

UNIVERSITÉ DE SHERBROOKE
DÉPARTEMENT D'INFORMATIQUE

Devoir # 2 - Automne 2017
IFT 287

Exploitation de base de données relationnelles et OO

Devoir à remettre au plus tard le vendredi 13 octobre 2017 à 23h59.

Ce devoir a pour but de vous faire pratiquer l'analyse et la conception d'un système basé sur une base de données relationnelle. Vous devrez concevoir le diagramme entité-association, le transformer en modèle relationnel ainsi qu'implémenter une solution au problème dans le langage Java en utilisant JDBC.

Noms : _____

Ce devoir est à faire en équipe de 2 obligatoirement.
Vous devez soumettre via *turnin* tous vos fichiers de code (.java), les fichiers SQL ainsi que le diagramme entité-association et le schéma relationnel. Votre fonction `main` doit se trouver dans la classe `Devoir2`.

Il est fortement recommandé de faire les différentes étapes du devoir dans l'ordre présenté.

Une erreur lors de la soumission vous fera perdre 25% de la note.

Tous vos fichiers (code et script SQL) doivent être au format UTF-8 sans BOM.

Pour accélérer la conception et l'implémentation de systèmes complexes utilisant des bases de données relationnelles, il est utile de réfléchir à l'interaction entre les données du système, en plus de la façon dont la base de données doit être construite. Dans ce devoir, vous avez à réaliser les tâches suivantes :

1. Écrire un diagramme entité-association.
2. Transformer le diagramme entité-association en modèle relationnel et écrire le script permettant de générer la base de données associée.
3. Écrire un programme en Java qui utilisera la base de données créée aux étapes précédentes.

Le détail de chacune des parties est donné dans les sections suivantes.

1 Partie 1

Le juge en chef du district judiciaire de St-François désire embaucher des étudiants pour développer un nouveau système de gestion des procès. Vous devez donc lui prouver que vous êtes les meilleures personnes pour développer ce logiciel. Afin de bien planifier le développement de l'application, vous devez créer le diagramme entité-association de ce problème. Pour ce faire, vous devez vous baser sur la description du problème qui suit.

1.1 Description du problème

Le logiciel demandé doit permettre de gérer les procès du district judiciaire de St-François. Le logiciel doit donc permettre à un greffier d'entrer toutes les informations nécessaires à la bonne tenue de la justice.

La première étape réalisée par le greffier est d'inscrire les juges dans le logiciel lors de leur nomination par la ou le Ministre de la Justice du Québec. Les informations sur un juge ou une juge ne peuvent être supprimées. Par contre, la personne occupant le poste de juge peut être retirée de la liste des juges pouvant présider un procès si elle est transférée dans un autre district, si elle prend sa retraite ou si elle décède. Il est important que les procès présidés par un ou une juge ne soit pas perdu en cas de départ.

Le greffier doit aussi pouvoir inscrire des procès. Un procès est présidé par un juge et se déroule potentiellement en plusieurs séances. Le procès peut être tenu soit devant un jury, composé de personnes issues de la société

civile, soit directement devant le juge. Chaque procès est composé d'une partie défenderesse et d'une partie qui poursuit.

Voici la liste complète des commandes que votre système doit pouvoir supporter, incluant les paramètres (entre <>).

- **ajouterJuge** <idJuge> <prenom> <nom> <age>
Permet d'ajouter un juge au système.
- **retirerJuge** <idJuge>
Permet de retirer un juge de la liste des juges disponibles.
- **ajouterAvocat** <idAvocat> <prenom> <nom> <type>
Permet d'ajouter un avocat pour représenter un client. Le type peut être 0 pour un avocat privé, 1 pour un avocat du directeur des poursuites criminelles et pénales.
- **ajouterPartie** <idPartie> <prenom> <nom> <idAvocat>
Permet d'ajouter une personne participant à un procès comme défendeur ou poursuivant.
- **creerProces** <idProces> <idJuge> <dateInitiale> <devantJury> <idPartieDefenderesse> <idPartiePoursuivante>
Permet de créer un nouveau procès en indiquant si le procès se tient devant jury (1) ou devant un juge seul (0).
- **inscrireJury** <nas> <prenom> <nom> <sexe> <age>
Permet d'inscrire une personne comme jury dans le système.
- **assignerJury** <nas> <idProces>
Permet d'assigner un jury à un procès.
- **ajouterSeance** <idSeance> <idProces> <dateSeance>
Permet d'ajouter une séance supplémentaire à un procès.
- **supprimerSeance** <idSeance>
Permet de supprimer une séance future à un procès non terminé.
- **terminerProces** <idProces> <decision>
Permet d'indiquer qu'un procès est terminé. Si la poursuite perd, la décision est à 0, si elle gagne, la décision est à 1.
- **afficherJuges**
Affiche la liste de tous les juges actifs.
- **afficherProces** <idProces>
Affiche les informations pour un procès.
- **afficherJurys**
Affiche la liste des jury disponibles pour participer à un procès.
- **quitter**

