Métropole – 2026 – sujet1 - Correction

Exercice 3 (8 points)

Partie A – Structure arborescente des arguments

1. Nom du point de départ

C'est la **racine** de l'arbre.

2. Arbre binaire ?

Non : dans un arbre binaire, chaque nœud a au plus deux fils. Ici, le sujet a **quatre** arguments directement rattachés, et une affirmation peut recevoir autant d'arguments qu'on veut.

3. Attributs de la classe Affirmation

attribut	type
contenu	str
sorte	str
arguments	list (liste d'objets Affirmation)
nb_likes	int
nb_dislikes	int

4. Méthode soutenir (lignes 10 à 12)

```
    assert argument.sorte == ""
    self.arguments.append(argument)
    argument.sorte = "pour"
```

👉*Commentaire : une affirmation pas encore utilisée a la sorte "" : l'assertion vérifie la précondition de la documentation.*

5. Méthode nb_contre

```
def nb_contre(self):
    n = 0
    for arg in self.arguments:
        if arg.sorte == "contre":
            n = n + 1
    return n
```

6. Méthode mystere

mystere renvoie **le plus grand nombre d'arguments contre directement rattachés à une même affirmation**, parmi l'affirmation et toutes celles de son sous-arbre (ses arguments, les arguments de ses arguments, etc.).

👉 *Commentaire : pour l'arbre de la figure 1, mystere renvoie 2 : le sujet est attaqué par deux arguments contre.*

7. Méthode evaluation

```
def evaluation(self):
    total = self.nb_likes - self.nb_dislikes
    for arg in self.arguments:
        if arg.sorte == "pour":
            total = total + arg.evaluation()
        else:
            total = total - arg.evaluation()
    return total
```

👉 *Commentaire : c'est une fonction récursive : l'évaluation d'un argument dépend de celle de ses propres arguments. Les arguments d'une affirmation sont « pour » ou « contre » : le else traite les arguments contre. On peut aussi écrire* `if arg.sorte == "contre":`.

Partie B – Base de données

8. Rôles d'un SGBD

Les propositions **a)**, **c)** et **d)** : sécuriser les accès, assurer la persistance des données et gérer les accès concurrents.

👉 *Commentaire : l'alimentation électrique relève du matériel, et HTTPS du serveur web, pas du SGBD.*

9. Autre clé primaire possible pour utilisateur

L'attribut **email** : deux comptes ne peuvent pas avoir la même adresse électronique, elle identifie donc chaque utilisateur de façon unique. Le nom ou le prénom ne conviendraient pas (homonymes).

10. Affirmations avec au moins 50 likes

```
SELECT COUNT(*)
FROM affirmation
WHERE nb_likes >= 50;
```

11. Sujets de Pierre Durand

```sql
SELECT affirmation.contenu, affirmation.nb_likes
FROM affirmation
JOIN utilisateur ON affirmation.auteur = utilisateur.pseudo
WHERE affirmation.sorte = 'sujet'
AND utilisateur.prenom = 'Pierre' AND utilisateur.nom = 'Durand';
```

12. Interprétation de la requête

Elle affiche les couples (affirmation, argument contre) où l'argument **contre** une affirmation a obtenu **au moins deux fois plus de likes** que l'affirmation qu'il attaque : le contenu de l'affirmation et celui de l'argument.

👉 *Commentaire : dans l'énoncé, la requête utilise rep.affirmation_repondue et aff.id_affirmation : d'après le schéma, ces attributs s'appellent aff_repondue et id_aff.*

13. Like de i<3descartes sur l'affirmation 108

```sql
INSERT INTO reaction
VALUES (108, 'i<3descartes', 'like');

UPDATE affirmation
SET nb_likes = nb_likes + 1
WHERE id_aff = 108;
```

👉 *Commentaire : les valeurs suivent l'ordre des attributs de reaction : aff_reagie, utilisateur, sorte.*

14. Supprimer les réactions avant l'utilisateur

Dans la table reaction, l'attribut utilisateur est une **clé étrangère** qui référence le pseudo de la table utilisateur. Si on supprimait d'abord l'utilisateur, ses réactions feraient référence à un utilisateur inexistant : la contrainte d'**intégrité référentielle** serait violée et le SGBD refuserait la suppression.

👉 *Commentaire : c'est aussi pour cela qu'on anonymise d'abord ses affirmations : auteur est une clé étrangère vers utilisateur.*

15. Suppression du compte i<3rgpd

```sql
UPDATE affirmation SET auteur = NULL WHERE auteur = 'i<3rgpd';
DELETE FROM reaction WHERE utilisateur = 'i<3rgpd';
DELETE FROM utilisateur WHERE pseudo = 'i<3rgpd';
```