

SHMITT Jonathan | THIS Elliot



Entités :

Media : Après réflexion entre choisir titre uniquement et risquer de perdre les œuvres avec les mêmes noms, le couple titre & éditeur les éditeurs ayant souvent des spécialités (dans l'imprimerie etc), on a choisis de donner à chaque médias un identifiant unique et de l'utiliser comme clé primaire.

Film / Livre / Musique :

Album : Pour pouvoir associer un album et un titre de média, chaque album aura un nom unique.

Histo_nb_partage : L'entité permet un historique du nombre de "scrobble" par média et par artiste semaine après semaine. Les dates de début et de fin sont nécessaires pour pouvoir identifier les plages de dates.

Palmares : Sert à donner les auteurs les plus “scroblés” par semaine ou depuis la création du site

Relations :

Produis :

La relation permet de faire le lien entre une liste des auteurs et un média. Un artiste peut produire un ou plusieurs medias, un média peut être produis par un ou plusieurs artistes (ex: groupes de musiques).

La relation sera traduite en relationnel en une table contenant les deux clés primaires des tables qui lui sont liés. (relation naire)

Estprincipal :

Cette relation a pour but de déterminer l'auteur principal du média, la clé primaire de la table Artiste sera présente dans la table Média en tant que clé étrangère. Un artiste peut être l'artiste principal d'aucun media ou de plusieurs medias, un média est forcément produit par un artiste principal.

Cette relation sera traduite par l'ajout de la clé primaire de la table Artiste dans la table Média comme clé étrangère, elle sera nommée nom_artiste. (relation 0,n/1,1)

Partager :

La relation Partager vise a permettre l'historique de tout les partages qui ont été réalisé sur le site. Un utilisateur peut ne pas partager ou partager plusieurs medias et un média peut être partagé par un user ou plusieurs. Les médias qui ne sont pas partagés n'ont pas à apparaitre dans partage.

La relation sera traduite en relationnel en une table contenant les deux clés primaires des tables qui lui sont liés en plus des ses propres champs support et date_partage. (relation naire)

Fais parti : La relation a pour but de pouvoir joindre le nom d'un album au nom d'une musique. Une musique fait partie de un et un seul album
Un album comprends 1 (single) ou plusieurs musiques.

Cette relation sera traduite par l'ajout de la clé primaire de la table Album dans la table Musique comme clé étrangère.(relation 0,n/1,1)

Spécialisation :

Nous avons choisis de dupliquer les identifiants du sur-type Media dans les différents sous-types. Nous préférons avoir les 3 tables de sous médias rangées séparément.

Contraintes d'intégrités :

L'attribut Utilisateur.Mail, est unique.

Tout les attributs sauf Media.url et Partager.support et Musique.nom_album sont non nulls.

