

UNIVERSITE DE STRASBOURG



UFR de mathématique et d'informatique

**Projet de Base de Données
Licence 3 en Informatique**

Sujet : *Réalisation d'une base de données
pour un site ou une application de gestion de
collections d'objets culturels*

Réalisé par : Baye Lahad MBACKE

Année universitaire 2022-2023

SOMMAIRE

INTRODUCTION.....	3
Modélisation : Entité Association	4
Relation entre les entités	6
Les contraintes d'intégrité	6
Quelques Test sur les contraintes	7
Le modèle logique relationnel	8
Outils utilisés.....	9

TABLES DES FIGURES

<u>Figure 1 : Modèle Entité Association</u>	<u>.4</u>
<u>Figure 2 : test contraint Login</u>	<u>7</u>
<u>Figure 3 : test contraint Password</u>	<u>8</u>
<u>Figure 4 : test contraint sur la note</u>	<u>8</u>

Introduction

Le but de ce projet est de développer une base de données pour un site ou une application de gestion de collections d'objets culturels dans laquelle les utilisateurs pourront exposer leurs objets par exemple livre, film, jeux, etc.

On traitera ici la partie 1 du projet en faisant la modélisation (modèle Entité Association), le model relationnel et aussi indiquer les contraintes. Le modèle sera ensuite concrétisé par sa mise en place dans une base de données Oracle

I. Modélisation : Entité Association

On représente ici le modèle entité-association de la base de données pour les collections d'objets représentant les besoins fonctionnels.

