

- EXERCICE 1 -

La base de données utilisée dans cet exercice liste des informations sur des villes et pays du monde entier. Elle comprend trois tables :

- La table "country" contient les pays du monde, identifiés par un code à trois lettres ainsi que de multiples informations sur ces pays, la capitale de chaque pays étant un entier correspondant à l'identifiant de cette ville.
country = ((Code : TEXT), (Name : TEXT), (Continent : TEXT), (SurfaceArea : INT), (Population : INT), (LifeExp : INT), (GNP : INT), (Capital : INT))
- La table "city" recense quelques grandes villes mondiales. A chaque ville est attribué un identifiant unique, le code du pays auquel elles appartiennent et d'autres informations.
city = ((ID : INT), (Name : TEXT), (CountryCode : TEXT), (Population : INT))
- La table "countryLanguage" contient des informations sur les langues parlées dans chaque pays et le pourcentage de la population qui les parlent.
countryLanguage = ((CountryCode : INT), (Language : TEXT), (Percentage : INT))

L'utilisation de l'aide mémoire SQL donné en annexe est utile pour répondre à certaines questions.

- Identifier les clés primaires des tables "country" et "city" et justifier que le couple (CountryCode, Language) constitue une clé primaire de la table "countryLanguage".
D'après l'énoncé, l'attribut Code est la clé primaire de la table "country" et l'attribut ID est la clé primaire de la table "city".
Pour la table "countryLanguage", le seul attribut CountryCode ou Language ne permet pas d'identifier une entité de manière univoque (plusieurs langues dans un même pays ou plusieurs pays ayant la même langue). Il faut donc choisir le couple (CountryCode, Language) comme clé primaire de la table "countryLanguage".
- Établir le schéma relationnel de cette base de données, en faisant apparaitre les clés étrangères.
L'attribut CountryCode de la table "city" est une clé étrangère qui référence la clé primaire Code de la table "country".
L'attribut CountryCode de la table "countryLanguage" est une clé étrangère qui référence la clé primaire Code de la table "country".
L'attribut Capital de la table "country" est une clé étrangère qui référence la clé primaire ID de la table "city".

country							
Code	Name	Continent	SurfaceArea	Population	LifeExp	GNP	Capital

city			
ID	Name	CountryCode	Population

countryLanguage		
CountryCode	Language	Percentage

- Déterminer le nom des villes et leur population de la table "city" dont la population est supérieure à 1000000.
 - SELECT Name, Polulation FROM city WHERE Population > 1000000
- Déterminer le nombre de villes de la table "city".
 - SELECT COUNT(*) FROM city
- Lister les villes de noms différents dans la table "city".
 - SELECT DISTINCT Name FROM city
- Déterminer le nombre de noms de villes différents dans la table "city".

- SELECT COUNT (DISTINCT Name) FROM city

- Déterminer le noms des pays de la table "country" situées en "Europe" classés par ordre décroissant de population.
 - SELECT Name FROM country WHERE Continent="Europe" ORDER BY Population DESC
- Déterminer la population du pays le plus peuplé d'Afrique ("Africa").
 - SELECT Max(Population) FROM country WHERE Continent="Africa"
- Déterminer le nom du pays le plus peuplé d'Afrique ("Africa"), en réutilisant la requête précédente.
 - SELECT Name FROM country
WHERE Continent="Africa"
AND Population = (SELECT Max(Population) FROM country WHERE Continent="Africa")
- Déterminer l'espérance de vie moyenne des pays d'Europe ayant un PNB (GNP dans la table) supérieur à 1000000.
 - SELECT AVG(LifeExpectancy) FROM country WHERE GNP > 1000000
- Déterminer le PNB par habitant des pays d'Europe.
 - SELECT GNP / Population FROM country WHERE Continent="Europe"
- Déterminer la superficie de l'Afrique.
 - SELECT SUM(SurfaceArea) FROM country WHERE Continent="Africa"
- Déterminer la liste des villes d'Europe avec leur pays d'appartenance.
 - SELECT city.Name, country.Name
FROM city INNER JOIN country ON CountryCode = Code
- Déterminer le nombre de langues parlées en Europe.
 - SELECT COUNT(DISTINCT Language)
FROM country INNER JOIN countryLanguage ON Code = countryCode
WHERE Continent="Europe"
- Déterminer la liste des noms des pays où plus d'un quart de la population parle français ("French").
 - SELECT country.Name
FROM countryLanguage INNER JOIN country ON CountryCode = Code
WHERE Language="French" AND Percentage > 25
- Déterminer la liste des noms des pays Européens dont la population est inférieure à la moyenne mondiale.
 - SELECT Name FROM country
WHERE Continent='Europe'
AND Population < (SELECT AVG(Population) FROM country)
- Déterminer le nom des pays où l'on parle l'anglais et le français.
 - SELECT Name
FROM countrylanguage AS L1 INNER JOIN countrylanguage AS L2 INNER JOIN country
ON L1.countrycode=L2.countrycode AND L1.countrycode=Code
WHERE L1.language="French" AND L2.language="English"
OU BIEN
(SELECT Name FROM country JOIN countrylanguage ON CountryCode=Code WHERE Language="French")
INTERSECT
(SELECT Name from country JOIN countrylanguage ON CountryCode=Code WHERE Language="English")