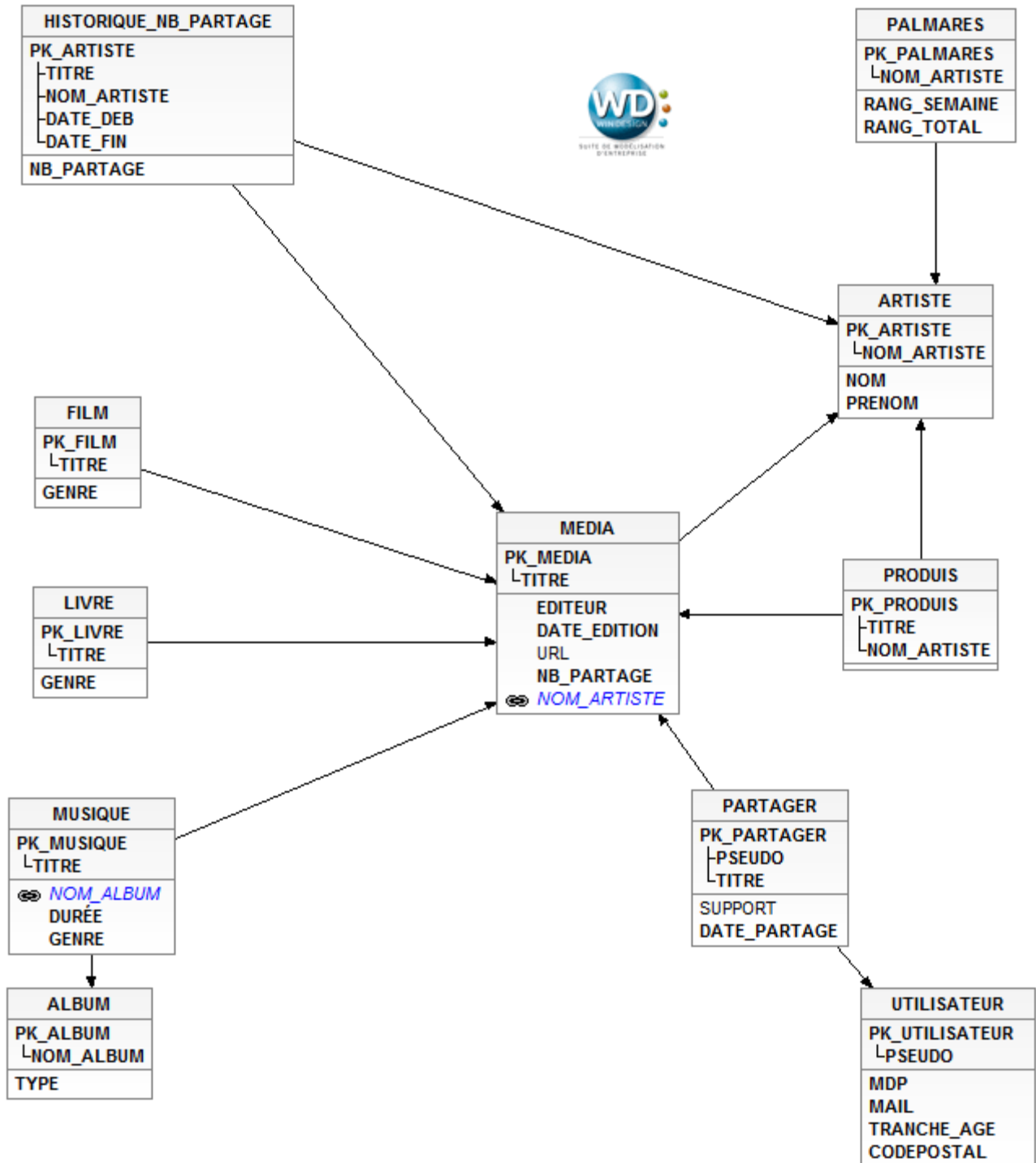
Partager.date_partage est <= SYSDATE.

La valeur par défaut de historique_nb_partages.nb_partage est 0.

Les valeurs de Palmares.Rang_Semaine et Palmares.Rang_total sont toujours >= 1.

Et toute les CI inhérentes aux PK & FK.

Justifications du modèle Relationnel :



Modèle Relationnel / logique (ligne par ligne) :

Tables :

ARTISTE(NOM_ARTISTE, NOM , PRENOM)

PRODUIT(TITRE, NOM_ARTISTE)

MEDIA(TITRE, EDATEUR, DATE_EDITION, URL, NB_PARTAGE_TOTAL, NOM_ARTISTE)

-FILM(TITRE, GENRE)

-LIVRE(TITRE, GENRE)

-MUSIQUE((TITRE, GENRE, DUREE, NOM_ALBUM)

ALBUM(NOM_ALBUM, TYPE)

PARTAGER(PSEUDO, TITRE, SUPPORT , DATE_PARTAGE)

UTILISATEUR(PSEUDO, MDP, MAIL, TRANCHE_AGE, CODEPOSTAL)

HISTO_NB_PARTAGE(TITRE, NOM_ARTISTE, DATE_DEB, DATE_FIN, NB_PARTAGE)

PALMARES(TITRE, NOM_ARTISTE, RANG_SEMAINE, RANG_TOTAL)

Références des foreign key:

MEDIA.TITRE Référence

PRODUIT.TITRE, FILM.TITRE, LIVRE.TITRE, MUSIQUE.TITRE, PARTAGER.TITRE,
HISTO_NB_PARTAGE.TITRE, PALMARES.TITRE .