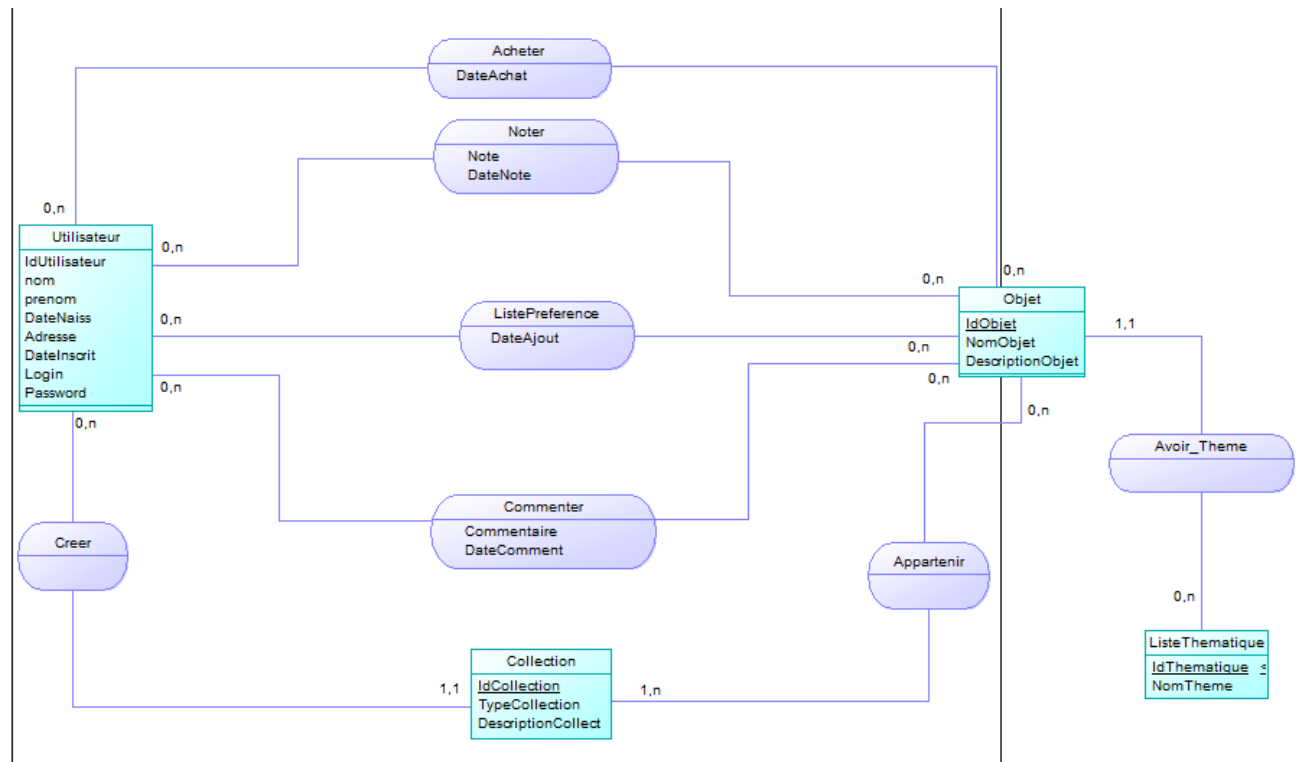


Figure 1 : Modèle Entité Association

Entité utilisateur

Cette entité contiendra l'ensemble des utilisateurs du site.

Les champs que je considère utile pour cette table sont :

- ✚ **IdUtilisaateur** : la clé primaire, chaque utilisateur pourra donc être identifié avec un numéro unique car chaque utilisateur doit pouvoir identifier.
- ✚ **Nom** : la clé primaire, chaque utilisateur pourra donc être identifié avec un numéro unique.
- ✚ **Prénom** : Le prénom de l'utilisateur
- ✚ **DateNais** : c'est la date de naissance de l'utilisateur
- ✚ **Adresse** : c'est l'adresse de l'utilisateur
- ✚ **DateInscrit** : c'est la date à laquelle l'utilisateur s'est inscrit dans le site
- ✚ **Login** : chaque utilisateur pourra être identifié avec login unique.
- ✚ **Password** : le mot de passe que l'utilisateur doit entrer pour se connecter

Entité Objet :

Cette entité contiendra l'ensemble des objets du site.

- + **IdObjet** : la clé primaire, chaque objet pourra donc être identifié avec un numéro unique.
- + **NomObjet** : c'est le nom d'un objet
- + **TypeObjet** : c'est le type d'un objet
- + **DescriptionObjet** : la description que possède un objet

Entité Collection :

Cette entité contiendra l'ensemble des objets du site.

- + **IdCollection** : la clé primaire, chaque collection pourra donc être identifié avec un numéro unique.
- + **TypeCollection** : c'est le type d'une collection c'est-à-dire le type d'objet qu'elle doit contenir
- + **DescriptionCollect** : la description que possède une collection

Entité ListThématique :

Cette entité contiendra l'ensemble des thèmes que peut avoir un objet culturel du site.

- + **IdThématique** : la clé primaire, chaque thème pourra donc être identifié avec un numéro unique.
- + **NomThématique** : c'est le nom d'un thème c'est-à-dire le thème d'un objet

II Relation entre les entités :

Utilisateur et collection : un utilisateur peut créer une collection. Il peut créer de 0 à n (plusieurs) collection, et une collection est créée par un seul et unique utilisateur. Ces deux entités sont liées par une association appelé *créer*

Objet et Collection : un objet peut ne pas appartenir (0) à une collection et aussi il peut appartenir à plusieurs (n) collections. Une collection doit avoir au moins un objet donc on a créé une association nomme *Appartenir* qui contiendra l'identifiant des tables (IdCollection et IdObjet)

Utilisateur et Objet : un utilisateur peut commenter de 0 à n (plusieurs) objets à une date donnée, un objet peut être commenté par 0 ou n (plusieurs) utilisateur donc on a créé une association *commentaire* qui contient (IdObjet, IdUtilisateur, Commentaire, DateComment), il peut aussi noter 0 à n (plusieurs) objets à une date donnée, un objet peut être noté par 0 ou n (plusieurs) utilisateur donc on a créé une association nommée *noter* qui contient (IdObjet, IdUtilisateur, Note, DateNote) et il peut aussi ajouter 0 à n (plusieurs) objets dans sa liste de préférence à une date donnée, un objet peut être ajouté à 0 ou n (plusieurs) fois dans la liste de préférence des utilisateurs donc on a créé une association appelé *ListePreference* qui contient (IdObjet, IdUtilisateur et DateAjout)

Objet et ListeThematique : Un objet a 1 et 1 seul thème et un thème se trouvant dans la ListThematique peut concerner 0 ou n (plusieurs) objets

III Les contraintes d'intégrité :

Pour l'utilisateur :

