Corrigé

informatique commune

Bases de données relationnelles

Exercice 1.

1. *La liste des continents et de leurs superficies, triés par superficie croissante.*

```sql
SELECT * FROM continent ORDER BY area ASC
```

2. *La liste des capitales mondiales, triées par ordre alphabétique.*

```sql
SELECT capital FROM country ORDER BY capital ASC
```

3. *Le nom et la ville de fondation des quatre plus anciennes organisations mondiales.*

```sql
SELECT name, city , established FROM organization
  ORDER BY established ASC
  FETCH FIRST 4 ROWS ONLY
```

4. *Le nom des volcans de plus de 6 000 mètres.*

```sql
SELECT name FROM mountain WHERE type = 'volcano' AND elevation > 6000
```

5. *Les dix villes les plus peuplées au monde.*

```sql
SELECT name, population FROM city
  WHERE population IS NOT NULL
  ORDER BY population DESC
  FETCH FIRST 10 ROWS ONLY
```

Exercice 2.

1. *Le nom des pays membres des Nations Unies, triés par ordre alphabétique.*

```sql
SELECT c.name FROM country c JOIN ismember i ON c.code = i.country
  WHERE i.organization = 'UN'
  ORDER BY c.name ASC
```

2. *Les monarchies constitutionnelles.*

```sql
SELECT c.name FROM country c JOIN politics p ON c.code = p.country
  WHERE p.government = 'constitutional monarchy'
```

3. *Les fleuves de France.*

```sql
SELECT DISTINCT r.name
  FROM river r JOIN Geo_River gr ON gr.river = r.name
               JOIN country c ON c.code = gr.country
  WHERE r.sea IS NOT NULL AND c.name = 'France'
```

4. *Le nom des montagnes d'Alaska de plus de 5 000 mètres.*

```sql
SELECT m.name, m.elevation
  FROM mountain m JOIN geo_mountain gm ON gm.mountain = m.name
  WHERE m.elevation > 5000 AND gm.province='Alaska'
```

5. *Les trois plus hautes montagnes africaines.*

```sql
SELECT m.name, m.elevation FROM mountain m
    JOIN geo_mountain gm ON m.name = gm.mountain
    JOIN country c ON gm.country = c.code
    JOIN encompasses e ON c.code = e.country
  WHERE e.continent = 'Africa'
  ORDER BY m.elevation DESC
  FETCH FIRST 3 ROWS ONLY
```

Exercice 3.

1. *La longueur moyenne des fleuves qui se jettent dans la Mer Noire.*

```sql
SELECT AVG (length) FROM river WHERE sea = 'Black Sea'
```

2. *Le nombre de rivières françaises présentes dans la base de données.*

```sql
SELECT COUNT(DISTINCT gm.river)
  FROM geo_river gm JOIN country c ON gm.country = c.code
  WHERE c.name = 'France'
```

3. *Le nombre total de pays traversés par chacun des fleuves se jetant dans la mer Méditerranée.*

```sql
SELECT name, COUNT(DISTINCT gr.country)
  FROM river r JOIN geo_river gr ON r.name = gr.river
  WHERE sea = 'Mediterranean Sea'
  GROUP BY r.name
```

4. *Les organisations regroupant plus de 100 pays, avec la population totale de ceux-ci.*

```sql
SELECT o.name, SUM(c.population)
  FROM organization o JOIN ismember i ON i.organization = o.abbreviation
                      JOIN country c ON c.code = i.country
  GROUP BY o.name HAVING COUNT(*) > 100
```

5. *La liste des pays des Amériques avec leur plus haute montagne.*

```sql
SELECT c.name, m.name, MAX(m.elevation)
  FROM geo_mountain gm JOIN mountain m ON gm.mountain = m.name
                       JOIN country c ON gm.country = c.code
                       JOIN encompasses e ON c.code = e.country
  WHERE e.continent = 'America'
  GROUP BY c.name
```

Exercice 4.

1. *Le nom des pays non membres des Nations Unies.*

```
SELECT name FROM country
  EXCEPT
SELECT c.name FROM country c JOIN ismember i ON c.code = i.country
  WHERE i.organization = 'UN'
```

2. *Les affluents directs du Nil ainsi que les affluents de ses affluents.*

```
SELECT name FROM river WHERE river = 'Nile'
UNION
SELECT name FROM river WHERE river IN
  (SELECT name FROM river WHERE river = 'Nile')
```

3. *Les pays frontaliers de la France, triés par population décroissante.*

```
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 c.population DESC)
```

4. *La longueur de la frontière française.*

```
SELECT SUM(length) FROM
  (SELECT b.length FROM country c JOIN borders b ON c.code = b.country1
   WHERE b.country2 = 'F'
   UNION
   SELECT b.length FROM country c JOIN borders b ON c.code = b.country2
   WHERE b.country1 = 'F')
```

5. *Le nombre de voisins de chaque pays d'Europe.*

```
SELECT name, SUM(co) FROM
  (SELECT c.name, COUNT(*) AS co
     FROM country c JOIN encompasses e ON c.code = e.country
                    JOIN borders b ON c.code = b.country1
     WHERE e.continent = 'Europe'
     GROUP BY c.name
   UNION
   SELECT c.name, COUNT(*) AS co
     FROM country c JOIN encompasses e ON c.code = e.country
                    JOIN borders b ON c.code = b.country2
     WHERE e.continent = 'Europe'
     GROUP BY c.name)
  GROUP BY name
```

6. *Les pays qui partagent une même montagne.*

```
SELECT DISTINCT m1.name, c1.name
  FROM mountain m1 JOIN geo_mountain gm1 ON m1.name = gm1.mountain
                   JOIN country c1 ON c1.Code = gm1.country
  WHERE m1.name IN
    (SELECT m2.name FROM mountain m2
                    JOIN geo_mountain gm2 ON m2.name = gm2.mountain
     AND gm1.country <> gm2.country)
```