

Instructions générales

1. Les instructions générales du projet de SQL restent d'application pour l'examen.
 - Utilisez un schéma nommé **examen**.
 - Pour toute fonctionnalité demandée :
 - Si c'est possible, implémentez-la sur base de contraintes d'intégrité.
 - Sinon, si c'est possible, implémentez-la sur base d'un trigger.
 - Sinon, si c'est possible, implémentez-la sur base d'une vue.
 - Sinon, si c'est possible, implémentez-la sur base d'une procédure stockée.
 - Utilisez le modèle transactionnel et évitez d'ignorer silencieusement les erreurs.
 - Attention aux injections de SQL.
 - Chaque PreparedStatement ne doit être préparé qu'une seule fois.
 - Il ne faut, en Java, faire des SELECT **que** sur une seule chose à la fois. Cette chose peut être une table, une vue ou une procédure stockée.
2. Vous n'avez pas accès à Internet, mais vous trouverez dans le répertoire U: une copie du contenu du cours qui se trouve sur moodle.
3. Vous devez créer :
 - un fichier script.sql contenant toutes les commandes SQL à exécuter au serveur ;
 - un projet Java contenant la ou les classes Java.
4. 15 minutes avant la fin de l'examen :
 - a. Vous recopiez le fichier script.sql à la racine de votre projet Java.
 - b. Vous créez un fichier NOM_PRENOM.zip (par exemple DAMAS_CHRISTOPHE.zip) contenant l'intégralité de votre projet Java.
 - c. Vous copiez ce fichier à la racine du disque U:.
 - d. **Finalement vous vérifiez que ce fichier contient bien le projet Java contenant le fichier script.sql et le répertoire src avec vos sources Java.**

ATTENTION DATAGRIP PLANTE PARFOIS : SAUVEZ SOUVENT !

Enoncé

Un magasin vous demande de commencer l'implémentation d'un système simplifié permettant de commander des articles.

Tables

Il y aura trois tables : articles, commandes et lignes_de_commande.

Pour un article, on retient son nom, son prix en euros (pour simplifier, utilisez un entier), son poids en grammes (entier). Certains articles ne peuvent être achetés qu'en quantité limitée. Pour ces articles, on retiendra la quantité maximale autorisée par commande. Par exemple, si la quantité maximale autorisée est de 4, une commande ne pourra contenir qu'au maximum 4 fois cet article. S'il n'y pas de quantité maximale autorisée pour un article, cette valeur sera mise à NULL.

Pour une commande, on retient le nom du client l'ayant passée, sa date, son type (« livraison » ou « à emporter ») et son poids total. Le poids total devra être égal à la somme des poids des articles contenus dans la commande. Lorsqu'on crée une commande, son poids total est initialisé à 0.

Pour une ligne de commande, on retient la commande, l'article et la quantité. Pour une commande donnée, il y a au maximum une ligne de commande par article.

Attention, dans le cadre de cet examen, vous ne pouvez pas rajouter de champs supplémentaires dans les tables hormis des clés primaires.

À développer :

- Il faut créer une procédure stockée pour ajouter un article à une commande. Cette procédure prend 2 paramètres : la commande et l'article. La procédure regardera s'il existe déjà une ligne de commande pour cette commande et cet article. Si une ligne de commande existe, alors la quantité de cette ligne de commande sera incrémentée de 1. Sinon, une nouvelle ligne de commande sera créée. La procédure retournera le nombre d'articles de la base de données qui ont été commandés dans au moins deux commandes différentes.

Une exception sera lancée dans les deux cas suivants :

- Si la quantité commandée dépasse, pour cette commande, la quantité maximale autorisée pour cet article.
- Si la commande est une livraison et que le prix total de la commande, suite à l'ajout, dépasse 1000 euros. Pour les commandes à emporter, il n'y a pas de prix limite.
- Il faut créer un mécanisme qui permet de mettre à jour le champ de la table commandes qui retient le poids total.
- Une classe Java qui se connecte au serveur : le programme demande à l'utilisateur le nom d'un client (via un Scanner) et affichera ensuite toutes ses commandes de type « livraison » triées par date. Pour une commande, on affichera son identifiant, sa date

et le nombre d'articles commandés au total. Si une commande est pour le moment vide, il faut quand même l'afficher avec 0 comme nombre d'articles commandés. Il n'est pas nécessaire de gérer les cas d'erreurs et il ne faut pas perdre de temps à afficher les données de manière jolie.

Pour rappel, voici deux instructions utiles pour les Scanner :

```
Scanner s = new Scanner(System.in);  
String nomClient = s.nextLine();
```

Si vous voulez tester votre système, vous pouvez insérer des articles et des commandes (via de simples INSERT) et appeler la procédure stockée d'ajout d'articles dans votre script.