

TP : BDA

Bases de Données Avancées

M1

2012-2013

L. Kahloul

Pré-requis

1. Des connaissances en Java (POO);
2. (on a mis quelques cours pour les 2LMD, vous pouvez les consulter)
3. Installer le netbean:
<https://fr.netbeans.org/>
5. Installer le wampserver (pour mysql);
<http://www.commentcamarche.net/download/t/elecharger-34055064-wampserver>
7. Installer le Access;

Objectifs

- Construire une base de données avec **Access (Microsoft)**;
- Construire une base de données avec **MySQL**;
- Utiliser le **langage Java** pour **se connecter localement** ou **à distance** à ces bases de données et de les manipuler.

Connexion entre Java et ces SGBDs

- Cette connexion exige la présence d'un ensemble des **API (JDBC)**, de classes (**java.sql.***), et de **pilotes** pour établir ces connexions (par exemple le pilote **ODBC** pour Access, et le pilote **mysql** pour Mysql);
- Quelques pilotes viennent avec le java (d'origine, comme **odbc: sun.jdbc.odbc.JdbcOdbcDriver**), et d'autre doivent être ajoutés à l'environnement (le cas de **mysql: com.mysql.jdbc.Driver**);

Partie: I

Les étapes:

1. Installation de Access, et de Mysql;
2. Création de la base de données avec Access ou Mysql ;
3. Installation des pilotes nécessaires pour la communication entre java et ces deux SGBDs;
4. Écriture d'un programme java : ensemble de classes permettant de consulter, modifier les bases de données.
5. Développer une interface graphique avec java permettant de faire ces manipulation

Communication avec une BD access (1)

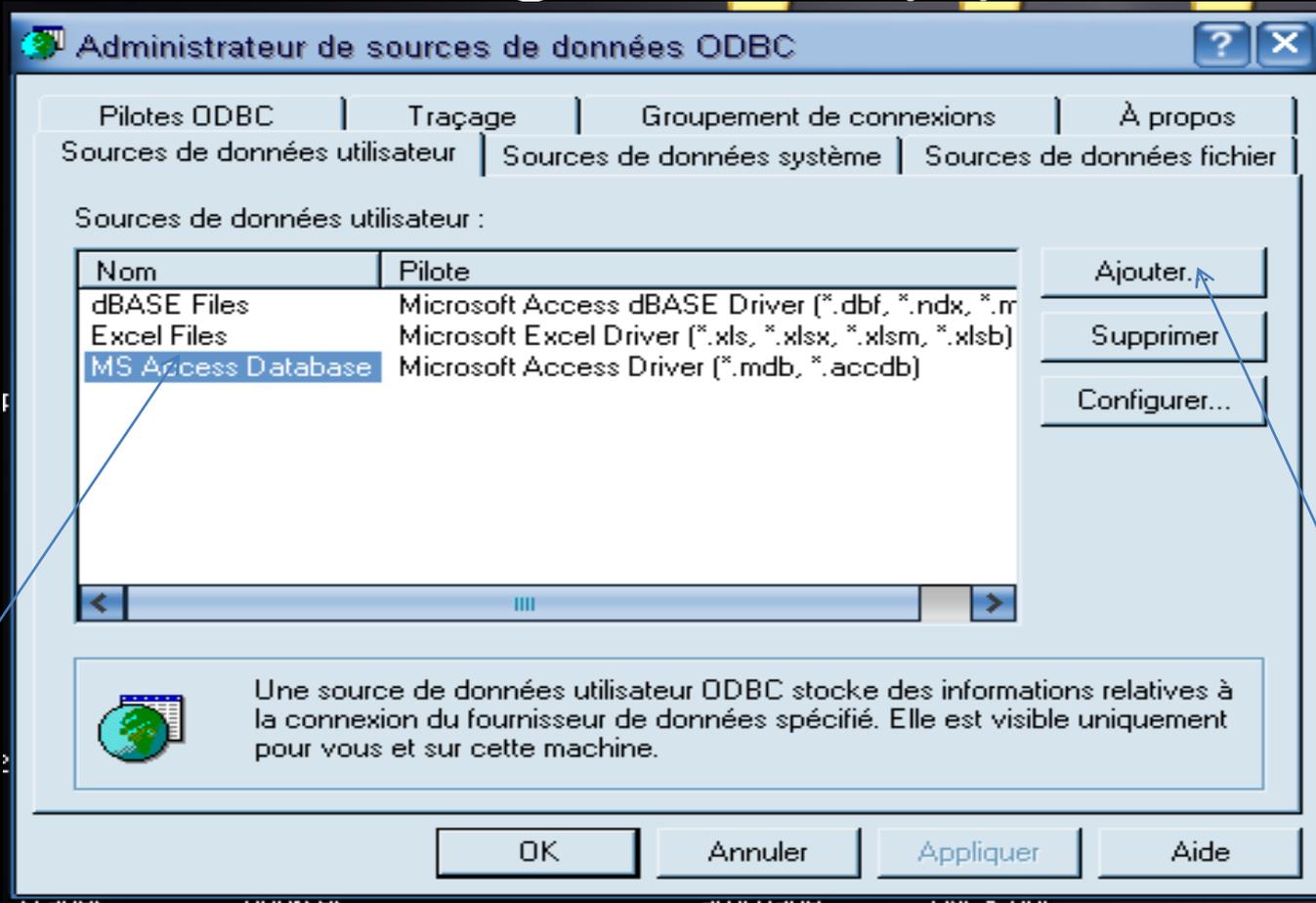
configuration (1)