- ✚ Toutes les informations concernant un utilisateur ne doivent pas être non vide
- ✚ Le mot de passe doit être uniquement en lettre majuscule ou minuscule ou chiffre ou '_'
- ✚ Le login doit être composé de la première lettre du prénom et des 7 premières lettres du nom (en minuscules) suivies de deux chiffres et doit être non vide
- ✚ L'identifiant (clé primaire) est non vide
- ✚ Le login d'un utilisateur doit être unique et doit être non vide

Pour l'objet :

- ✚ L'identifiant (clé primaire) est non vide

- ✚ Le nom et la description d'un objet ne doivent être non vide car c'est avec la description d'un objet qu'on peut savoir dans quelle collection doit-on le placer

Pour la collection :

- ✚ La description d'une collection ne doit être non vide
- ✚ L'identifiant (clé primaire) est non vide
- ✚ Le type d'une collection doit être forcément fourni et sa description aussi (non vide) car c'est avec la description d'une collection qu'on peut savoir les objets qu'elle doit contenir
- ✚ Un objet dont le type est différent à celui de la collection ne doit pas être accepté dans la collection.

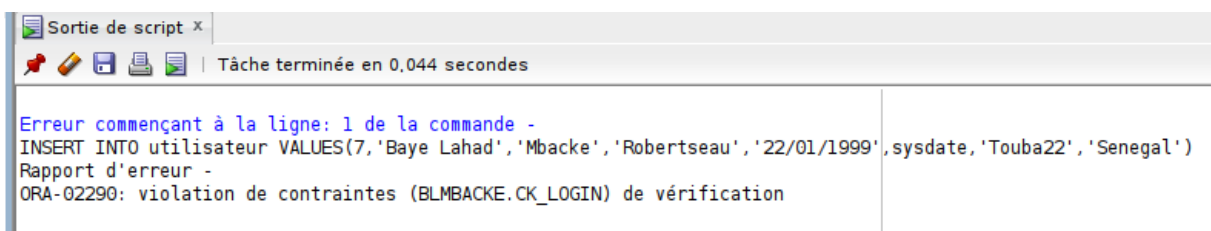
Exemple : un objet de type A ne doit pas être dans une collection de type B mais plutôt dans une collection de type A

Pour la note et le commentaire :

- ✚ L'attribut Note ne doit pas avoir une valeur inférieure à 1 ni supérieur à 20 et doit être aussi non vide
- ✚ Une note ou en commentaire doit être faite à une date donnée donc si leurs champs ne doivent pas être non vide
- ✚ Un utilisateur ne peut noter ou commenter un objet une seule fois

IV Quelques Test sur les contraintes :

Pour le Login :



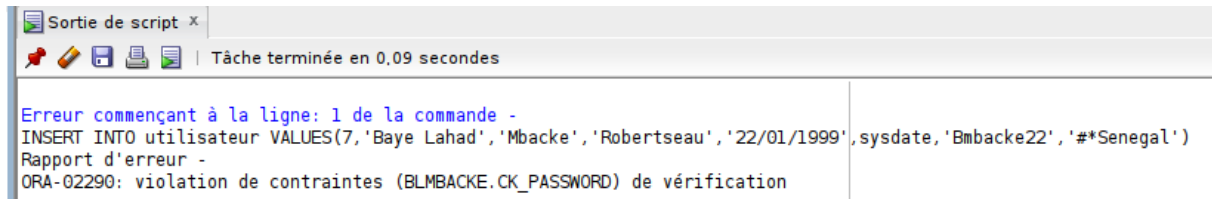
Sortie de script x

Tâche terminée en 0,044 secondes

```
Erreur commençant à la ligne: 1 de la commande -  
INSERT INTO utilisateur VALUES(7,'Baye Lahad','Mbacke','Robertseau','22/01/1999',sysdate,'Touba22','Senegal')  
Rapport d'erreur -  
ORA-02290: violation de contraintes (BLMBACKE.CK_LOGIN) de vérification
```

Figure 2 : test contrainte Login

Pour le password :

