

TD/TP 6

1 JDBC: exécution d'ordre SQL non select

1. Ecrire un programme complet et autonome Java/JDBC qui effectue l'action 1 du cahier des charges (employé : créer un village). Pour simplifier il n'y aura aucune lecture au clavier. Indication : Inspirez-vous de l'exemple du cours.

Structure du TD Le but du TD est de programmer en Java/JDBC les actions du CC. Pour simplifier toutes les actions seront faites sur le compte du programmeur, et on ne vérifiera ni les contraintes, ni les paramètres.

- (a) Récupérez et exécutez le corrigé du TD qui permet en particulier de peupler la base.
 - (b) Récupérez sur la page du cours, compilez et exécutez `Menu.java`. Entrez les différents choix possibles pour vous familiariser avec.
 - (c) Ce TD va consister à écrire le corps des fonctions Java du menu ci-dessus, et à en rajouter d'autres bien sûr. Attention: cela peut nécessiter d'adapter leurs paramètres et leurs valeurs de retour. Dans tout le TD JDBC on n'écrira aucune fonction PL/SQL, mais on appellera celles qu'on a écrites lors des TD sur PL/SQL quand l'énoncé le demandera explicitement.
 - (d) Dans tous les exercices, les éventuels paramètres des actions seront lus au clavier dans la fonction (donc pas passés en paramètres dans `Menu.java`), et leurs valeurs de retour affichées (donc pas renvoyées).
 - (e) Testez bien sûr chaque fonction, y compris en vérifiant sous un client interactif.
2. *Exécution d'ordre SQL (non curseur)* Écrivez la fonction Java effectuant l'action des employés : créer un village. Les valeurs des colonnes de Village sont lues au clavier (comme indiqué plus haut).
 3. *Exécution d'ordre SQL (non select)* Écrivez la fonction Java effectuant l'action des employés de modifier un village. On suppose trois paramètres : l'identifiant, la capacité et l'activité.
 4. *Exécution d'ordre SQL (non select)* Écrivez la fonction *traitement3* effectuant l'action des employés : traitement 3 (n'appellez pas la procédure PL/pgSQL correspondante).

2 JDBC: connexion, portabilité

1. *Connexion*
 - (a) Ecrivez la fonction Java de connexion à votre compte SGBD, et appelez-la en début du *main*. Pour simplifier son utilisation, codez en dur dedans les paramètres de connexion.
 - (b) Ecrivez la fonction Java de déconnexion.
2. *Connexion (et ordre SQL non select)* Ecrivez une fonction qui insère un village dans votre table Village, et qui en insère aussi un autre dans celle d'un autre compte. Attention : vous n'avez pas les droits directs sur cette dernière table, utilisez donc son compte SGBD, y compris pour tester sous le client interactif.

3 JDBC : gestion des erreurs, curseurs

1. Gestion des erreurs

- (a) Exécutez votre programme en faisant quelques tests en lui entrant des valeurs créant des erreurs, par exemple :
 - déconnexion puis traitement 3,
 - traitement 3 après avoir détruit la table séjour,
 - insertion d'une chaîne trop longue.
- (b) On reprend la fonction Java effectuant le traitement 3, mais cette fois en utilisant *try-catch*. Affichez bien sûr le message d'erreur de Java, précédé d'un message personnel. Testez avec les mêmes exemples que ci-dessus.

2. Exécution d'ordre SQL : curseur (1/2)

- (a) Ecrivez la fonction effectuant l'action des clients *authentification* : consulter son (éventuelle) ligne dans la table *Client*, à partir de son nom et de son prénom et afficher le message correspondant.
- (b) Ecrivez la fonction effectuant l'action des employés : consulter les villages (n'appellez pas la procédure PL/pgSQL).
- (c) Ecrivez la fonction effectuant l'action des clients : traitement 1 (n'appellez pas la procédure PL/pgSQL). Ecrivez la fonction effectuant l'action des clients : consultation des lignes lui correspondant dans les tables *Sejour* et *Village* (n'appellez pas la procédure PL/pgSQL correspondante).

3. Exécution d'ordre SQL : curseur (2/2) Ecrivez la fonction effectuant l'action des clients : traitement 2 (n'appellez pas la procédure PL/pgSQL correspondante).

4 JBDC : appel de procédures stockées

1. Exécution d'ordre non SQL (procédure stockée)

- (a) Ecrivez la fonction Java qui appelle la procédure stockée PL/pgSQL *créerVillage*.
- (b) Ecrivez la fonction Java qui appelle la fonction stockée PL/pgSQL *traitement3*.
- (c) Ecrivez la fonction Java qui appelle la procédure stockée PL/pgSQL *traitement3_out*.
- (d) Ecrivez la fonction Java qui appelle la procédure PL/pgSQL *traitement2*.