

*Mini Projet En BASE DE DONNEES
De Fin De Semestre 2*

Médiathèque
Oracle SQL developer

Encadré par : M. Rim MGHIRBI

Réalisé par : Khouloud SAID,
Eya HAJ AYED,
Manar FARAH,
Yasmine AMERI

Année Universitaire: 2021/2022

INTRODUCTION

Notre problématique, remet en question les difficultés que rencontrent les responsables d'une médiathèque en termes de gestion et d'organisation des informations relatives aux catalogue contenant des livres, des films, et des albums de musique élaborés par des auteurs, réalisateurs, acteurs ou musiciens. .

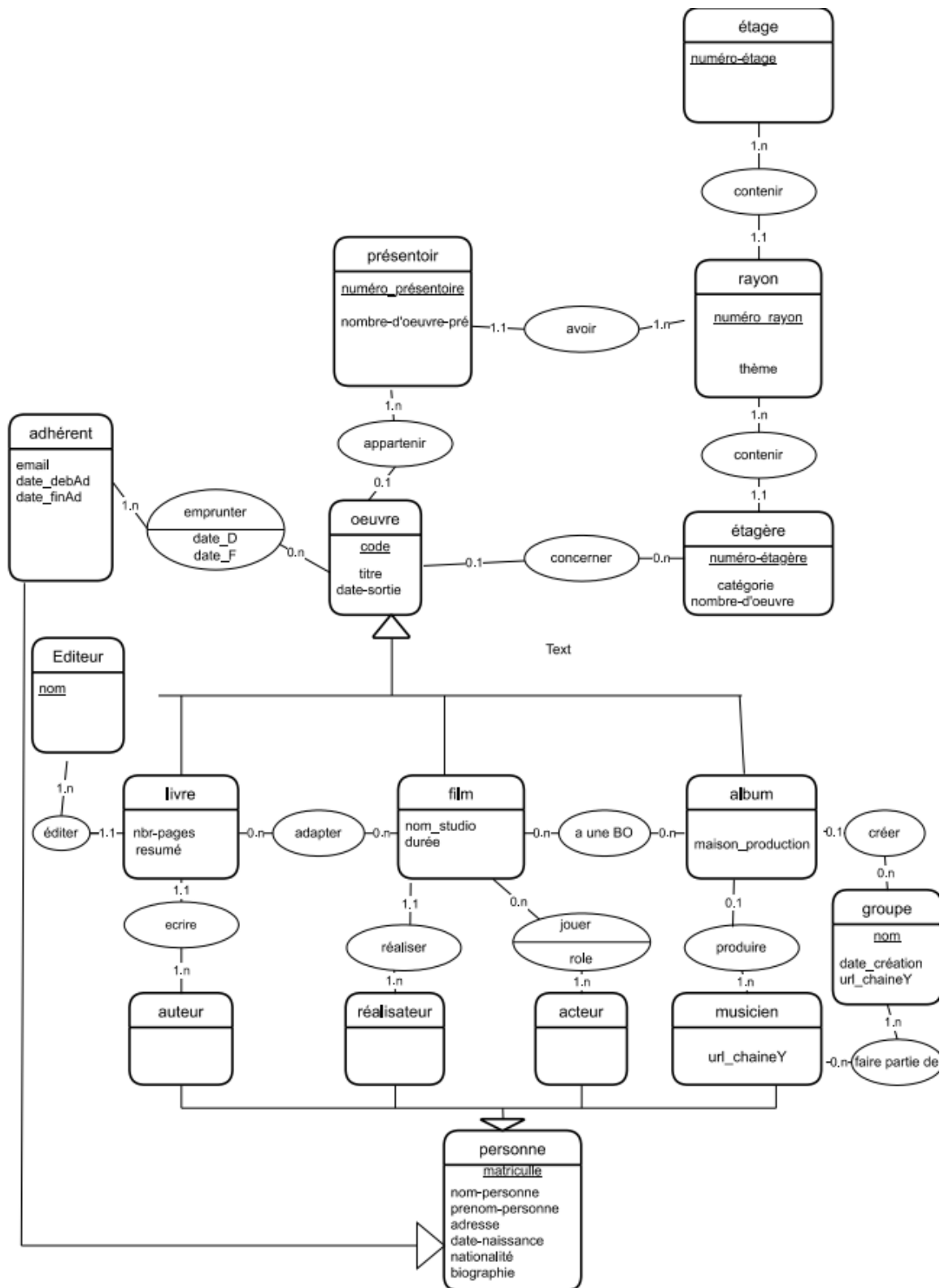
Pour répondre à cette problématique, nous avons procéder à la réalisation d'une base de données permettant d'optimiser la gestion des informations .

