Contrôle 01

Partie 1 : sur papier

Exercice 1 - Fonction d'Ackermann-Péter

4pts

Cette fonction est définie pour tout $m \in \mathbb{N}$ et tout $n \in \mathbb{N}$ par

$$A(m,n) = \begin{cases} n+1 & \text{si } m = 0 \\ A(m-1,1) & \text{si } m > 0 \text{ et } n = 0 \\ A(m-1, A(m,n-1)) & \text{si } m > 0 \text{ et } n > 0 \end{cases}$$

Écrire la fonction A en PYTHON.

Réponse

On peut recopier tel quel, mais on peut aussi utiliser `if elif else` comme ceci :

```
def A(m: int, n: int) -> int:
    if m == 0:
        return n + 1
    elif n == 0:
        return A(m - 1, n)
    else:
        return A(m - 1, A(m, n - 1))
```

Exercice 2 - Fonction mystère

4pts

Que fait la fonction suivante ? Ne pas expliquer le code ligne par ligne, dire ce que renvoie la fonction par rapport au paramètre d'entrée.

Code Python

```
def mystery(lst: list) -> list:
    if len(lst) < 2:
        return lst
    else:
        return [lst[-1]] + mystery(lst[1:-1]) + [lst[0]]
```

1. Que renvoie `mystery([1])` ?

Réponse

`mystery([1])` renvoie `[1]` : c'est un cas d'arrêt.

2. Que renvoie `mystery([1, 2, 3])` ?

Réponse

`mystery([1, 2, 3])` renvoie `[3] + mystery([2]) + [1]` ce qui donne `[3, 2, 1]`.

3. Quel est le rôle de cette fonction ?

Réponse

Lorsque la liste est vide ou contient un seul élément, la fonction renvoie cette même liste, sinon elle une liste dont le premier élément est le dernier de `lst`, dont le dernier est le premier de `lst` et dont la liste du « milieu » est le résultat d'un appel récursif.
Pour faire simple, elle permute les extrémités de la liste et fait de même avec la sous-liste composée de cette liste sans les extrémités.
Ainsi la fonction renvoie la liste de départ en inversant l'ordre des éléments.

Exercice 3

On veut construire une base de données d'œuvres artistiques. Voici un résumé des données que l'on a récoltées :

Œuvre

id	titre	creation	id_categorie	id_artiste
130	Cheval et cavalier	1511	5	40
196	Colombe de la paix	1949	2	56
454	Guernica	1937	4	56
546	L'Homme de Vitruve	1490	2	40
591	L'escalier de Chambord	1516	3	40
634	La Cène	1498	4	40
649	Les Éléphants	1948	4	78
685	La Girafe en feu	1937	4	78
706	La Joconde	1519	4	40
...	...	...	...	...

Artiste

id	nom	naissance	mort
40	Léonard de Vinci	1452	1519
56	Pablo Picasso	1881	1973
78	Salvador Dali	1904	1989

Catégorie

id	classement
1	céramique
2	dessin
3	objet
4	peinture
5	sculpture

Seul le premier tableau n'est pas donné en entier car il comporte trop de lignes.

Donner le modèle relationnel en ligne de la BDD. Ne pas oublier d'indiquer les types des attributs, de souligner en trait plein les clés primaires et en pointillés les clés étrangères.

Réponse

Voici un modèle possible :
Artiste(id_artiste INTEGER, nom TEXT, naissance DATE, mort DATE)

Categorie(id_categorie INTEGER, classement TEXT)

Œuvre(id INTEGER, titre TEXT, creation DATE, id_categorie, id_artiste)

Exercice 4 - CinéHit, c'est plus de hits! 16pts

Voici le schéma relationnel de la base de données du site de cinéphiles CinéHit :

Schéma relationnel

- **Realisateur**(id_realisateur INTEGER, nom VARCHAR(48))

