

TDM : JDBC

Préliminaire :

Il faut absolument qu'un EDI (Environnement de Développement Intégré) Eclipse ou netbeans, etc. et MySQL soient déjà installés (ou accessibles) sur votre machine et que vous ayez les droits vous permettant de configurer ces installations. Donc, un serveur de base de données est à votre disposition sur une machine distante (mais sur le réseau local du Bâtiment Poincaré). Ce serveur nous permet de centraliser vos bases de données. On ait déjà créé une base distincte pour chacun de vous, avec le même nom que votre utilisateur, lors de la création de vos comptes.

Avant de commencer, vous devriez apprendre à vous servir du client de commande de MySQL ; ça serait, pour l'instant, un moyen le plus simple et le plus rapide d'effectuer des tests et vérifier les impacts de votre code Java sur la base de données.

Vous allez donc instancier les bases de données MySQL que vous traitez par la suite à partir de classes Java et en utilisant JDBC. Elles sont localisées sur la machine 192.168.22.48. Alors, tentez de vous connecter au serveur MySQL. Pour cela, il suffit de lancer le client MySQL, à partir de la console :

```
mysql -h 192.168.22.48 -u utilisateur -p mot_de_passe
```

Faire démarrer le travail :

L'interface entre une base de données et un programme Java se réalise par l'intermédiaire d'un driver JDBC ou pont ODBC-JDBC (Java DataBase Connectivity). En important le package "java.sql" et en installant un driver pour votre SGBD, vous aurez la possibilité de rédiger des requêtes SQL vers une base de données.

Pré-requis des installations :

Le logiciel « mysql-connector/j » se présente comme une librairie qui sera nécessaire pour qu'une application Java puisse se connecter à MySQL. Il est disponible, sur le lien : <https://dev.mysql.com/downloads/connector/j/>.

Donc, téléchargez et décompressez le fichier mysql-connector-java- x.x.tar.gz ou .zip dans un endroit de votre espace de travail qu'on appellera par la suite : CONNECTOR

A ce stade nous avons tout ce qui nous faut pour commencer à travailler avec JDBC et EDI en utilisant le SGBD MySQL.

- i. Lancez Eclipse ou netbeans ou bien un autre EDI de votre choix.
- ii. Créez un projet de type Java Project ayant comme répertoire des sources un répertoire appelé src et comme répertoire des binaires un répertoire bin.
- iii. Faites en sorte que ce projet puisse avoir accès à connector/j en spécifiant le fichier archive java (.jar) disponible dans le répertoire CONNECTOR comme faisant partie de la librairie du projet. Pour cela, ajoutez « mysql-connector/j » dans class path comme « add external libraries » de votre projet.

Exercice 1

Écrire un programme Java JDBC (une classe avec une seule méthode main) permettant de créer et de remplir une table sur votre compte du serveur de la base de données MySQL.

Vous pouvez accéder au serveur mysql à partir de votre application Java. Voici, l'instruction pour charger le pilote JDBC MySQL :

```
Class.forName ("com.mysql.jdbc.Driver");
```

Le port par défaut du serveur est 3306. Donc, l'url de la connexion avec le SGBD est comme suivant :

```
jdbc:mysql://192.168.22.48:3306/utilisateur
```

Utilisez la base courante. Dans cette base, créez et remplissez ensuite la table *Produit* ci-dessous :

Table Produit

<i>Référence</i>	<i>Désignation</i>	<i>Quantité</i>	<i>TVA</i>	<i>Prix_HT</i>
0001	PC MM MMX	5	19.6	6500.00
0002	Imprimante HP jet	20	19.6	3500.00
0003	Ecran 17 "	15	19.6	1200.00
0004	CD*10	100	19.6	99.00

Exercice 2

Ecrire un programme Java JDBC permettant d'afficher à l'écran toutes les données de la table *Produit*. La première ligne affichera les noms des attributs qu'on supposera connus à l'avance et les lignes suivantes contiennent les valeurs de ces attributs.

Exercice 3

Refaire l'exercice précédent mais cette fois-ci en affichant les produits dont le prix est supérieur à une certaine valeur entrée au clavier et en utilisant un *PreparedStatement*.

Exercice 4

Ecrire un programme Java JDBC qui permet de rentrer une requête quelconque au clavier et afficher son résultat comme dans les deux exercices précédents.