Objectifs

- **Minimiser le temps**
- **Permettre l'accès aux données de façon simple**
- **Pouvoir répondre à des requêtes avec un niveau de performances adapté**
- **Faciliter la mise à jour des données**
- **La gestion des œuvres et de leurs stocks**
- **La gestion des clients**
- **La gestion des différents auteurs, artistes, acteurs et éditeurs**

MODEL A E/A:: Présentation

Ce modèle E/A est un formalisme pour la modélisation de données.



MODELE Relationnel

Ce modèle E/A est un formalisme pour la modélisation de données.

ALBUM (codealb , titre , date_sortie ,maison_prod)

GROUPE(nom_grp , date_creation)

MUSICIEN(matricule_mus ,nom, prenom , adresse , nationalité ,date_naissance)

ALBUM_SOLO (#codealb ,#matricule_mus)

ALBUM_GRP (#codealb ,#nom_grp)

ETAGE (num_etage)

RAYON (num_rayon , thème ,#num_etage)

PRESENTOIR (num_presentoir , nbre_oeuvre ,#num_rayon)

ETAGERE (num_etagere ,catégorie ,nbre_oeuvre , #num_rayon)

EDITEUR (nom_maison_edition,site_url ,adresse)

OEUVRE (code ,#id_etagère , #id_presentoire)

AUTEUR (matricule , nom ,prenom , date_naissance , nationalité)

LIVRE (code , nbpages , ISBN ,date_sortie ,titre, #matricule_aut , #nom_maison_edition)

ARTISTE (matricule , nom ,prenom , date_naissance , nationalité)

FILM (code_F , titre, annee, #matricule_real)

FILM_ALBUM (#codealb,#code_F)

FILM_ADAPTE_LIVRE (#codeL , #code_F)

ROLE (ROLEE_NAME ,#code_F_R ,#matricule_act)

ADHERENT (matricule_Adh ,email_adh , phone_adh, date_debadh, date_finadh ,
nom_adh, prenom_adh ,NATIONALITE ,adresse_adh, date_naiss_adh)

EMPRUNT (#code , date_debut ,#matricule_adh , date_fin)

Remarque:

***Un client peut emprunter de 0 à plusieurs œuvres**

***Certains de ces rayons peuvent comprendre à la fois des albums, des livres et des films. .**

***Chaque étage comprend plusieurs rayons**

***Un acteur peut jouer plusieurs rôles dans le même film.**

*Un artiste peut être à la fois un acteur et un réalisateur

*Chaque étagère peut comprendre un certain nombre d'œuvres, mais d'une seule catégorie.