ARTISTE.NOM_ARTISTE Référence

PRODUIT.NOM_ARTISTE, HISTO_NB_PARTAGE.NOM_ARTISTE, PALMARES.NOM_ARTISTE,
MEDIA.NOM_ARTISTE.

UTILISATEUR.PSEUDO Référence

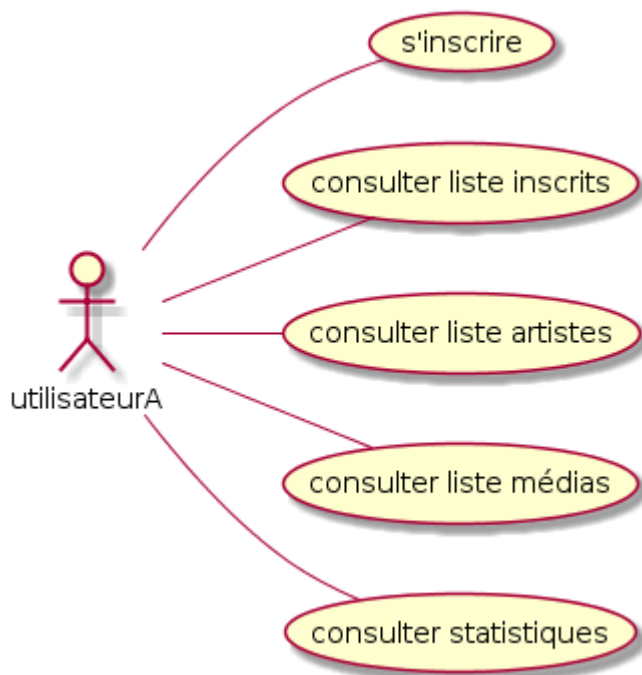
PARTAGER.PSEUDO.

ALBUM.NOM_ALBUM Référence

MUSIQUE.NOM_ALBUM .

Cas d'utilisations:

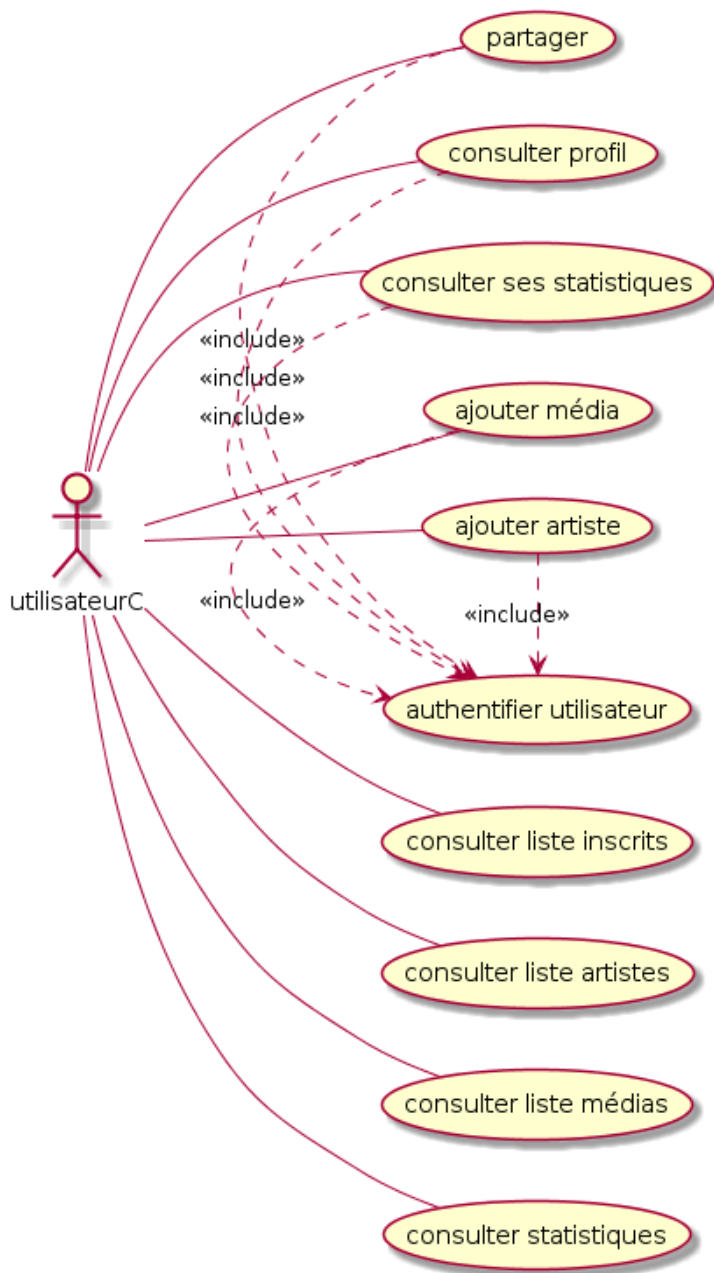
Cas d'utilisation pour un utilisateur anonyme :