La méthode **afficherJuges** affiche la liste de tous les juges actifs enre-

gistrées dans le système, incluant la liste des procès qu'ils ont présidés. La méthode `afficherProces` affiche toutes les informations sur un procès et la liste des différentes séances. La méthode `afficherJurys` affiche tous les jurys disponibles pour participer à un procès. Les jurys assignés à un procès en cours ne doivent pas apparaître dans cette liste. Les autres commandes ne devraient pas avoir besoin de plus de descriptions. Attention de bien gérer les contraintes sur les commandes. Par exemple, il n'est pas possible de supprimer une séance qui a déjà eu lieu. On peut par contre supprimer n'importe quelle séance à venir. Si on termine un procès et qu'il reste des séances prévues, ces dernières doivent être supprimées. On ne peut pas non plus terminer un procès avant la date initiale prévue ni ajouter de séance à un procès terminé.

2 Partie 2

Afin de répondre aux besoins du juge en chef, vous avez décidé d'utiliser une base de données relationnelle. Vous devez créer un schéma relationnel pour votre base de données, basé sur le diagramme entité-association modélisé précédemment. Vous devez donc produire le schéma relationnel, en plus de créer le script SQL qui permet de construire la base de données dans PostgreSQL. En plus de votre diagramme relationnel, vous devez produire trois fichiers de commandes SQL. Le premier sert à créer votre base de données et doit se nommer `creation.sql`. Le second fichier doit pouvoir détruire toutes vos tables afin de faire le ménage. Ce second fichier doit se nommer `destruction.sql`. Les seules commandes dans ce fichier devraient être des `DROP TABLE`. Finalement, le troisième fichier doit se nommer `affiche.sql` et doit contenir des commandes `SELECT *` pour chacune des tables de votre base de données.

3 Partie 3

Maintenant que vous avez conçu les schémas du système qui gèrera les procès, vous devez implémenter une solution de base en Java pour montrer aux décideurs que vous pouvez développer l'application demandée. Vous devez donc créer un programme qui lit des commandes à la console et qui les exécute. Les commandes sont celles définies dans la partie 1. Les commandes

doivent modifier la base de données. Vous devez vous baser sur le code fourni sur le site web du devoir. Votre programme doit donc recevoir en paramètre le nom du serveur (local ou dinf), le nom de la base de données, le nom de l'utilisateur de la base de données et son mot de passe (comme dans les exemples du cours et le code fourni).

4 Soumission

Vous devez soumettre, sur *<http://opus.dinf.usherbrooke.ca/>* dans le projet TP2, un fichier nommé TP2.zip contenant vos deux schémas au format PDF (diagramme entité-association et schéma relationnel). Vous devez de plus inclure tous vos fichiers de code (.java), ainsi que trois fichiers SQL (*creation.sql*, *destruction.sql* et *affiche.sql*). Afin de soumettre correctement vos fichiers de code, vous devez exporter votre projet. Pour exporter votre projet vous devez cliquer droit sur votre projet et choisir l'option Exporter. Choisissez ensuite l'option Système de fichier. Sur la page suivante, vous devez sélectionner le projet que vous souhaitez exporter, ainsi que l'emplacement où faire la sauvegarde. Une fois votre projet exporté, vous pourrez copier vos fichiers de schéma et vos scripts SQL à la racine du dossier créé (le dossier contenant le dossier `src`). Vous pourrez ensuite compresser votre répertoire racine sous le nom de TP2.zip. Vous pourrez soumettre votre fichier compressé sur *turnin*.

Votre soumission devrait contenir au minimum la description qui suit. Eclipse ajoute plusieurs fichiers, il est donc possible, et normal, que vous ayez plus de fichiers dans votre soumission.

TP2.zip

```
+--> TP2 (dossier)
    +--> src (dossier)
        |   +--> tp2 (dossier)
        |       |   +--> Vos fichiers de code (.java)
    +--> creation.sql
    +--> destruction.sql
    +--> affiche.sql
    +--> Diagramme Entité association, en PDF
    +--> Diagramme relationnel, en PDF
```

Bonne chance!