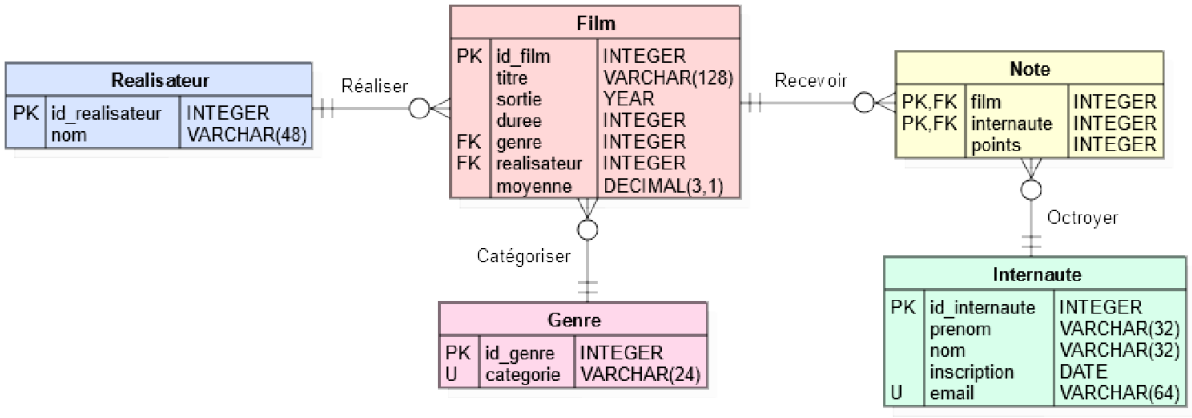
- **Internaute**(id_internaute INTEGER, prenom VARCHAR(32), nom VARCHAR(32), inscription DATE, email VARCHAR(64))
  Avec la contrainte utilisateur que l'inscription est postérieure à `2003-01-01` et que l'email est unique.

- **Genre**(id_genre INTEGER, categorie VARCHAR(24))
  Avec la contrainte que la catégorie est unique.

- **Film**(id_film INTEGER, titre VARCHAR(128), sortie DATE, duree INTEGER, genre INTEGER, realisateur INTEGER, moyenne DECIMAL)
  Avec la contrainte que la sortie est postérieure à 1895.
  La clé étrangère genre référence la clé primaire id_genre de la relation **Genre**, et la clé étrangère realisateur référence la clé primaire id_realisateur de la relation **Realisateur**.
  L'attribut duree est la durée du film convertie en minutes, et l'attribut moyenne est un nombre décimal entre 0 et 10 et comportant un chiffre après la virgule.

- **Note**(film INTEGER, internaute INTEGER, points INTEGER)
  Avec la contrainte que les points sont entre 0 et 10.
  La clé primaire est le couple (film, internaute) formé de la clé étrangère film qui référence la clé primaire id_film de la relation **Film** et la clé étrangère internaute qui référence la clé id_internaute de la relation **Internaute**.

On peut résumer tout ceci par le diagramme ci-dessous.



1. Que fait la requête `SELECT categorie FROM Genre;` ?

Réponse

Cette requête produit la table de toutes les catégories qui figurent dans la table **Genre**.

2. Que fait la requête `SELECT * FROM Realisateur WHERE nom LIKE 'Jean%';` ?

Réponse

Cette requête produit la table des tuples de **Realisateur** dont le nom commence par Jean.

3. Que fait la requête suivante ?

```
SELECT titre,categorie
FROM Film
    JOIN Genre ON Film(genre) = Genre(id_genre);
```

Réponse

Cette requête affiche le titre et la catégorie correspondante de tous les films.

4. Quelle requête donne la table complète des films sortis en 1984 ?

Code SQL

```
SELECT * FROM Film WHERE sortie = 1984;
```

5. Quelle requête donne la table complète des internautes inscrits entre début 2018 et fin 2019 ?

Code SQL

```
SELECT * FROM Internaute
WHERE inscription BETWEEN '2018-01-01' AND '2019-12-31';
```

6. Quelle requête donne la table des titres et moyenne des films durant au moins 3h ?

Code SQL

```sql
SELECT titre,moyenne FROM Film WHERE duree >= 180;
```

7. Quelle requête donne la table des titres des films sortis après 2000 et dont la moyenne est supérieure à 8 ?

Code SQL

```sql
SELECT titre FROM Film WHERE sortie > 2000 AND moyenne >= 8;
```

8. Quelle requête donne la table complète des films d'Alfred Hitchcock ?

Code SQL

```sql
SELECT * FROM Film
    JOIN Realisateur ON Film.realisateur = Realisateur.id_realisateur
WHERE nom = 'Alfred Hitchcock';
```

