

TP1 : Base des données

Création de la base des données:

```
2      use Master;
3      CREATE TABLE Auteur (
4          codeAuteur NUMBER PRIMARY KEY,
5          nom VARCHAR2(50),
6          prenom VARCHAR2(50),
7          nationalite VARCHAR2(50) );
8
9      CREATE TABLE Livre (
10         codeLivre NUMBER PRIMARY KEY,
11         titre VARCHAR2(100),
12         nbPages NUMBER,
13         annee NUMBER,
14         prix NUMBER );
15
16      CREATE TABLE Membre (
17         codeMembre NUMBER PRIMARY KEY,
18         nom VARCHAR2(50),
19         prenom VARCHAR2(50),
20         dateInscription DATE );
21
22      CREATE TABLE Possède (
23         codeLivre NUMBER,
24         codeAuteur NUMBER,
25         PRIMARY KEY (codeLivre, codeAuteur),
26         FOREIGN KEY (codeLivre) REFERENCES Livre(codeLivre),
```

```
9      CREATE TABLE Prêt (
10         codePrêt NUMBER PRIMARY KEY,
11         codeMembre NUMBER,
12         codeLivre NUMBER,
13         datePrêt DATE,
14         dateRetour DATE,
15         FOREIGN KEY fk_codemembre(codeMembre) REFERENCES Membre(codeMembre),
16         FOREIGN KEY fk_codelivre(codeLivre) REFERENCES Livre(codeLivre) );
17
```

1 : Connexion à oracle:

```
CREATE USER adia IDENTIFIED BY 1234;
GRANT CREATE SESSION, CREATE TABLE TO adia;
CONNECT adia/1234;
SELECT object_name, object_type FROM user_objects;
ALTER TABLE Membre ADD email VARCHAR2(100);
ALTER TABLE Membre ADD adresse VARCHAR2(255);
ALTER TABLE Prêt DROP CONSTRAINT fk_codemembre;
ALTER TABLE Prêt ADD CONSTRAINT fk_Nmembre FOREIGN KEY (Nmembre) REFERENCES Membre(codeMembre);
ALTER TABLE Prêt ADD CONSTRAINT fk_Nlivre FOREIGN KEY (Nlivre) REFERENCES Livre(codeLivre);
ALTER TABLE Prêt ADD CONSTRAINT pk_Prêt PRIMARY KEY (codePrêt);
ALTER TABLE Livre ADD Nauteur NUMBER;
ALTER TABLE Livre ADD CONSTRAINT fk_Nauteur FOREIGN KEY (Nauteur) REFERENCES Auteur(codeAuteur);
DROP TABLE Possède;
ALTER TABLE Livre ADD CONSTRAINT ck_titre CHECK (titre IN ('Valeur1', 'Valeur2', 'Valeur3'));
ALTER TABLE Livre ADD langue VARCHAR2(50);
ALTER TABLE Livre ADD NbreExemplaires NUMBER;
ALTER TABLE Membre ADD CONSTRAINT uq_nom_prenom UNIQUE (nom, prenom);
CREATE INDEX idx_pk_Prêt ON Prêt(codePrêt);
CREATE INDEX idx_fk_Nmembre ON Prêt(Nmembre);
CREATE INDEX idx_nom_auteur ON Auteur(nom);
CREATE INDEX idx_nom_prenom_membre ON Membre(nom, prenom);
CREATE INDEX idx_titre_livre ON Livre(titre);
SELECT index_name FROM user_indexes;
DROP INDEX idx_nom_auteur;
ALTER TABLE Prêt ADD CONSTRAINT ck_dateRetour CHECK (dateRetour >= datePrêt);
SELECT constraint_name, constraint_type, table_name FROM user_constraints;
```

2. Créer un utilisateur nommé adia :

```
CREATE USER adia IDENTIFIED BY 1234;
```

3. Donner à cet utilisateur les droits pour se connecter à la base (create session) et créer des objets de BD (create table) :

```
GRANT CREATE SESSION, CREATE TABLE TO adia;
```