- Après la création de la BD sous Access, on doit l'enregistrer avec le ODBC fourni avec le système d'exploitation. Cet enregistrement rend cette BD reconnu par Java



- À retrouver dans les outils d'administration du panneau de configuration

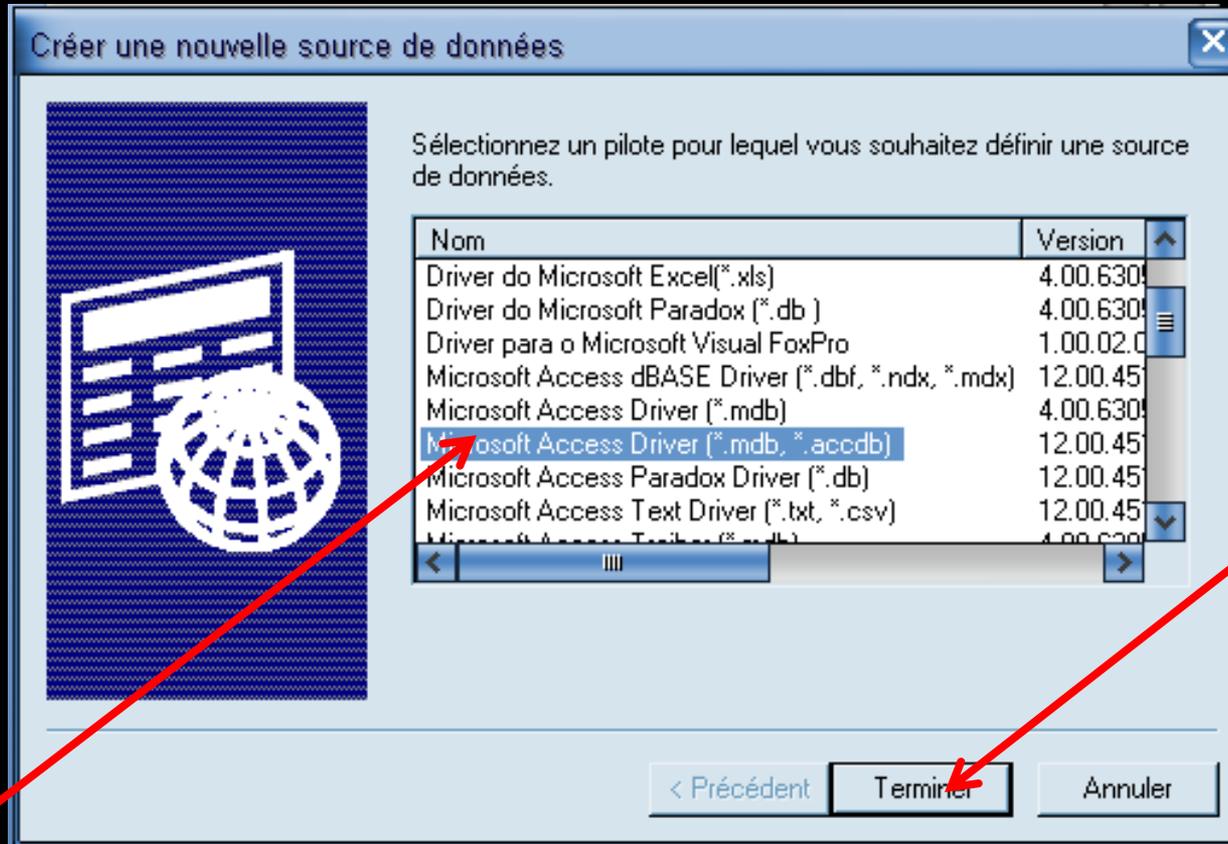
Communication avec une BD access(2) configuration (2)