9. Quelle requête donne la table donnant l'identifiant, le titre et l'année de sortie des thrillers ?

Code SQL

```sql
SELECT id_film,titre,sortie FROM Film
    JOIN Genre ON genre = id_genre
WHERE categorie = 'thriller';
```

10. Quelle requête donne la table des noms de réalisateurs et titres de films dont la moyenne est 8.5 ou plus ?

Code SQL

```sql
SELECT nom,titre FROM Realisateur
    JOIN Film ON id_realisateur = realisateur
WHERE moyenne >= 8.5;
```

11. Quelle requête donne la table des prénoms et noms des internautes ayant octroyé 1 seul point à un film, en supprimant les doublons ?

Code SQL

```sql
SELECT DISTINCT prenom, nom FROM Internaute
    JOIN Note ON id_internaute = internaute
WHERE points = 1;
```

12. Quelle requête donne la table des adresses email et des points des internautes qui ont notés « Les Profs » ?

Code SQL

```sql
SELECT email,points FROM Internaute
    JOIN Note ON id_internaute = internaute
    JOIN Film ON film = id_film
WHERE titre = 'Les Profs';
```

13. La requête suivante échoue. Émettre une hypothèse pour expliquer cet échec.

```sql
INSERT INTO Note VALUES(248, 93, 17);
```

Réponse

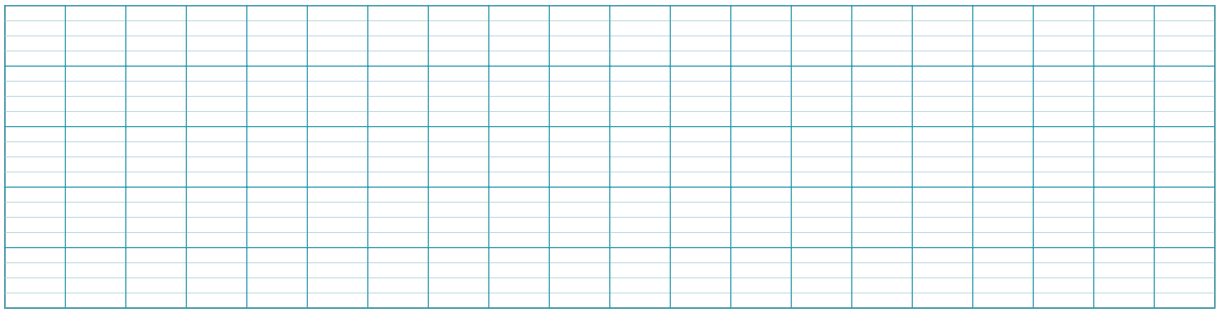
Le nombre de points attribués n'est pas compris entre 0 et 10.

14. Expliquer pourquoi la requête suivante échoue :

```sql
DELETE FROM Internaute WHERE id_internaute = 50;
```

Réponse

Un enregistrement de la table **Note** possède une clé étrangère internaute qui se réfère à la clé primaire id_internaute de valeur 50.



15. Donner les requêtes SQL permettant d'inscrire l'utilisateur et le film suivant :
Léo Part s'est inscrit sur le site le 21 juin 2020 avec l'adresse mail `leo.part@alapla.ge`. Il a l'identifiant 104.
Il a mis la note de 9 sur 10 au film de science-fiction « Contact » de Robert Zemeckis qui est sorti en 1997 et qui dure 2 h 33 min.
Le genre de ce film est 13 et son identifiant 694.

Code SQL

```sql
INSERT INTO Internaute VALUES(104, "Léo", "Part", "2020-06-21",
 ↪  "leo.part@alapla.ge");
INSERT INTO Film VALUES(694, "Contact", 1997, 153, 13, 129, 9.0);
INSERT INTO Note VALUES(694, 104, 9);
```

16. Donner les requêtes SQL afin de supprimer le film « Fatal » de la BDD du site CinéHit tout en maintenant son intégrité. L'identifiant du film est 446.

Code SQL

```sql
DELETE FROM Note WHERE film = 446;
DELETE FROM Film WHERE id_film = 446;
```