4. Se connecter à Oracle en utilisant le login de l'utilisateur créé :

```
CONNECT adia/ 1234;
```

5. Préciser les clés primaires et les clés étrangères des relations ci-dessus :

6. Créer les tables de cette base de données :

```
1  Créer la table Auteur :
2      CREATE TABLE Auteur (
3          codeAuteur NUMBER PRIMARY KEY,
4          nom VARCHAR2(50),
5          prenom VARCHAR2(50),
6          nationalite VARCHAR2(50) );
7  Créer la table Livre :
8      CREATE TABLE Livre (
9          codeLivre NUMBER PRIMARY KEY,
10         titre VARCHAR2(100),
11         nbPages NUMBER,
12         annee NUMBER,
13         prix NUMBER );
14  Créer la table Membre :
15      CREATE TABLE Membre (
16          codeMembre NUMBER PRIMARY KEY,
17          nom VARCHAR2(50),
18          prenom VARCHAR2(50),
19          dateInscription DATE );
20  Créer la table Possède (relation entre Livre et Auteur) :
21      CREATE TABLE Possède (
22          codeLivre NUMBER,
23          codeAuteur NUMBER,
24          PRIMARY KEY (codeLivre, codeAuteur),
25          FOREIGN KEY fk1_codelivre (codeLivre) REFERENCES Livre(codeLivre),
26          FOREIGN KEY fk_codeauteur(codeAuteur) REFERENCES Auteur(codeAuteur) );
```

```
Créer la table Prêt :
CREATE TABLE Prêt (
    codePret NUMBER PRIMARY KEY,
    codeMembre NUMBER,
    codeLivre NUMBER,
    datePret DATE,
    dateRetour DATE,
    FOREIGN KEY fk_codemembre (codeMembre) REFERENCES Membre(codeMembre),
    FOREIGN KEY fk2_codelivre (codeLivre) REFERENCES Livre(codeLivre) );
```

7. Utiliser la vue user_objects pour vérifier la présence des objets créés :

```
SELECT object_name, object_type FROM user_objects;
```

8. Ajouter le champ email à la table membre :

```
ALTER TABLE membre ADD email VARCHAR2(100);
```

9. Ajouter la colonne adresse à la table membre :

```
ALTER TABLE membre ADD adresse VARCHAR2(255);
```

10. Supprimer les contraintes de la table prêt :

```
ALTER TABLE prêt DROP CONSTRAINT fk_codemembre ;
```

11. Modifier la table Prêt en y ajoutant les contraintes suivantes :

```
ALTER TABLE Prêt ADD CONSTRAINT fk_Nmembre FOREIGN KEY (Nmembre)  
REFERENCES Membre(Nmembre);
```

```
ALTER TABLE Prêt ADD CONSTRAINT fk_Nlivre FOREIGN KEY (Nlivre)  
REFERENCES Livre(Nlivre);
```

12. Ajouter la contrainte de clé primaire à prêt :

```
ALTER TABLE Prêt ADD CONSTRAINT pk_pret PRIMARY KEY (codePret);
```

13. Modifier la table livre en y ajoutant la colonne Nauteur et la contrainte indiquant que cette colonne est une clé étrangère :

```
ALTER TABLE Livre ADD Nauteur NUMBER;
```

```
ALTER TABLE Livre ADD CONSTRAINT fk_Nauteur FOREIGN KEY (Nauteur)  
REFERENCES Auteur(codeAuteur);
```

14. Supprimer la table possède :

```
DROP TABLE Possède;
```

15. Ajouter une contrainte à la table livre qui assure que titre de livre est une valeur bien déterminée :

```
ALTER TABLE Livre ADD CONSTRAINT ck_titre CHECK (titre IN ('Valeur1',  
'Valeur2', 'Valeur3'));
```

16. Ajouter les champs langue et NbreExemplaires à la table livre :

```
ALTER TABLE Livre ADD langue VARCHAR2(50);
```

```
ALTER TABLE Livre ADD NbreExemplaires NUMBER;
```

17. Ajouter la contrainte qui assure l'unicité du nom et prénom du membre :

```
ALTER TABLE Membre ADD CONSTRAINT uq_nom_prenom UNIQUE (nom,  
prenom);
```

18. Créer des index sur la clé primaire et une des clés étrangères de la table prêt :

```
CREATE INDEX idx_pk_pret ON Prêt(codePret);  
  
CREATE INDEX idx_fk_Nmembre ON Prêt(Nmembre);
```

- On peut déduire que des index ont été créés pour accélérer les recherches basées sur la clé primaire et la clé étrangère.

19. Créer des index sur les champs suivants :

```
CREATE INDEX idx_nom_auteur ON Auteur(nom);  
  
CREATE INDEX idx_nom_prenom_membre ON Membre(nom, prenom);  
  
CREATE INDEX idx_titre_livre ON Livre(titre);
```

20. Chercher les index créés en utilisant la vue user_indexes :

```
SELECT index_name FROM user_indexes;
```

21. Supprimer l'index déjà créé sur la table auteur :

```
DROP INDEX idx_nom_auteur;
```

22. Ajouter la contrainte qui spécifie que dateRetour est >= datePret :

```
ALTER TABLE Prêt ADD CONSTRAINT ck_dateRetour CHECK (dateRetour >= datePret);
```

23. Afficher les contraintes en utilisant la vue user_Constraints :

```
SELECT constraint_name, constraint_type, table_name FROM user_constraints;
```