

Exercices

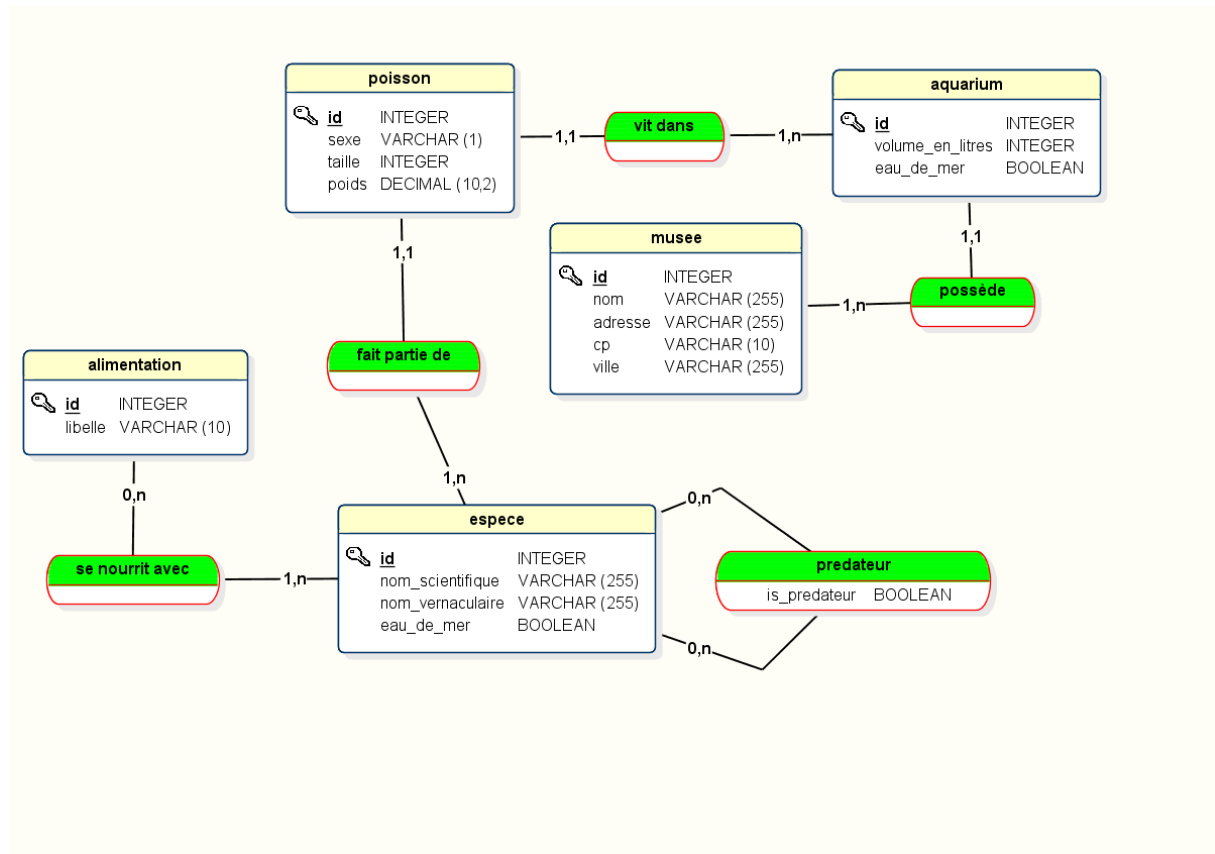
Entités :

- Musée (Nausicaa, Museum d'Histoire Naturel, MarineLand...)
- Aquarium (avec sa taille en litres, le type d'eau)
- Espèce (Avec son nom, son milieu de vie etc...)
- Prédateur (Avec deux ids d'espèces et le lien qui les unit de proie à prédateur, ou non. On ne cherche pas à savoir qui est le prédateur de qui, seulement s'il y a un risque à mettre ces deux espèces de poissons dans le même bassin)
- Poisson (avec la taille, le poids, le sexe... Ici on n'est plus au niveau de l'espèce mais au niveau de l'individu)
- Alimentation (Algues, plancton, autres poissons...)

Règles :

- Un musée possède plusieurs aquariums
- Un aquarium contient plusieurs poissons
- Un aquarium n'appartient qu'à un musée
- Une espèce peut être le prédateur naturel ou la proie d'un autre. S'il existe un risque de conflits entre deux espèces, on veut le noter dans la table « prédateur » pour pouvoir les éviter.
- Un poisson n'appartient qu'à une seule espèce évidemment
- Une espèce peut se nourrir de différentes façons

MCD :



Exercices :

1 – Compter le nombre de poissons par aquarium

```
SELECT aquarium.id as id_aquarium, COUNT(*)
FROM poisson
INNER JOIN aquarium on aquarium.id = poisson.id_aquarium
GROUP BY aquarium.id;
```

2 – Compter le nombre de représentant de chaque espèce dans toute la base

```
SELECT e.nom_vernaculaire, COUNT(*) as nb_poissons
FROM espece e
INNER JOIN poisson p on p.id_espece = e.id
GROUP BY e.nom_vernaculaire;
```

3 – Compter le nombre d'aquarium par musée

```
SELECT musee.nom, COUNT(*) as nb_aquarium
FROM aquarium
INNER JOIN musee ON musee.id = aquarium.id_musee
GROUP BY musee.nom;
```

4 – Calculer la moyenne du nombre de poissons par aquarium dans le musée Nausicaa