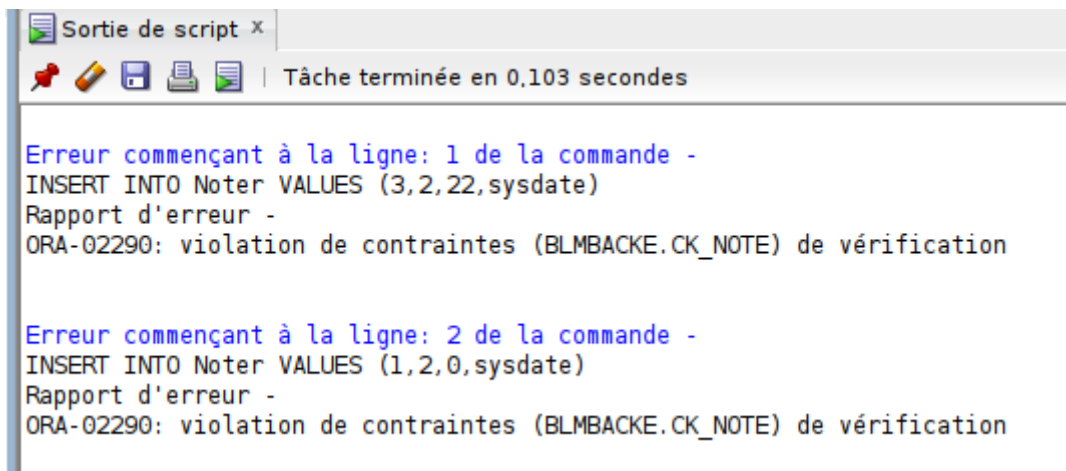


```
Sortie de script x
Tâche terminée en 0,09 secondes

Erreur commençant à la ligne: 1 de la commande -
INSERT INTO utilisateur VALUES(7,'Baye Lahad','Mbacke','Robertseau','22/01/1999',sysdate,'Bmbacke22','*Senegal')
Rapport d'erreur -
ORA-02290: violation de contraintes (BLMBACKE.CK_PASSWORD) de vérification
```

Figure 3 : test contraint Password

Pour la note :



```
Sortie de script x
Tâche terminée en 0,103 secondes

Erreur commençant à la ligne: 1 de la commande -
INSERT INTO Noter VALUES (3,2,22,sysdate)
Rapport d'erreur -
ORA-02290: violation de contraintes (BLMBACKE.CK_NOTE) de vérification

Erreur commençant à la ligne: 2 de la commande -
INSERT INTO Noter VALUES (1,2,0,sysdate)
Rapport d'erreur -
ORA-02290: violation de contraintes (BLMBACKE.CK_NOTE) de vérification
```

Figure 4 : test contraint sur la note

V Le modèle logique relationnel :

On a construit le modèle logique relationnel de la base de données en se basant sur le modèle entité association

Utilisateur (IdUtilisateur : int, nom : String, prenom : String, DateNais : Date, Adresse : String, DateInscrit Date, Login : String, Password : String)

Objet(IdObjet :int,NomObjet:String,DescriptionObjet:String,#IdThematique: int)

Collections(IdCollection:int,TypeCollection :String,DescriptionCollect:String,
#IdUilsateur : int)

ListeThematique (IdThematique : int, NomTheme : String)

Commenter(#IdUtilisateur:int,#IdObjet:int,commentaire:String, DateComment : Date)

Noter (#IdUtilisateur : int, #IdObjet : int, note : int, DateNote : Date)

Appartenir (#IdCollection : int, #IdObjet : int)

Acheter (#IdUtilisateur : int, #IdObjet : int, DateAchat : Date)

ListePreference (#IdUtilisateur : int, #IdObjet : int, DateAjout : Date)

Les attributs soulignés sont *les clés primaires*, dans certains table la clé primaire est compose

Exemple : Dans la table **commenter** la clé primaire est forme par les 2 clés étrangers (#IdUtilisateur,#IdObjet).

Les attributs précédés d'un symbole # sont des *clés étrangères*

VI Outils utilisés

On a utilisé le logiciel POWER AMC pour faire la modélisation Entité Association de la base, et aussi pour tester nos scripts SQL on a utilisé **Oracle SQL Developer**

NB :

Pour les scripts, on a créé 4 scripts :

- ✓ **Suppression_Tables.sql** : ce script permet de supprimer toutes les tables
- ✓ **Creation_Tables.sql** : ce script permet de faire la création des tables ainsi l'ajout des contraintes
- ✓ **Insertion_Des_Donnees.sql** : ce script permet de faire l'insertion des données dans les tables
- ✓ **ScriptGeneral.sql** : ce script permet d'exécuter les 3 scripts qu'on cite en dessus