1

2

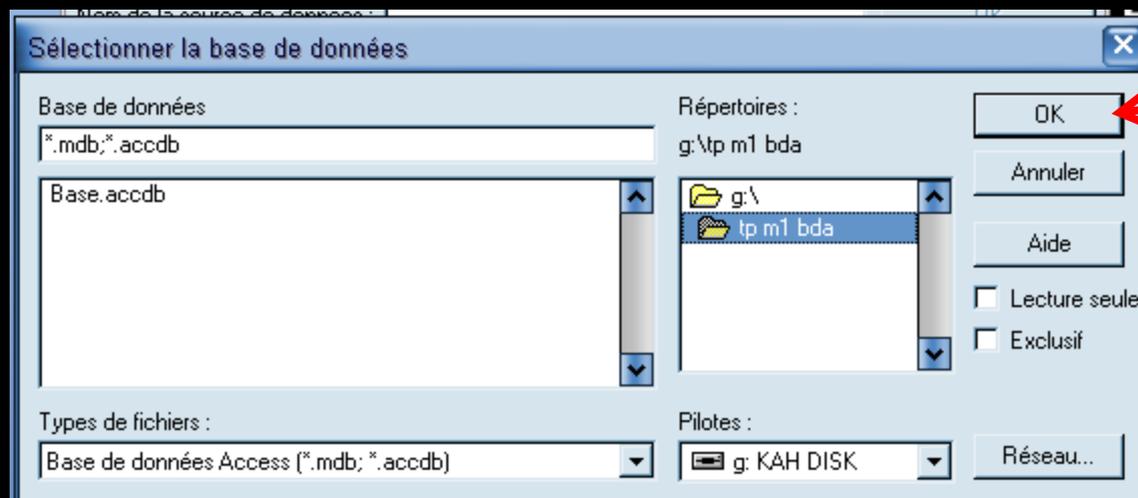
Communication avec une BD access(3) configuration (3)



1

2

Communication avec une BD access(4) configuration (4)



Communication avec une BD access(5) configuration (5)

À ce stade, la base de données base1 est reconnue comme une source de donnée, et donc peut être accessible via le pilote **odbc** installé avec le java

Communication avec une BD access (6) le programme Java(1)

Trois étapes:

- 1) Charger le pilote de la BD;
- 2) Établir la connexion avec la BD;
- 3) Exécuter des requêtes sur la BD.

Communication avec une BD access (7) le programme Java(2)

(1)Chargement du pilote

```
try{  
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");  
}  
catch(java.lang.ClassNotFoundException e){  
System.out.print(e);  
}
```

Communication avec une BD access(8) le programme Java(3)

(2)Établir la connexion

```
String DBurl = "jdbc:odbc:base1" ;  
  
try {  
    Connection con = DriverManager.getConnection(DBurl);  
  
catch (SQLException e) {  
    System.out.print(e);//traitement de l'exception  
}
```

The diagram consists of blue arrows pointing from text labels to parts of the code. The label 'Base de données' points to 'base1'. The label 'Sous protocole' points to 'odbc'. The label 'Protocole' points to 'jdbc'. The label 'Connexion avec la BD' points to the 'getConnection' method call.

Base de données

Sous protocole

Protocole

Connexion avec la BD

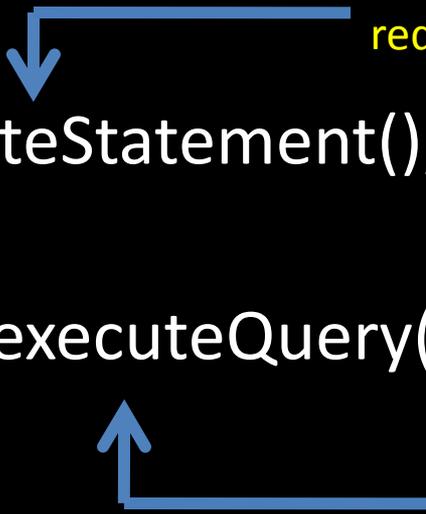
Communication avec une BD access (7) le programme Java(4)

(3) Exécution d'une requête

```
String requete = "SELECT * FROM table1";  
try {  
    Statement stmt = con.createStatement();  
  
    ResultSet résultats = stmt.executeQuery(requete);  
}  
catch(SQLException e){}
```