Un utilisateur anonyme peut :

- s'inscrire sur le site (insert utilisateur dans la base)
- consulter la liste des inscrits (select dans utilisateurs)
- consulter la liste des artistes répertoriés sur le site (select dans artiste)
- consulter la liste des médias répertorié sur le site (select dans média)
- consulter les statistique (select table historique_nb_partage)

Cas d'utilisation pour un utilisateur connecté :



Un utilisateur connecté peut :

- partager un média (insert table partage, si le média est inconnu ajout du média, si artiste pas connu ajout de l'artiste)
- consulter son profil et celui des autres utilisateurs (select utilisateur)
- consulter ses statistiques (select table historique_nb_partage)
- ajouter un média (insert table média)
- ajouter un artistes (insert table artiste)
- consulter la liste des inscrits (select des utilisateurs)
- consulter la liste des artistes répertoriés sur le site (select artiste)
- consulter la liste des médias répertorié sur le site (select média)
- consulter les statistique (select table historique_nb_partage)

Scénarios :

Cas d'utilisation : authentifier utilisateur

Acteur(s) : personne

Scénario nominal :

1. la personne demande à se connecter au système
2. le système demande le login
3. la personne entre son login
4. le système demande le mot de passe
5. la personne entre son mot de passe
6. le système vérifie si le couple (login, mot de passe) existe dans la table User de la base de données
7. Si oui, le système valide la connexion
8. Sinon, le système refuse la connexion

Cas d'utilisation : partager

Acteur(s) : personne(connecté)

Scénario nominal :

1. la personne accède à la partie partage du site
2. la personne demande la liste des médias
3. le système renvoi la liste des médias
4. la personne choisit un média si il existe (l'artiste sera automatiquement sélectionné)
5. sinon elle l'ajoute

Cas d'utilisation : consulter liste inscrits

Acteur(s) : personne (connecté)

Scénario nominal :

1. la personne demande la liste des inscrits au système
2. le système cherche les informations et les affichent
3. la personne peut consulter la liste des inscrits

Justifications des prises de partis "pas de mauvaise solutions" :

Nous avons décidé de ranger les artistes ayant travaillé sur un média dans la table Produis pour garder une trace des artistes qui ne sont pas "principaux" et de leurs médias.

On à aussi choisi de garder un champ Nombre de partage (total) dans média, simplement pour qu'un utilisateur curieux puisse connaître la popularité de ce média précis.