5 – Afficher l’aquarium d’eau de mer le plus volumineux du Museum d’Histoire Naturelle

```
SELECT a.*
FROM aquarium a
INNER JOIN musee m ON m.id = a.id_musee
WHERE m.nom = "Museum d'histoire Naturelle"
AND a.eau_de_mer = 1
ORDER BY volume_en_litres DESC
LIMIT 1;
```

6 – Lister toutes les espèces de poissons présentant un risque potentiel avec la « Murène »

```
SELECT e.*
FROM espece e
LEFT JOIN predateur p ON p.id_espece1 = e.id OR p.id_espece2 = e.id
WHERE p.is_predateur = 1
AND (p.id_espece1 = 1 OR p.id_espece2 = 1)
AND e.id != 1;
```

7 – Calculer le volume total des aquariums du musée Nausicaa

```
SELECT SUM(volume_en_litres)
FROM aquarium
INNER JOIN musee on musee.id = aquarium.id_musee
WHERE musee.nom = "Nausicaa"
```

8 – Calculer le volume moyen des aquariums par musée

```
SELECT ROUND(AVG(volume_en_litres), 2) as moyenne_volume, musee.nom
FROM aquarium
INNER JOIN musee on musee.id = aquarium.id_musee
GROUP BY musee.id;
```

9 – Afin de prévoir la tournée d’alimentation des aquariums, lister tous les types d’alimentation requis dans l’aquarium N°4 de Nausicaa

```
SELECT DISTINCT alim.*
FROM alimentation alim
INNER JOIN espece_alimentation ea ON ea.id_alimentation = alim.id
INNER JOIN espece e ON e.id = ea.id_espece
INNER JOIN poisson p ON p.id_espece = e.id
INNER JOIN aquarium aqua ON aqua.id = p.id_aquarium
INNER JOIN musee m ON m.id = aqua.id_musee
WHERE aqua.id = 4
AND m.nom = "Nausicaa";
```

10 – On souhaite insérer un poisson supplémentaire dans l’aquarium N°4. Mais avant de faire ça, on va vérifier si l’espèce « Poisson Clown » ne présente aucun risque avec les autres espèces de l’aquarium

```
SELECT pre.*
FROM predateur pre
INNER JOIN espece e ON e.id = pre.id_espece1 OR e.id = pre.id_espece2
INNER JOIN poisson p ON e.id = p.id_espece
WHERE p.id_aquarium = 4
AND e.id = 2
AND pre.id_espece1 IN (SELECT id_espece FROM poisson WHERE id_aquarium = 4)
AND pre.id_espece2 IN (SELECT id_espece FROM poisson WHERE id_aquarium = 4);
```

11 – Insérer un poisson de l’espèce « Poisson Clown » dans l’aquarium N° 4

```
INSERT INTO poisson (sexe, taille, poids, id_aquarium, id_espece)
VALUES ("M", 30, 1, 4, 2);
```

12 – Chez Nausicaa, on veut alléger notre stock de coquillages en offrant une tournée du patron à tous les poissons qui peuvent en manger. Il faut donc lister tous les aquariums dans lesquels il y a des poissons à nourrir avec des coquillages.

```
SELECT DISTINCT a.*
FROM aquarium a
INNER JOIN musee m ON m.id = a.id_musee
INNER JOIN poisson p ON p.id_aquarium = a.id
INNER JOIN espece e ON e.id = p.id_espece
INNER JOIN espece_alimentation ea ON ea.id_espece = e.id
INNER JOIN alimentation alim ON alim.id = ea.id_alimentation
WHERE alim.libelle = "Coquillages"
AND m.nom = "Nausicaa";
```

13 – Oups, mauvaise idée. Les coquillages étaient avariés et les poissons ont fait une intoxication alimentaire. Nous allons donc devoir mettre à jour la table « Poisson » afin d’y rajouter un champ « vivant » en tant que BOOLEAN qui aura par défaut la valeur 1

```
ALTER TABLE `poisson` ADD `vivant` BOOLEAN NOT NULL DEFAULT 1 AFTER `poids`;
```

14 – Puis nous allons passer le champ « vivant » à « 0 » pour tous les poissons qui mangent des coquillages.

```
UPDATE poisson p
INNER JOIN aquarium aqua ON aqua.id = p.id_aquarium
INNER JOIN musee m ON m.id = aqua.id_musee
INNER JOIN espece e ON e.id = p.id_espece
INNER JOIN espece_alimentation ea ON ea.id_espece = e.id
INNER JOIN alimentation alim ON alim.id = ea.id_alimentation
SET p.vivant = 0
```

```
WHERE alim.libelle = "Coquillages"
AND m.nom = "Nausicaa";
```

15 – Lister tous les aquariums qui ont des poissons morts dedans, afin de pouvoir les repeupler avec de nouveaux poissons.

```
SELECT DISTINCT aquarium.*
FROM aquarium
INNER JOIN poisson p ON p.id_aquarium = aquarium.id
INNER JOIN musee m ON m.id = aquarium.id_musee
WHERE p.vivant = 0
AND m.nom = "Nausicaa";
```

16 – Lister tous les aquariums qui n'ont plus de poissons vivants dedans, afin de pouvoir les repeupler avec de nouveaux poissons.

```
SELECT DISTINCT aquarium.*
FROM aquarium
INNER JOIN poisson p ON p.id_aquarium = aquarium.id
INNER JOIN musee m ON m.id = aquarium.id_musee
WHERE aquarium.id NOT IN (SELECT id_aquarium
                           FROM poisson
                           WHERE vivant = 1)
AND m.nom = "Nausicaa";
```