pays', MAX(nombre\_enfants\_par\_femme\_pays) FROM t\_pays;

* Sélectionne le nom des Pays et affiche celui qui a le plus grand nombre de naissance par femme

SELECT libelle\_pays AS 'Pays', MAX(population\_plus\_65\_pays) FROM t\_pays;

* Sélectionne le nom des Pays et affiche celui qui a le plus grand taux de +65 ans

SELECT libelle\_pays, AVG(esperance\_vie\_pays) FROM t\_pays;

* Calcule la moyenne de l’espérance de vie de tous les pays

SELECT libelle\_region, AVG(taux\_natalite\_pays) FROM t\_pays INNER JOIN t\_regions ON (t\_pays.region\_id=t\_regions.id\_region) GROUP BY libelle\_region;

* Calcule la moyenne du taux de natalité pour chaque région

SELECT libelle\_continent, AVG(taux\_natalite\_pays) FROM t\_pays INNER JOIN t\_continents ON (t\_pays.continent\_id=t\_continents.id\_continent) GROUP BY libelle\_continent;

* Calcule la moyenne du taux de natalité pour chaque continent

SELECT libelle\_region, AVG(taux\_mortalite\_pays) FROM t\_pays INNER JOIN t\_regions ON (t\_pays.region\_id=t\_regions.id\_region) GROUP BY libelle\_region;

* Calcule la moyenne du taux de mortalité pour chaque région

SELECT libelle\_continent, AVG(taux\_mortalite\_pays) FROM t\_pays INNER JOIN t\_continents ON (t\_pays.continent\_id=t\_continents.id\_continent) GROUP BY libelle\_continent;

* Calcule la moyenne du taux de mortalité pour chaque continent

SELECT libelle\_pays FROM t\_pays WHERE libelle\_pays LIKE 'A%';

* Sélectionne tous les pays commençants par la lettre A

SELECT SUM(population\_pays) FROM t\_pays INNER JOIN t\_continents ON (t\_pays.continent\_id=t\_continents.id\_continent) WHERE libelle\_continent LIKE 'Amérique%';

* Calcule la somme de la population en liant pays et continents
* Affiche la somme pour les Amériques

SELECT SUM(population\_plus\_65\_pays) / SUM(population\_pays) \* 100 FROM t\_pays;

* Calcule le % des +65 mondial

SELECT SUM(population\_pays) FROM t\_pays WHERE libelle\_pays LIKE 'Corée%';

* Calcule la somme de la population en Corée (nord/sud)

SELECT SUM(population\_pays) FROM t\_pays WHERE libelle\_pays LIKE '%fran%';

* Calcule la somme de la population en France (Métro/DOM)

SELECT taux\_natalite\_pays FROM t\_pays WHERE taux\_natalite\_pays > 17;

* Affiche les pays où taux natalité > 17

SELECT population\_pays FROM t\_pays WHERE population\_pays > 20000;

* Affiche les pays où population > 20M

SELECT taux\_croissance\_pays FROM t\_pays WHERE taux\_croissance\_pays < 0;

* Affiche les pays où taux croissance démo. Négatif

SELECT SUM(population\_pays) FROM t\_pays WHERE libelle\_pays LIKE '%Chin%';

* Calcule la somme de la population en Chine (Chine + Macao + Hong-Kong)

SELECT (SUM(population\_pays) / (SELECT SUM(population\_pays) FROM t\_pays)) \* 100 AS pourcentage\_population\_chine FROM t\_pays WHERE libelle\_pays LIKE '%Chin%';

* Calcule le % de la Chine par rapport à la pop. Mondiale

SELECT

(SELECT SUM(population\_pays)

FROM t\_pays

INNER JOIN t\_continents ON t\_pays.continent\_id = t\_continents.id\_continent

WHERE libelle\_continent LIKE 'Europe%') +

(SELECT SUM(population\_pays)

FROM t\_pays

WHERE libelle\_pays LIKE 'Turquie%') AS total\_population;

* Additionner la pop totale Turquie + Europe

**Exo7**