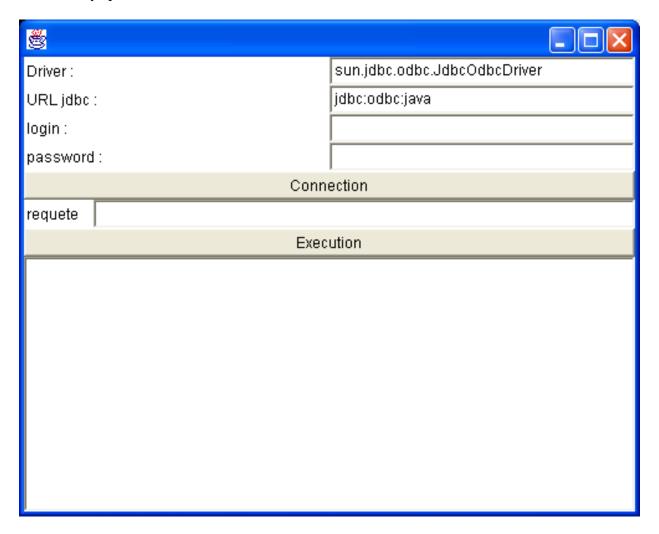
TP 6: Java-JDBC

Pierre Lefebvre

- 1) Tester un programme JDBC donné en cours pour valider la configuration de votre environnement
- 2) A partir de la maquette d'IHM suivante (source fourni), compléter l'implémentation de ce programme pour permettre une connexion et une consultation de la base de données Mysql. Vous vous limiterez à la consultation de tables.



3) L'objectif de ces exercices est de développer des méthodes de la classe Java ExoJDBC pour extraire et mettre à jour des données des tables du schéma *Parc Informatique*.

Curseur statique

Ecrire les méthodes:

- *ArrayList getSalles()* qui retourne sous la forme d'une liste les enregistrements de la table *Salle*.
- *Main* qui se connecte à la base, appelle la méthode *getSalles* et affiche les résultats (exemple donné ci-dessous)

nSalle	nomSalle	nbPoste	indIP
s01	Salle 1	3	130.120.80
s02	Salle 2	2	130.120.80

Ajouter une nouvelle salle dans la table Salle dans l'interface de commande, et lancez à nouveau le programme pour vérifier

Curseur modifiable

Ecrivez la méthode *void deleteSalle(int)* qui supprime de la table *Salle* l'enregistrement de rang passé en paramètre. Vous utiliserez la méthode *deleteRow* appliquée à un curseur modifiable. Appelez dans le *main* cette méthode pour supprimer l'enregistrement de la table *Salle* que vous avez ajouté en test, dans l'exercice précédent. Si l'enregistrement est rattaché à un enregistrement fils, ne forcez pas la contrainte référentielle, contentez-vous d'afficher le message 1451 renvoyé par MySQL, dans le bloc des exceptions.

Source de l'interface graphique

```
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
public class JDBCTest extends Panel
implements ActionListener
       TextField
                     nomDriver;
       TextField
                     urlConnection;
       TextField
                     nomLogin;
       TextField
                     motPasse;
       Button
                     boutonConnection;
       TextField
                     requeteSQL;
                     resultatRequete;
       List
                     boutonExecuter;
       Button
       public JDBCTest()
              Panel haut;
              Panel bas:
              haut = new Panel();
              bas = new Panel();
              boutonConnection = new Button("Connection");
              boutonConnection.addActionListener(this);
              boutonExecuter = new Button("Execution");
              boutonExecuter.addActionListener(this);
              Panel p1 = new Panel();
              p1.setLayout(new GridLayout(4, 2));
              p1.add(new Label("Driver:"));
              p1.add(nomDriver = new TextField(32));
              p1.add(new Label("URL idbc:"));
              p1.add(urlConnection = new TextField(32));
              p1.add(new Label("login:"));
              p1.add(nomLogin = new TextField(32));
              p1.add(new Label("password:"));
              p1.add(motPasse = new TextField(32));
              haut.setLayout(new BorderLayout());
              haut.add(p1, BorderLayout.NORTH);
              haut.add(boutonConnection, BorderLayout.SOUTH);
              Panel p2 = new Panel();
              p2.setLayout(new BorderLayout());
              p2.add(new Label("requete"), BorderLayout.WEST);
```

```
p2.add(requeteSQL = new TextField(32), BorderLayout.CENTER);
       Panel p3 = new Panel();
       p3.setLayout(new BorderLayout());
       p3.add(p2, BorderLayout.NORTH);
       p3.add(boutonExecuter, BorderLayout.SOUTH);
       bas.setLayout(new BorderLayout());
       bas.add(p3, BorderLayout.NORTH);
       bas.add(resultatRequete = new List(20));
       setLayout(new BorderLayout());
       add(haut, BorderLayout.NORTH);
       add(bas, BorderLayout.CENTER);
}
public void actionPerformed(ActionEvent evt)
       //System.out.prinln("Not implemented !!");
       resultatRequete.add("au boulot les gars !!!");
}
public static void main(String[] arg)
       JDBCTest
                    test;
       Frame f = new Frame();
       f.setSize(500, 400);
       test = new JDBCTest( );
       f.add(test, BorderLayout.CENTER);
       f.addWindowListener(new WindowAdapter() {
              public void windowClosing(WindowEvent e)
                    System.exit(0);
              }} );
       f.setVisible(true);
}
```

}