

## PTI #01 : Administration d'une base de données

## Compétences abordées

C21 : Installer et configurer un microordinateur

C22 : Installer et configurer un réseau

C37 : Administrer une base de données

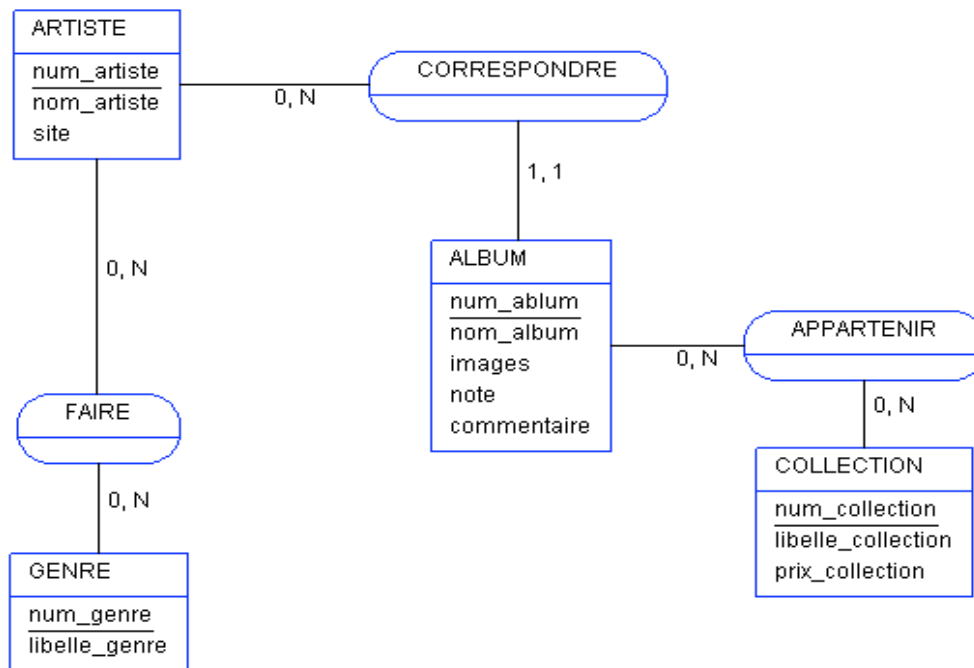
## Contexte

Un magasin de musique veut gérer ses produits grâce à une base de données MySQL.

## Objectifs

Crée une base de données qui permet de gérer tous les albums et en limiter l'accès.

## Modèle Conceptuel des Données



## Outils utilisés

- Système d'exploitation :
  - Linux Fedora Core 5
  - Mysql
  - Apache
  - PHP
- Caractéristiques matérielles :
  - Micro-ordinateur

Schéma relationnel :

```
ARTISTE (num_artiste, nom_artiste, site)
GENRE (num_genre, libelle_genre)
COLLECTION (num_collection, libelle_collection, prix_collection)
ALBUM (num_album, nom_album, images, note, commentaire, #num_artiste,
#num_collection)
FAIRE (#num_artiste, #num_genre)
```

Création de la base de données :

```
CREATE DATABASE music ;
```

Création de l'administrateur de la base :

```
GRANT ALL PRIVILEGES ON `music` . * TO 'admin'@'%' WITH GRANT OPTION ;
```

Création des tables

```
CREATE TABLE artiste (
num_artiste INT NOT NULL AUTO_INCREMENT,
nom_artiste VARCHAR(255) NOT NULL,
site VARCHAR(255) NOT NULL,
PRIMARY KEY (num_artiste));
```

```
CREATE TABLE genre (
num_genre INT NOT NULL AUTO_INCREMENT,
libelle_genre VARCHAR(255) NOT NULL,
PRIMARY KEY (num_genre));
```

```
CREATE TABLE collection (
num_collection INT NOT NULL AUTO_INCREMENT,
libelle_collection VARCHAR(255) NOT NULL,
prix_collection INT NOT NULL,
PRIMARY KEY (num_collection));
```

```
CREATE TABLE album (
num_album INT NOT NULL AUTO_INCREMENT,
nom_album VARCHAR(255) NOT NULL,
images VARCHAR(255) NOT NULL,
note INT NOT NULL,
commentaire VARCHAR(255) NOT NULL,
num_artiste INT NOT NULL,
num_collection INT NOT NULL,
PRIMARY KEY (num_album),
FOREIGN KEY (num_artiste) REFERENCES artiste(num_artiste),
FOREIGN KEY (num_collection) REFERENCES collection(num_collection));
```

```
CREATE TABLE faire (  
  num_artiste INT NOT NULL,  
  num_genre INT NOT NULL,  
  PRIMARY KEY (num_artiste,num_genre),  
  FOREIGN KEY (num_artiste) REFERENCES artiste(num_artiste),  
  FOREIGN KEY (num_genre) REFERENCES genre(num_genre));
```

Création de la vue permettant aux simples utilisateurs de voir que les albums et les nom des artistes

```
CREATE VIEW touslesalbums AS(  
  SELECT nom_album, nom_artiste  
  FROM album, artiste  
  WHERE artiste.num_artiste=album.num_artiste);
```

Création de l'utilisateur de la vue

```
CREATE USER user IDENTIFIED BY '123456';
```

Le droit sur la vue a l'utilisateur "user"

```
GRANT select ON test TO user;
```