Exercice 1

1.

2. (a)

```
SELECT nom FROM editeurs
```

(b)

```
SELECT nom FROM jeux
WHERE prix < 15
```

(c)

```
SELECT prenom, nom FROM joueurs
ORDER BY naissance
LIMIT 1
```

(d)

```
SELECT prenom, nom FROM joueurs
WHERE inscription - naissance = 18
```

(e)

```
SELECT temps_jeu FROM jeu_joueur
ORDER BY temps_jeu DESC
LIMIT 1 OFFSET 2
```

(f)

```
SELECT prenom, j.nom, naissance
FROM joueurs AS j
JOIN villes AS v ON idville = ville
WHERE v.nom = "Paris"
```

(g)

```
SELECT prenom, nom FROM joueurs
JOIN jeu_joueur ON joueur = idjoueur
GROUP BY joueur
HAVING COUNT(*) =
    (SELECT COUNT(*) FROM jeu_joueur
     GROUP BY joueur
     ORDER BY COUNT(*) DESC
     LIMIT 1)
```

(h)

```
SELECT e.nom, AVG(prix) FROM jeux
JOIN editeurs AS e ON editeur = idediteur
GROUP BY editeur
```

(i)

```
SELECT e.nom, SUM(prix) FROM jeux
JOIN editeurs AS e ON editeur = idediteur
JOIN jeu_joueur ON jeu = idjeu
GROUP BY editeur
```

(j)

```sql
SELECT nom FROM jeux
JOIN jeu_joueur ON jeu = idjeu
GROUP BY jeu
ORDER BY COUNT(*) DESC
LIMIT 5
```

(k)

```sql
SELECT nom, SUM(prix) FROM jeux
JOIN jeu_joueur ON jeu = idjeu
GROUP BY jeu
ORDER BY SUM(prix) DESC
LIMIT 5
```

(l)

```sql
SELECT nom FROM categories
JOIN jeu_categorie ON categorie = idcategorie
GROUP BY categorie
ORDER BY COUNT(*) DESC
LIMIT 1
```

(m)

```sql
SELECT nom FROM categories
JOIN jeu_categorie AS jc ON categorie = idcategorie
JOIN jeu_joueur AS jj ON jj.jeu = jc.jeu
GROUP BY categorie
ORDER BY COUNT(*) DESC
LIMIT 1
```

(n)

```sql
SELECT  villes.nom FROM villes
JOIN joueurs ON ville = idville
GROUP BY ville
ORDER BY COUNT(*) * 1.0 / habitants DESC
LIMIT 1
```

(o)

```sql
SELECT nom, AVG(note) FROM jeux
JOIN jeu_joueur ON jeu = idjeu
WHERE note IS NOT NULL
GROUP BY jeu
HAVING AVG(note) > 8
```

(p)

```sql
SELECT j2.nom, COUNT(*) FROM jeu_joueur AS jj1
JOIN jeux AS j1 ON j1.idjeu = jj1.jeu
JOIN jeu_joueur AS jj2 ON jj1.joueur = jj2.joueur
JOIN jeux AS j2 ON j2.idjeu = jj2.jeu
WHERE j1.nom = "Wizard of Hell 2"
GROUP BY j2.idjeu
```

(q)
```sql
SELECT DISTINCT j2.prenom, j2.nom FROM joueurs AS j1
JOIN jeu_joueur AS jj1 ON j1.idjoueur = jj1.joueur
JOIN jeu_joueur AS jj2 ON jj1.jeu = jj2.jeu
JOIN joueurs AS j2 ON j2.idjoueur = jj2.joueur
WHERE j1.prenom = "Laurent"
AND j1.nom = "DUPONT"
```

(r)
```sql
SELECT e.nom FROM editeurs AS e
JOIN joueurs AS j ON ville = siege
JOIN jeu_joueur ON joueur = idjoueur
JOIN jeux ON jeu = idjeu
WHERE editeur = idediteur
GROUP BY editeur
ORDER BY COUNT(*) DESC
LIMIT 1
```

Exercice 2

1.
```sql
SELECT capital FROM country
```

2.
```sql
SELECT name FROM lake
WHERE depth > 500
```

3.
```sql
SELECT name FROM country
ORDER BY area
```

4.
```sql
SELECT name, population FROM city
ORDER BY population DESC
LIMIT 1 OFFSET 4
```

5.
```sql
SELECT ci.name FROM city AS ci
JOIN country AS co ON code = country
WHERE co.name = 'France'
ORDER BY latitude DESC
LIMIT 1
```

6.
```sql
SELECT COUNT(*) FROM lake
WHERE type = 'salt'
AND latitude > 0
```

7.
```sql
SELECT co.name, AVG(ci.population) FROM country AS co
JOIN city AS ci ON code = country
GROUP BY country
```

8.
```
SELECT AVG(population / area) FROM province
```

9.
```
SELECT population / area FROM province
ORDER BY population / area
LIMIT 1 OFFSET (SELECT COUNT(*) / 2 FROM province)
```

10.
```
SELECT SUM(length) FROM borders
JOIN country AS c1, country AS c2
ON c1.code = country1 AND c2.code = country2
WHERE c1.name = 'France' OR c2.name = 'France'
```

11.
```
SELECT COUNT(*) FROM river
WHERE sea = 'Pacific Ocean'
```

12.
```
SELECT sea.name, MAX(length) FROM sea
JOIN river ON river.sea = sea.name
GROUP BY sea.name
```

13.
```
SELECT sea.name, r1.name, length FROM sea
JOIN river AS r1 ON r1.sea = sea.name
WHERE length =
(SELECT MAX(r2.length) FROM river AS r2
 WHERE r2.sea = sea.name)
GROUP BY sea.name
```

14.
```
SELECT name FROM river
WHERE river = 'Nile'
UNION
SELECT r1.name FROM river AS r1
JOIN river AS r2 ON r1.river = r2.name
WHERE r2.river = 'Nile'
```

15.
```
SELECT mountain FROM geo_mountain
GROUP BY mountain
HAVING COUNT(DISTINCT country) >= 3
```

16.
```
SELECT name FROM mountain,
(SELECT AVG(elevation) AS moy FROM mountain)
WHERE ABS(elevation - moy) < moy / 10
```

17.
```sql
SELECT mw1.sea1 FROM mergeswith AS mw1
JOIN mergeswith AS mw2 ON mw1.sea2 = mw2.sea1 OR mw1.sea2 = mw2.sea2
WHERE mw2.sea1 = 'Pacific Ocean' OR mw2.sea2 = 'Pacific Ocean'
UNION
SELECT mw1.sea2 FROM mergeswith AS mw1
JOIN mergeswith AS mw2 ON mw1.sea1 = mw2.sea1 OR mw1.sea1 = mw2.sea2
WHERE mw2.sea1 = 'Pacific Ocean' OR mw2.sea2 = 'Pacific Ocean'
```

18.
```sql
SELECT name FROM country
WHERE code NOT IN
(SELECT country FROM ismember
 WHERE organization = 'UN')
```

19.
```sql
SELECT AVG(nbMembres) FROM
(SELECT organization, COUNT(*) AS nbMembres FROM ismember
 GROUP BY organization)
JOIN organization AS o, country as c
ON organization = abbreviation
AND code = o.country
WHERE c.name = 'France'
```