
Laboratoire 1

Conception du schéma relationnel

1 Introduction

Au cours de la session, les laboratoires auront comme sujet le développement d'une application de gestion de données, à l'image des projets qui se font en entreprise. Chacun des quatre (4) laboratoires du cours portera sur un ou plusieurs concepts vus en classe, et sera effectué en équipe de **trois (3) à quatre (4)** étudiants.

Dans ce premier laboratoire, votre travail consistera à faire l'analyse et la conception du schéma relationnel et des règles d'affaires de l'application pour la location en-ligne de films (*voir le document de mise en situation*). L'évaluation du travail sera faite à partir d'un rapport décrivant les principales étapes de conception, ainsi que de code source implémentant les fonctionnalités précisées dans cet énoncé.

2 Travail à réaliser

Tâche 1 : Schéma conceptuel

Pour cette tâche, vous devrez identifier toutes les entités du système, leur attributs et leurs associations. Vous devrez ensuite représenter ces informations sous la forme d'un diagramme de classe UML. **Note** : ce diagramme sera utilisé au cours du laboratoire 2 pour définir les objets de votre application.

Consigne 1 : Considérez les huit (8) cas d'utilisation du document de mise en situation.

Consigne 2 : Assurez-vous de bien identifier les informations suivantes :

- Le nom des classes ;
- Le nom, le type et la **multiplicité** des attributs ;
- Le nom d'une association (sauf si agrégation, composition ou spécialisation) ainsi que ses multiplicités ;
- Les contraintes sur les relations de spécialisation (disjointe/chevauchante, complète/incomplète)

Consigne 3 : Prenez soin de bien identifier les classes partageant des attributs communs. Ces classes devraient être des spécialisations d'une même classe parente.

Consigne 4 : **Ne mettez pas** de clés primaires artificielles ni de clés étrangères dans ce diagramme. On devrait retrouver ces éléments **uniquement** dans le schéma relationnel de la Tâche 2.

Tâche 2 : Schéma relationnel

Réalisez le schéma conceptuel fait à la tâche précédente en schéma relationnel. Référez-vous au chapitre 3 du manuel pour cette étape. **Évitez la redondance de données en normalisant vos tables en 3FN.** Dans un premier temps, vous devrez modéliser le schéma relationnel sous la forme d'un nouveau diagramme UML montrant pour chaque table :

- Le nom de la table ;
- Le nom et le type **Oracle** de chaque colonne de la table (voir la page [Web suivante](#)) ;
- La clé primaire ;
- Les clé étrangères et les tables référées par ces clés. Celles-ci doivent être représentées à l'aide de flèches allant vers la table référencée par la clé.

Ensuite, vous utiliserez ce diagramme pour écrire un script SQL faisant la création de vos tables (CREATE TABLE).

Consigne 1 : Assurez-vous de bien identifier les contraintes de valeur (NOT NULL) et des longueurs de caractères appropriées pour les attributs de type VARCHAR dans vos tables, ainsi que les clés candidates (UNIQUE).

Consigne 2 : Évitez d'utiliser SQL Developer pour la création de vos tables. Bien que cet outil permet de générer le code SQL demandé, ce code n'est pas dans un format facilitant la lecture. Vous serez pénalisés si le script SQL n'est pas **suffisamment lisible**.

Consigne 3 : Bien que n'êtes pas obligé de mettre des commentaires dans votre script, choisissez des noms simples et descriptifs pour vos tables, colonnes et contraintes, et évitez le plus possible les acronymes obscurs (ex : PK_TB_FR). Évitez également les accents dans les noms.

Tâche 3 : Contraintes et procédures

Pour la troisième tâche, vous devrez identifier et implémenter les règles d'affaires (contraintes) associées à l'application de location en-ligne de films (**cas d'utilisation 1 à 4 seulement**). Pour l'implémentation des contraintes identifiées, vous choisirez une des trois approches suivantes :

1. Contrainte statique dans la table (CHECK) ;
2. Contrainte de domaine à l'aide d'une clé étrangère (FOREIGN KEY) ;
3. Contrainte dynamique complexe (TRIGGER).