*Les œuvres sont identifiées par des codes

CREATION DES TABLES

Voir les fichiers ci-joints

```
CREATE TABLE album (  
    codealb INT,  
    titre VARCHAR(20),  
    date_sortie date,  
    maison_prod VARCHAR(20),  
    PRIMARY KEY (codealb));  
CREATE TABLE groupe (nom_grp VARCHAR(20),  
    date_creation date,  
    PRIMARY KEY (nom_grp));  
CREATE TABLE musicien (  
    matricule_mus VARCHAR (20),  
    nom VARCHAR (20),  
    prenom VARCHAR (20),  
    adresse VARCHAR (20),  
    nationalite VARCHAR (20),  
    date_naissance date,  
    url_chaine VARCHAR (20),  
    nom_grp VARCHAR (20),  
    PRIMARY KEY (matricule_mus),  
    FOREIGN KEY (nom_grp) REFERENCES groupe(nom_grp));  
CREATE TABLE album_solo (  
    codealb INT,  
    matricule_mus VARCHAR (20),  
    PRIMARY KEY (codealb,matricule_mus),  
    FOREIGN KEY (codealb) REFERENCES album(codealb),  
    FOREIGN KEY (matricule_mus) REFERENCES  
musicien(matricule_mus));  
CREATE TABLE album_grp (codealb INT,  
    nom_grp VARCHAR (20),  
    PRIMARY KEY (codealb),  
    FOREIGN KEY (codealb) REFERENCES album(codealb),  
    FOREIGN KEY (nom_grp) REFERENCES groupe (nom_grp));  
CREATE TABLE Etag (num_etage DECIMAL(5),
```

```

PRIMARY KEY (num_etage));

CREATE TABLE Rayon (num_rayon DECIMAL(5),
                    theme VARCHAR (50),
                    num_etage DECIMAL(5),
                    PRIMARY KEY (num_rayon),
                    FOREIGN KEY (num_etage) REFERENCES
Etage(num_etage));

CREATE TABLE Presentoir (num_presentoir DECIMAL(5),
                          nbre_oeuvre_pre DECIMAL(5),
                          num_rayon DECIMAL(5),
                          PRIMARY KEY (num_presentoir),
                          FOREIGN KEY (num_rayon) REFERENCES
Rayon(num_rayon));

CREATE TABLE Etagere ( num_etagere DECIMAL(5),
                        nbre_oeuvre DECIMAL(5),
                        categorie VARCHAR (50),
                        num_rayon DECIMAL(5),
                        PRIMARY KEY (num_etagere),
                        FOREIGN KEY (num_rayon) REFERENCES
Rayon(num_rayon));

create table editeur (
    nom_maison_edition VARCHAR(50) ,
    site_url VARCHAR(1000),
    adresse VARCHAR(50),
    PRIMARY key (nom_maison_edition)
);

create table oeuvre (
    code INT NOT NULL ,
    id_etagere INT ,
    id_presentoir INT ,
    PRIMARY KEY (code),
    FOREIGN KEY (id_etagere) REFERENCES Etagere(num_etagere),
    FOREIGN KEY (id_presentoir) REFERENCES Presentoir(num_presentoir)
);

create table auteur (
    matricule INT ,
    nom VARCHAR(50) NOT NULL,
    prenom VARCHAR(50),
    date_naissance DATE,
    nationalite VARCHAR(50),
    PRIMARY KEY (matricule)

```

```

);
create table livre (
    code INT ,
    nbpages INT,
    ISBN VARCHAR(50),
    titre VARCHAR(70) NOT NULL,
    date_sortie DATE,
    matricule_aut INT NOT NULL,
    nom_maison_edition VARCHAR(50),
    PRIMARY key (code),
    FOREIGN key (code) REFERENCES oeuvre(code) ON DELETE CASCADE ,
    FOREIGN key (nom_maison_edition) REFERENCES
editeur(nom_maison_edition) ON DELETE CASCADE ,
    FOREIGN key (matricule_aut) REFERENCES auteur(matricule)
);
create table ARTISTE (
    MATRICULE INT ,
    NOM VARCHAR(50),
    PRENOM VARCHAR(50),
    DATE_NAISSANCE DATE,
    NATIONALITE VARCHAR(50),
    PRIMARY KEY (MATRICULE)
);
create table Film (
    CODE_F INT,
    titre VARCHAR(70),
    annee VARCHAR(50),
    MATRICULE_REAL INT,
    PRIMARY KEY (CODE_F),
    FOREIGN key (MATRICULE_REAL) REFERENCES ARTISTE (MATRICULE)
);

CREATE TABLE film_album(CODE_F INT,
                           codealb INT,
                           PRIMARY KEY (CODE_F,codealb),
                           FOREIGN KEY (codealb) REFERENCES
album(codealb),
                           FOREIGN key (CODE_F) REFERENCES Film(CODE_F));

create table film_adapte_livre (
    CODE_F int,
    codeL INT ,
    PRIMARY KEY (codeL),