18. Déterminer le nom des pays où l'on parle l'anglais ou le français.

- SELECT Name
FROM countrylanguage AS L1 INNER JOIN countrylanguage AS L2 INNER JOIN country
ON L1.countrycode=L2.countrycode AND L1.countrycode=Code
WHERE L1.language="French" OR L2.language="English"
OU BIEN
- (SELECT Name FROM country JOIN countrylanguage ON CountryCode=Code WHERE Language="French")
UNION
(SELECT Name from country JOIN countrylanguage ON CountryCode=Code WHERE Language="English")

19. Déterminer le nom des pays où l'on parle le français mais pas l'anglais.

- (SELECT Name FROM country JOIN countrylanguage ON CountryCode=Code WHERE Language="French")
EXCEPT
(SELECT Name from country JOIN countrylanguage ON CountryCode=Code WHERE Language="English")

20. Déterminer le nombre de villes de France dont la population dépasse la population moyenne des villes de France.

- SELECT COUNT(*)
FROM city INNER JOIN country ON countryCode=Code
WHERE country.Name='France'
AND city.Population >
(SELECT AVG(city.Population) FROM city INNER JOIN country ON countryCode=Code WHERE coun-
try.Name='France')

21. Déterminer le pourcentage de pays d'Europe pour lesquels le PNB par habitant est supérieur à la moyenne des pays de l'Europe.

- SELECT
COUNT(*)*100 / (SELECT COUNT(*) FROM country WHERE Continent="Europe")
FROM country WHERE Continent="Europe"
AND GNP / Population
< (SELECT AVG(GNP) / Population FROM country WHERE Continent="Europe")

Des requêtes

CREATE TABLE table

Crée la table nommée table

SELECT col1, col2, ... FROM table;

Renvoie un tableau formé des colonnes nommées col1, col2 de la table nommée table. On peut utiliser * pour sélection-
ner toutes les colonnes et aussi la clause WHERE, en fin de requête, pour ne sélectionner que certains enregistrements.

INSERT INTO table(col1, col2, col3, ...) VALUES (val1, val2, val3, ...);

Ajoute un nouvel enregistrement dans la table nommée table en affectant aux champs col1, col2, col3, ... les valeurs val1,
val2, val3, ... Les champs des colonnes absentes, qui ne sont pas de type auto-incrémenté, prennent la valeur NULL.

UPDATE table SET col1 = val1, col2 = val2, ... WHERE condition

Met à jour les colonnes col1, col2, ... de la table nommée table pour les enregistrements vérifiant la condition.

SELECT col1, col2, ... FROM table1 INNER JOIN table2 ON colA=colB...

Réalisation d'une jointure. Chaque colonne de col1, col2, ... appartient à l'une des deux tables, *colA* appartient à une
table et *colB* à l'autre.

SELECT ... FROM table1 ... UNION SELECT ... FROM table2 ...;

Combine les résultats de plusieurs sélections d'enregistrements dans un unique tableau, chaque enregistrement n'étant
sélectionné qu'une seule fois même s'il est présent dans les deux tables.

SELECT ... FROM table1 ... INTERSECT SELECT ... FROM table2 ...;

Cette commande permet donc de sélectionner les enregistrements communs à deux sélections.

SELECT ... FROM table1 ... EXCEPT SELECT ... FROM table2 ...;

Cette requête permet de sélectionner les résultats de table1 sans inclure les enregistrements de la table1 qui sont aussi
dans la table2.

Des commandes et opérateurs

WHERE - Commande qui dans une requête permet de sélectionner les enregistrements d'une table qui respectent une
condition.

ORDER BY col ASC - Permet d'obtenir un résultat d'une requête où les enregistrements sont triés dans l'ordre croissant
suivant les valeurs du champ col. Il est possible de trier les données sur une ou plusieurs colonnes, par ordre croissant
(ASC) ou décroissant (DESC).

DISTINCT - Placée après SELECT dans une requête, permet de ne sélectionner que des enregistrements distincts deux
à deux.

AND, OR, NOT - Opérateurs logiques pour les clauses WHERE et ON.

Des fonctions d'agrégation

MAX - Retourne la valeur maximale d'une colonne dans un ensemble d'enregistrement. La fonction peut s'appliquer à
des données numériques ou alphanumériques.

MIN - Retourne la valeur minimale d'une colonne d'un ensemble d'enregistrements. La fonction peut s'appliquer à des
données numériques ou alphanumériques.

SUM - Retourne la somme des valeurs d'une colonne dans un ensemble d'enregistrements.

AVG - Retourne la moyenne des valeurs d'une colonne dans un ensemble d'enregistrements.

COUNT - Retourne le nombre d'enregistrement d'un ensemble d'enregistrements.