Tout nos champs calculés seront fais a partir de vues stockées dans des variables :

Le script qui se lance tout les dimanches fera le compte des médias partagés durant toute la semaine et inscrira dans la table Historique_nb_partages le résultat de ses calculs; il associera au titre son auteur principal, les tranches de dates (Date D - 7 jours en deb et D en fin) pour enfin mettre une valeur dans nb_partage. Une autre vue se chargera de faire l'extraction de la valeur ainsi que la somme de toute les autres valeurs précédentes du média correspondant pour le palmarès par artiste. En ce qui concerne les données d'un utilisateurs, ses classements seront extrait a l'aide d'une vue sans stockage qui utilisera les infos de la table Partage.

Dictionnaire de données du modèle relationnel :

Nom attribut	Définition	Domaine	Atomique	Monovalué	Obligatoire	Calculé
Media.titre	Le titre du média	char(32)	Y	Y	Y	N
Media.editeur	L'éditeur du média	char(32)	Y	Y	Y	N
Media.date_edition	La date d'édition du média	Date	Y	Y	Y	N
Media.ULR	L'URL du média	char(128)	Y	Y	N	N
Media.nb_partage	Le nombre de partage du média	Number(10)	Y	Y	Y	Y
Media.nom_auteur	Le nom de l'auteur principal du média	char(32)	Y	Y	Y	N
Utilisateur.pseudo	Le pseudo de l'utilisateur	char(32)	Y	Y	Y	N
Utilisateur.mdp	Le mot de passe de l'utilisateur	char(64)	Y	Y	Y	N
Utilisateur.mail	Le mail de l'utilisateur	char(32)	Y	Y	Y	N
Utilisateur.tranche_age	La tranche d'âge de l'utilisateur	char(16)	N	Y	Y	N
Utilisateur.codepostal	Le code postal de l'utilisateur	Number(10)	Y	Y	Y	N
Partager.titre	Le titre du média partagé	char(32)	Y	Y	Y	N
Partager.pseudo	Le pseudo de l'utilisateur qui a partagé	char(32)	Y	Y	Y	N
Partager.support	Le support du média partagé	char(32)	Y	Y	N	N
Partager.date_partage	La date du partage	Date	Y	Y	Y	N
Produis.nom_artiste	Le nom des auteurs qui ont produit le média	char(32)	Y	Y	Y	N
Produis.titre	Le titre du média	char(32)	Y	Y	Y	N
Artiste.nom_artiste	Le nom d'artiste de l'artiste	char(32)	Y	Y	Y	N
Artiste.nom	Le nom civil de l'artiste	char(32)	Y	Y	Y	N
Artiste.prenom	Le prénom de l'artiste	char(32)	Y	Y	Y	N
Musique.titre	Le titre de la musique	char(32)	Y	Y	Y	N
Musique.nom_album	Le nom de l'album qui contient la musique (si la musique en fait partie d'un)	char(32)	Y	Y	N	N
Musique.durée	La durée de la musique	char(16)	Y	Y	Y	N
Musique.genre	Le genre de la musique	char(32)	Y	Y	Y	N
Album.nom_album	Le nom de l'album	char(32)	Y	Y	Y	N
Album.type	Le type de l'album	ENUM: «Single,Album,minialbum»	Y	Y	Y	N
Livre.titre	Le titre du livre	char(32)	Y	Y	Y	N
Livre.genre	Le genre du livre	char(32)	Y	Y	Y	N
Film.titre	Le titre du film	char(32)	Y	Y	Y	N
Film.genre	Le genre du film	char(32)	Y	Y	Y	N
Palmares.nom_artiste	Le nom de l'artiste	char(32)	Y	Y	Y	N

Palmares.rang_semaine	Le rang de l'artiste pour la semaine	Number(10)	Y	Y	Y	Y
Palmares.rang_total	Le rang de l'artiste pour la totalité	Number(10)	Y	Y	Y	Y
Historique [...].titre	Le titre du média scroblé	char(32)	Y	Y	Y	N
Historique [...].nom_artiste	Le nom de l'artiste principal	char(32)	Y	Y	Y	N
Historique [...].nb_partage	Le nombre de partage sur une semaine du media	Number(10)	Y	Y	Y	Y
Historique [...].date_deb	La date de début de comptabilisation	Date	Y	Y	Y	N
Historique [...].date_fin	La date de fin de comptabilisation	Date	Y	Y	Y	N