```

```

        FOREIGN key (CODE_F) REFERENCES Film(CODE_F),
        FOREIGN key (codeL) REFERENCES livre (CODE)

);

create table Rolee (
    ROLEE_NAME VARCHAR(50),
    CODE_F_R int ,
    MATRICULE_ACT INT ,
    PRIMARY KEY (ROLEE_NAME, CODE_F_R, MATRICULE_ACT),
    FOREIGN key (CODE_F_R) REFERENCES Film(CODE_F),
    FOREIGN key (MATRICULE_ACT) REFERENCES ARTISTE (MATRICULE)
);

create table adherent (
    matricule_adh INT PRIMARY KEY NOT NULL,
    email_adh VARCHAR(50) NOT NULL,
    phone_adh VARCHAR(50),
    NATIONALITE VARCHAR(50),
    nom_adh VARCHAR(50),
    prenom_adh VARCHAR(50),
    date_naiss_ad DATE,
    adress_adh VARCHAR(50),
    date_Debadh DATE,
    date_Finadh DATE
);

CREATE TABLE Emprunt (
    code INT,
    date_debut date NOT NULL ,
    matricule_adh int,
    date_fin date,
    PRIMARY KEY (code ,matricule_adh, date_debut),
    FOREIGN KEY (code) REFERENCES oeuvre (code),
    FOREIGN KEY (matricule_adh) REFERENCES
adherent(matricule_adh)
);

```

Data Source:

https://mockaroo.com/?fbclid=IwAR3ijLKhmQX_05U4ibta_1u2nN42sS72rxp9YD1fozl6KSjGMgQbX4yK018

REEMPLISSAGE DES TABLES

Voir les fichiers ci-joints

PARTIE INTERROGATION :

Khouloud SAID

1/ Donner le nombre des oeuvres pour chaque rayon

```
select r.num_rayon, count(*) "nombre oeuvres" from rayon r , oeuvre o, etagere e
      where e.num_etagere = o.id_etagere and r.num_rayon = e.num_rayon
      group by r.num_rayon;
```

NUM_RAYON	nombre oeuvres
1	18
2	19
5	11
4	18
3	10

|

2/ afficher code film, titre_film ,titre_album, nom_musicien qui ont adapté la musique d'un album solo

```
select f.code_f , album.titre , f.titre , musicien.prenom
from album_solo a , film f , film_album fl , album , musicien
where a.codealb=fl.codealb and f.code_f=fl.code_f and album.codealb=a.codealb and musicien.matricule_mus=a.matricule_mus
```

CODE_F	TITRE	TITRE	PRENOM
21	Konklux	Don t Look Up	Maffulli
41	Konklux	Piege de cristal	Maffulli
22	Tresom	Bohemian Rhapsody	Herrieven
42	Tresom	Une journee en enfer	Herrieven
23	Viva	Dachra	Fearenside
43	Viva	48 minutes pour vivre	Fearenside
24	Quo Lux	Special Correspondents	Salazar
44	Quo Lux	Pulp Fiction	Salazar
25	Y-Solowarm	Annie Hall	MacKartan
45	Y-Solowarm	Broken Arrow	MacKartan
26	Aerified	Brazil	Biodex
46	Aerified	Volte-Face	Biodex
27	Otcom	Underground	Home Ing
47	Otcom	Titanic	Home Ing
28	Namfix	Metropolis	Maffulli
48	Namfix	Mission Impossible 2	Maffulli
29	Cardify	Impitoyable	Lauri
49	Cardify	Mission Impossible	Lauri
30	Zontrax	Vertigo	Mat Lam Tam
50	Zontrax	Edward aux mains d'argent	Mat Lam Tam

20 lignes sélectionnées.