Ensuite, vous devrez identifier les opérations de l'application de location de films (**cas d'utilisation 1 à 4 seulement**) qui **mettent à jour l'état de la BD**, et encapsuler ces opérations à l'aide de procédures stockées (CREATE PROCEDURE). Par exemple, une procédure `p_ajouterClient` pour la création d'un nouveau client.

Consigne 1 : Lorsque possible, employez un CHECK au lieu d'un TRIGGER.

Consigne 2 : Pour les contraintes complexes sur les chaînes de caractères, utilisez la fonction d'Oracle REGEXP LIKE.

Consigne 3 : Assurez vous de **bien tester** vos contraintes.

Consigne 4 : Évitez de mettre une clé artificielle comme paramètre d'une procédure d'encapsulation. Par exemple, une procédure p_ajouterClient devrait seulement prendre en paramètre les informations relatives au nouveau client, et la clé artificielle (ex : ID_client) devrait être générée à l'intérieur de la procédure.

Tâche 4 : Insertion des données

La quatrième tâche sera d'écrire un programme Java utilisant JDBC (dernière version, téléchargement du site d'oracle) pour faire l'insertion des données (fichiers xml) fournies dans votre BD. De plus, vous pouvez vous servir du programme de lecture fourni avec l'énoncé.

Consigne 1 : Commentez votre code pour en faciliter la compréhension.

Consigne 2 : Si vous utilisez le programme Java de lecture, assurez-vous d'inclure les libraires xmlpull 1.1.3.4c.jar et xpp3-1.1.3.4.C.jar.

Consigne 3 : Pour éviter des problèmes avec les caractères réservés (ex : '&') et les types spéciaux (ex : DATE), utilisez l'approche PreparedStatement au lieu de Statement. De même, utilisez la méthode addBatch() pour accélérer l'insertion des données.

Consigne 4 : Assurez-vous que **toutes** les **données** ainsi que leurs **attributs** soient insérées par votre programme :

- 21550 clients ; (avant les triggers/checks)
- 631 films ;
- 4549 personnes.

Consigne 5 : Pour le nombre de copies initialement disponibles d'un film, générez aléatoirement un nombre entier compris entre 1 et 100.

Consigne 6 : Veuillez noter que des points pourraient être retirés pour un temps d'insertion excessif pour les données (+10 min).

Consigne 7 : Vous pouvez utiliser SQL pour la création des tables

Tâche 5 : Rédaction du rapport

Enfin, la dernière tâche du laboratoire sera de rédiger un rapport décrivant votre travail et justifiant vos décisions de conception.

Note : utilisez le gabarit prévu à cet effet.

3 Consignes de remise

- Avant la **fin de la première séance** du laboratoire, vous devrez envoyer par courriel à votre responsable des travaux le *document de planification* complété.
- Avant la date limite disponible sur le calendrier du cours, vous devrez remettre sur Moodle les trois éléments suivants :
 1. Le gabarit de rapport complété ;
 2. Le script SQL pour la création de vos tables ;
 3. Le code source de votre programme Java d'insertion.

Note : Tous les fichiers de remise doivent être dans un répertoire compressé ayant comme nom `log660-lab01-equipeX`, où X est le numéro de votre équipe. Consultez le site du cours pour avoir plus de détails sur les consignes de remise.

4 Barème de correction

L'évaluation du laboratoire sera faite sur un total de 100 points, distribués selon la grille de correction du fichier excel dans la boîte de remise.

5 Pénalités et retard

Un travail **remis en retard recevra automatiquement la note 0**. Une pénalité allant jusqu'à 10% pourra également être appliquée à tout travail ne respectant pas rigoureusement les directives de remise.

0 Recap des tâches

Tâche 1 : Schéma conceptuel

Tâche 2 : Schéma relationnel

Tâche 3 : Contraintes et procédures

Tâche 4 : Insertion des données

Tâche 5 : Rédaction du rapport