Esquisse du modèle utilisateur (IHM) :



Script de création de la base de donnée :

```
DROP TABLE MUSIQUE CASCADE CONSTRAINTS;
DROP TABLE LIVRE CASCADE CONSTRAINTS;
DROP TABLE FILM CASCADE CONSTRAINTS;
DROP TABLE MEDIA CASCADE CONSTRAINTS;
DROP TABLE PRODUIS CASCADE CONSTRAINTS;
DROP TABLE ARTISTE CASCADE CONSTRAINTS;
DROP TABLE PALMARES CASCADE CONSTRAINTS;
DROP TABLE HISTORIQUE_NB_PARTAGE CASCADE CONSTRAINTS;
DROP TABLE ALBUM CASCADE CONSTRAINTS;
DROP TABLE PARTAGER CASCADE CONSTRAINTS;
DROP TABLE UTILISATEUR CASCADE CONSTRAINTS;
```

```
CREATE TABLE MUSIQUE (
    TITRE CHAR(32) NOT NULL,
    NOM_ALBUM CHAR(32) NOT NULL,
    DURÉE CHAR(16) NOT NULL,
    GENRE CHAR(32) NOT NULL,
    CONSTRAINT PK_MUSIQUE PRIMARY KEY (TITRE) );
```

```
CREATE TABLE LIVRE (
    GENRE CHAR(32) NOT NULL,
    TITRE CHAR(32) NOT NULL,
    CONSTRAINT PK_LIVRE PRIMARY KEY (TITRE) );
```

```
CREATE TABLE FILM (
    TITRE CHAR(32) NOT NULL,
    GENRE CHAR(32) NOT NULL,
    CONSTRAINT PK_FILM PRIMARY KEY (TITRE) );
```

```
CREATE TABLE MEDIA(
    TITRE CHAR(32) NOT NULL,
    EDATEUR CHAR(32) NOT NULL,
    DATE_EDITION DATE NOT NULL,
    URL CHAR(128) NULL,
    NB_PARTAGE NUMBER(10,2) NOT NULL,
    NOM_ARTISTE CHAR(32) NOT NULL,
    CONSTRAINT PK_MEDIA PRIMARY KEY (TITRE) );
```

```
CREATE TABLE PRODUIS(
    NOM_ARTISTE CHAR(32) NOT NULL,
    TITRE CHAR(32) NOT NULL,
    CONSTRAINT PK_PRODUIS PRIMARY KEY (TITRE, NOM_ARTISTE) );
```

```
CREATE TABLE ARTISTE (
    NOM_ARTISTE CHAR(32) NOT NULL,
    NOM CHAR(32) NOT NULL,
    PRENOM CHAR(32) NOT NULL,
    CONSTRAINT PK_ARTISTE PRIMARY KEY (NOM_ARTISTE) );
```

```
CREATE TABLE PALMARES (
    NOM_ARTISTE CHAR(32) NOT NULL,
    RANG_SEMAINE NUMBER(10,2) NOT NULL,
    RANG_TOTAL NUMBER(10,2) NOT NULL,
    CONSTRAINT PK_PALMARES PRIMARY KEY (NOM_ARTISTE) );
```

```
CREATE TABLE HISTORIQUE_NB_PARTAGE (
    DATE_DEB DATE NOT NULL,
    DATE_FIN DATE NOT NULL,
    NB_PARTAGE NUMBER(10,2) NOT NULL,
    NOM_ARTISTE CHAR(32) NOT NULL,
    TITRE CHAR(32) NOT NULL,
    CONSTRAINT PK_ARTISTE PRIMARY KEY (TITRE, NOM_ARTISTE, DATE_DEB, DATE_FIN) );
```

```
CREATE TABLE ALBUM(
    NOM_ALBUM CHAR(32) NOT NULL,