3/ afficher par categorie tout les oeuvres presents dans les etageres du rayon numero '1'

```
select e.categorie , o.code
from etagere e, oeuvre o
where e.num_rayon =1 and o.id_etagere = e.num_etagere
```

CATEGORIE	CODE
War	3
Comedy	18
Drama	19
Comedy	25
Drama	28
Thriller	36
War	38
Crime	40
War	50
Action	52
Romance	54
Drama	55
Comedy	57
Action	59
War	62
Adventure	64
Drama	65
Drama	70

18 lignes sélectionnées.

4/ afficher les numéro d'étagères ,qui ont un nombre d'œuvres dépassant la moyenne des nombre d'œuvres et leurs catégorie ='DRAMA '

```
select num_etagere ,categorie
from etagere
where upper(categorie) ='DRAMA' and nbre_oeuvre >(select( avg (nbre_oeuvre)) from etagere );
```

NUM_ETAGERE	CATEGORIE
29	Drama
46	Drama
50	Drama
59	Drama
86	Drama
97	Drama

6 lignes sélectionnées.

5/ donner la moyenne des ages des adherents qui sont intressés par Comedy

```
select avg(to_char(sysdate,'yyyy')-to_char(date_naiss_ad,'yyyy')) la_moyenne_des_ages
from adherent a , emprunt e , oeuvre o , etagere ee
where a.matricule_adh = e.matricule_adh and e.code = o.code and ee.num_etagere = o.id_etagere and upper(ee.categorie)='COMEDY';
```

LA_MOYENNE_DES_AGES
32.1578947

6/ afficher l'historique d'emprunt de l'adherent ayant la matricule num 15

```
select a.matricule_adh ,e.code , e.date_debut , e.date_fin
from emprunt e, adherent a
where a.matricule_adh =15 and e.matricule_adh = a.matricule_adh;
```

MATRICULE_ADH	CODE	DATE_DEB	DATE_FIN
15	46	30-01-21	16-08-22
15	72	05-10-21	19-07-22
15	51	20-04-21	01-02-22
15	24	20-05-21	25-04-22

7/ supprimer les adherents dont leurs abonnements sont expirées (date_fin adh> date_aujourd'hui=20/06/2021)

avant delete

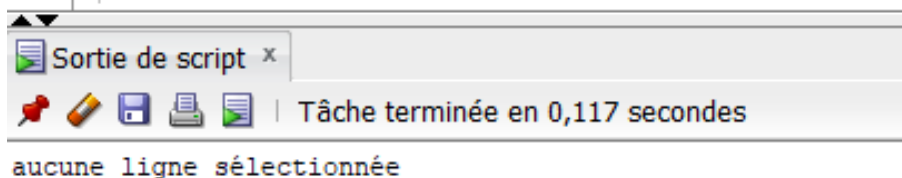
```
select * from adherent
where to_char(sysdate)>to_char(date_finadh);
```

MATRICULE_ADH	EMAIL_ADH	PHONE_ADH	NATIONALITE
7	bnucci6@buzzfeed.com	+1 655 675 7366	Canada
15	lceccolie@apache.org	+262 963 175 4198	Reunion
21	vdiegank@1688.com	+502 290 759 5367	Guatemala
29	pbaignards@tiny.cc	+351 398 816 1816	Portugal

apres delete

```
delete from adherent
where to_char(sysdate)>to_char(date_finadh);
```

```
select * from adherent
where to_char(sysdate)>to_char(date_finadh);
```



8/ supprimer les adherents de nationalite russe

```
delete from adherent
where upper(nationalite)='RUSSIA';
```

2 lignes supprimé.

9/ ajouter dans la table film_adapte_livre le film “48 minutes pour vivre”

Query Builder

```
insert into film_adapte_livre SELECT
f.code_f , l.code from livre l , film f
where f.titre='48 minutes pour vivre' and l.titre='48 minutes pour vivre';
```

FILM_ADAPTE_LIVRE x media x

ColonnesDonnéesModelContraintesDroitsStatistiques

Trier... | Filtre :

	CODE_F	CODEL
1	52	19
2	53	17
3	54	13
4	55	7
5	56	6
6	43	14

1 ligne inséré.