Bases de données
Série n°2 : AR ~ SQL / Python ~ SQLITE

Exercice 1 : Concours blanc 2017-2018 ~ CORRECTION

On considère la base de données « **base1** » constituée des trois relations suivantes :

Acteur (**numa**, nom, spec ,taille, poids)

Colonne	Description	Type
numa	Numéro de l'acteur	Entier
nom	Nom de l'acteur	Texte
spec	Spécialité de l'acteur	Texte
taille	Taille	Réel
poids	Poids	Réel

Film (**numf**, titre, genre, annee, duree, budget, realisateur)

Colonne	Description	Type
numf	Numéro du film	Entier
titre	Titre du film	Texte
genre	Genre du film (comique,drame …)	Texte
annee	Année de sortie	Entier
duree	Durée	Entier
budget	budget	Réel
realisateur	Réalisateur	Texte

Jouer (**#numf, #numa**, desc ,sal)

Colonne	Description	Type
numf	N° du film	Entier
numa	N° de l'acteur	Entier
desc	Description du rôle	Texte
sal	Salaire pour le rôle	Réel

- **Les clés primaires sont soulignées et les clés étrangères sont précédées par #.**
- **Tous les attributs doivent être non vides**

1) Donner les instructions **SQL** qui permettent de créer les trois relations de la base de données **« base1 »** en respectant toutes les contraintes de domaines et d'intégrités référentielles.

```
--Question 1
create table Acteur (
    numa    int     primary key not null,
    nom     text    not null,
    spec    text    not null,
    taille  real    not null,
    poids   real    not null
)

create table Film (
    numf    int     primary key not null,
    titre   text    not null,
    genre   text    not null,
    annee   int     not null,
    duree   int     not null,
    budget  real    not null,
    realisateur text not null )
```


anis_saied@hotmail.com

```
18.
19. create table Jouer (
20.     numf    int     not null,
21.     numa    int not null,
22.     description     text    not null,
23.     sal real    not null,
24.     primary key(numf,numa),
25.     foreign key (numf) references Film(numf),
26.     foreign key (numa) references Acteur(numa)
27. )
```

2) Ecrire un script python qui :

- Charge le module **sqlite3** ;
- Etablit une connexion avec la base **« base1 » ;**
- Insère dans la table **Acteur** autant de lignes que l'utilisateur le souhaite ;
- Et affiche le contenu de la table **Acteur**.

```
1. #Question 2
2. import sqlite3
3. con = sqlite3.connect("base1.db")
4.
5. requete ="insert into Acteur values (?,?,?,?,?)"
6.
7. L=[]
8. while 1:
9.     while 1:
10.         try:
11.             numa = int(input("numéro acteur ="))
12.             taille = float(input("numéro acteur ="))
13.             poids = float(input("numéro acteur ="))
14.             break
15.         except:
16.             continue
17.
18.     nom = input("nom acteur =")
19.     spec = input("spec acteur =")
20.
21.     L.append((numa,nom,spec,taille,poids))
22.
23.     rep = input("Insérer Autre Acteur ? O/N")
24.     if rep == 'N':
25.         break
26.
27. con.executemany(requete,L)
28. con.commit()
29.
30. #Afficher table acteur
31. req = "SELECT * FROM Acteur"
32. cur = con.cursor()
33. cur.execute(req)
34. L = cur.fetchAll()#retourne une liste de tuples
35. print(resultat)
36. for i in range(len(L)):
37.     print("Num:",L[i][0],"Nom:",L[i][1])
38.
39. con.close()
```


anis_saied@hotmail.com

3) Répondre en algèbre relationnelle puis en **SQL** aux requêtes suivantes :

a) Afficher le titre des films qui ont une durée supérieure à 80 min et un budget ne dépassant pas les 100000 dinars ;

```sql
SELECT titre
FROM film
WHERE duree >80 and budget <= 100000
```

b) Afficher les rôles joués par l'actrice « Aymen Sahli ».

```sql
SELECT description
FROM Jouer JOIN Acteur  ON Jouer.numa = Acteur.numa
WHERE nom = 'Aymen Sahli'
```

4) Répondre en **SQL** uniquement aux questions suivantes :

a) Afficher l'intitulé du genre ainsi que le nombre de films correspondants ;

```sql
SELECT genre, count(*) AS 'nb de films'
FROM film
GROUP BY genre
```

b) Afficher les genres qui ont plus que 20 films ;

```sql
SELECT genre, COUNT(*) AS 'Nbre de films'
FROM film
GROUP BY genre
HAVING 'Nbre de films' > 20
```

c) Donner le titre et l'année du/de(s) film(s) le(s) plus long(s) ;

```sql
SELECT titre, annee
FROM film
ORDER BY duree DESC LIMIT 1
```

d) Calculer le total des salaires des acteurs qui ont joué dans « TAXI 2 » ;

```sql
SELECT SUM(sal)
FROM   Jouer JOIN Film ON Film.numf = Jouer.numf
WHERE  titre = 'TAXI 2'
```

e) Afficher les noms des acteurs qui n'ont joué dans aucun film jusqu'à maintenant ;

```sql
SELECT numa FROM Acteur
MINUS
SELECT numa FROM Jouer
```


anis_saied@hotmail.com