    TYPE ENUM('Single','Album','mini_album') NOT NULL,
    CONSTRAINT PK_ALBUM PRIMARY KEY (NOM_ALBUM) );
```

```
CREATE TABLE PARTAGER(
    TITRE CHAR(32) NOT NULL,
    PSEUDO CHAR(32) NOT NULL,
    SUPPORT CHAR(32) NULL,
    DATE_PARTAGE DATE NOT NULL,
    CONSTRAINT PK_PARTAGER PRIMARY KEY (PSEUDO, TITRE) );
```

```
CREATE TABLE UTILISATEUR(  
  PSEUDO CHAR(32) NOT NULL,  
  MDP CHAR(64) NOT NULL,  
  MAIL CHAR(32) NOT NULL,  
  TRANCHE_AGE CHAR(16) NOT NULL,  
  CODEPOSTAL NUMBER(10,2) NOT NULL,  
  CONSTRAINT PK_UTILISATEUR PRIMARY KEY (PSEUDO) );
```

```
ALTER TABLE MUSIQUE ADD (  
  CONSTRAINT FK_MUSIQUE_MEDIA  
  FOREIGN KEY (TITRE)  
  REFERENCES MEDIA (TITRE)) ;
```

```
ALTER TABLE MUSIQUE ADD (  
  CONSTRAINT FK_MUSIQUE_ALBUM  
  FOREIGN KEY (NOM_ALBUM)  
  REFERENCES ALBUM (NOM_ALBUM)) ;
```

```
ALTER TABLE LIVRE ADD (  
  CONSTRAINT FK_LIVRE_MEDIA  
  FOREIGN KEY (TITRE)  
  REFERENCES MEDIA (TITRE)) ;
```

```
ALTER TABLE FILM ADD (  
  CONSTRAINT FK_FILM_MEDIA  
  FOREIGN KEY (TITRE)  
  REFERENCES MEDIA (TITRE)) ;
```

```
ALTER TABLE MEDIA ADD (  
  CONSTRAINT FK_MEDIA_ARTISTE  
  FOREIGN KEY (NOM_ARTISTE)  
  REFERENCES ARTISTE (NOM_ARTISTE)) ;
```

```
ALTER TABLE PRODUIS ADD (  
  CONSTRAINT FK_PRODUIS_ARTISTE  
  FOREIGN KEY (NOM_ARTISTE)  
  REFERENCES ARTISTE (NOM_ARTISTE)) ;
```

```
ALTER TABLE PRODUIS ADD (  
  CONSTRAINT FK_PRODUIS_MEDIA  
  FOREIGN KEY (TITRE)  
  REFERENCES MEDIA (TITRE)) ;
```

```
ALTER TABLE PALMARES ADD (  
  CONSTRAINT FK_PALMARES_ARTISTE  
  FOREIGN KEY (NOM_ARTISTE)  
  REFERENCES ARTISTE (NOM_ARTISTE)) ;
```

```
ALTER TABLE HISTORIQUE_NB_PARTAGE ADD (  
  CONSTRAINT FK_HISTORIQUE_NB_PARTAGE_ARTIS  
  FOREIGN KEY (NOM_ARTISTE)  
  REFERENCES ARTISTE (NOM_ARTISTE)) ;
```

```
ALTER TABLE HISTORIQUE_NB_PARTAGE ADD (  
  CONSTRAINT FK_HISTORIQUE_NB_PARTAGE_MEDIA  
  FOREIGN KEY (TITRE)  
  REFERENCES MEDIA (TITRE)) ;
```

```
ALTER TABLE PARTAGER ADD (  
  CONSTRAINT FK_PARTAGER_MEDIA  
  FOREIGN KEY (TITRE)  
  REFERENCES MEDIA (TITRE)) ;
```

```
ALTER TABLE PARTAGER ADD (  
  CONSTRAINT FK_PARTAGER_UTILISATEUR  
  FOREIGN KEY (PSEUDO)  
  REFERENCES UTILISATEUR (PSEUDO)) ;
```