

MP 25/26 – T.D. d'INFORMATIQUE n°3

Durant ce T.D. vous allez utiliser la base de données *mondial* que vous trouverez sur cahier de prépas. Voici une partie du schéma relationnel de la base de données mondial. Sont soulignés le ou les attributs constituant la clé primaire de chacune des tables.

Continent : Name, Area

Country : Name, Code, Capital, Province, Area, Population ; (Province est la région de la capitale)

City : Name, Country, Province, Population, Longitude, Latitude, Elevation ; (Country est le code du pays)

Encompasses : Country, Continent, Percentage ;

Borders : Country1, Country2, length ; (Country1 < Country2 pour l'ordre lexicographique)

Organization : Abbreviation, Name, City, Country, Province, Established ; (Established est la date de fondation)

IsMember : Country, Organization, type ;

Population : Country, Population_growth, infant_mortality ;

Economy : Country, GDP, Agriculture, Service, Industry, Inflation, Unemployment ; (GDP est le PIB)

Politics : Country, Independence, WasDependent, Dependent, Government ; (Indépendance est une date, Dependent l'ex-pays colonisateur)

Language : Name, Country, Percentage ;

EthnicGroup : Name, Country, Percentage ;

Religion : Name, Country, Percentage ;

Mountain : Name, Mountains, Elevation, Type, Longitude, Latitude ;

Geo_Mountain : Mountain, Country, Province ;

Sea : Name, Area, Depth ;

Geo_Sea : Sea, Country, Province ;

Lake : Name, River, Area, Elevation, Depth, Height, Type, River, Latitude, Longitude ;

Geo_Lake : Lake, Country, Province ;

River : Name, River, Lake, Sea, Length ; (la rivière se jette dans une rivière, un lac ou la mer)

Geo_River : River, Country, Province.

Rédiger une requête SQL pour obtenir :

(Requêtes de base)

- 1) la liste des continents et de leurs superficies, triés par superficie décroissante ;
- 2) la liste des capitales mondiales, triées par ordre alphabétique ;
- 3) le nom des dix pays ayant la plus petite superficie ;
- 4) les cinq villes les plus peuplées au monde.

(Jointures)

- 5) le nom des pays qui sont à cheval sur plusieurs continents ;
- 6) les pays du continent africain qui comptent moins de 10 habitants par km² ;
- 7) les fleuves de France (c'est-à-dire les rivières de France qui se jettent dans la mer) ;
- 8) les trois plus hautes montagnes africaines.

(Fonctions d'agrégation)

- 9) la liste ordonnée des cinq langues parlées dans le plus de pays différents ;
- 10) les langues parlées par moins de 500 000 personnes dans le monde ;
- 11) les fleuves se jetant dans la mer Méditerranée traversant plus d'un pays et le nombre de pays qu'ils traversent ;

(Sous-requêtes et opérations ensemblistes)

- 12) les noms des pays non membres de l'UNESCO ;
- 13) les pays frontaliers de la France, triés par population décroissante ;
- 14) le nombre de voisins de chaque pays d'Europe.

Réponses :

1) Asie... Océanie ; **2)** 244 réponses : Abu Dhabi... Zagreb ; **3)** Holy See... Nauru ; **4)** Shanghai... Mumbai ; **5)** Egypte, Indonésie, Kazakhstan, Russie, Turquie ; **6)** 8 pays : Gabon... Western Sahara ; **7)** 6 réponses : Garonne... Seine ; **8)** 3 réponses : kilimandjaro, ..., Mawenzi ; **9)** English, Spanish, ..., German ; **10)** 3 réponses : Futunian, ..., Tongan ; **11)** 7 réponses : Buna ... Vardar ; **12)** 36 pays : American Samoa... Western Sahara ; **13)** 8 pays : Germany... Monaco ; **14)** 46 réponses : Albany, 4 ... United Kingdom, 1.

- 1) `SELECT Name, Area FROM Continent ORDER BY Area DESC`
- 2) `SELECT Capital FROM Country ORDER BY Capital`
- 3) `SELECT Name FROM Country ORDER BY Area ASC LIMIT 10`
- 4) `SELECT Name, Population FROM City ORDER BY Population DESC LIMIT 5`
- 5) `SELECT DISTINCT C.name FROM Country C JOIN Encompasses E ON C.code = E.Country WHERE E.Percentage < 100`
- 6) `SELECT Country.name FROM Country JOIN Encompasses ON Country.code = Encompasses.Country WHERE country.population / Country.Area < 10 AND Encompasses.Continent = 'Africa'`
- 7) `SELECT Name FROM River R JOIN Geo_River GR ON R.Name = GR.River WHERE Country = 'F' AND Sea IS NOT NULL GROUP BY Name`
- 8) `SELECT M.Name FROM Mountain M JOIN Geo_Mountain GM JOIN Encompasses E ON GM.Country = E.Country AND M.Name = GM.Mountain WHERE E.Continent = 'Africa' ORDER BY M.Elevation DESC LIMIT 3`
- 9) `SELECT Name FROM Language GROUP BY Name ORDER BY COUNT(Country) DESC LIMIT 10`
- 10) `SELECT L.Name FROM Language L JOIN Country C ON L.country = C.code GROUP BY L.Name HAVING SUM(C.Population * L.percentage / 100) < 500000`
- 11) `SELECT R.Name, COUNT (DISTINCT GV.Country) AS nbre FROM River R JOIN geo_River GV ON R.name = GV.River WHERE R.Sea = 'Mediterranean Sea' GROUP BY R.Name HAVING nbre > 1`
- 12) `SELECT Name FROM Country EXCEPT SELECT C.Name FROM Country C JOIN Ismember I ON C.Code = I.Country WHERE I.Organization = 'UNESCO'`
- 13) `SELECT Name FROM (SELECT C.Name, C.population FROM Country C JOIN borders B ON C.Code = B.Country1 WHERE B.Country2 = 'F' UNION SELECT C.Name, C.population FROM Country C JOIN borders B ON C.Code = B.Country2 WHERE B.country1 = 'F') ORDER BY population DESC`
- 14) `SELECT n, SUM(nbv) FROM (SELECT C1.Name AS n, COUNT(C2.Name) AS nbv FROM Country C1 JOIN Country C2 JOIN Borders B JOIN Encompasses E ON C1.Code = B.Country1 AND C2.Code = B.Country2 AND E.Country = C1.Code WHERE E.Continent = 'Europe' GROUP BY C1.Name UNION SELECT C2.Name AS n, COUNT(C1.Name) AS nbv FROM Country C1 JOIN Country C2 JOIN Borders B JOIN Encompasses E ON C1.Code = B.Country1 AND C2.Code = B.Country2 AND E. Country = C2.Code WHERE E.Continent = 'Europe' GROUP BY C2.Name) GROUP BY n`