Création de la requête à envoyer

Exécution de la requête à envoyée et retour des résultats



Communication avec une BD access (9)

le programme Java(5)

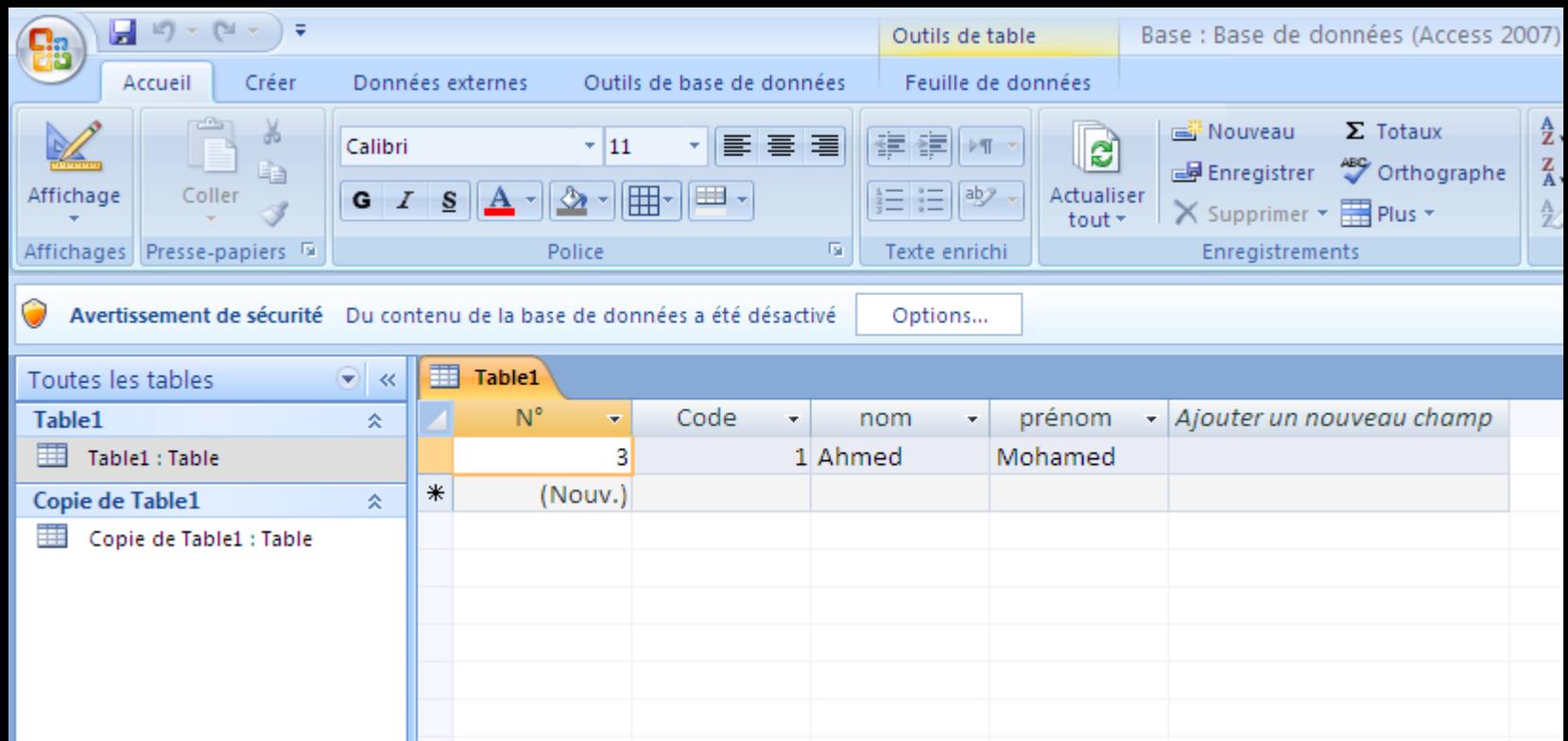
(4) Exploitation des résultats retournés par la requête:

```
try {
    ResultSetMetaData rsmd = résultats.getMetaData();
    int nbCols = rsmd.getColumnCount();
    while (résultats.next()) {
        for (int i = 1; i <= nbCols; i++)
            System.out.print(résultats.getString(i) + " ");
        System.out.println();
    }
    résultats.close();
} catch (SQLException e) {
    //traitement de l'exception
}
```

Ce code permet de lister toutes les éléments de la table

Communication avec une BD access (9) le programme Java(6)

Exemple : BD sous Access

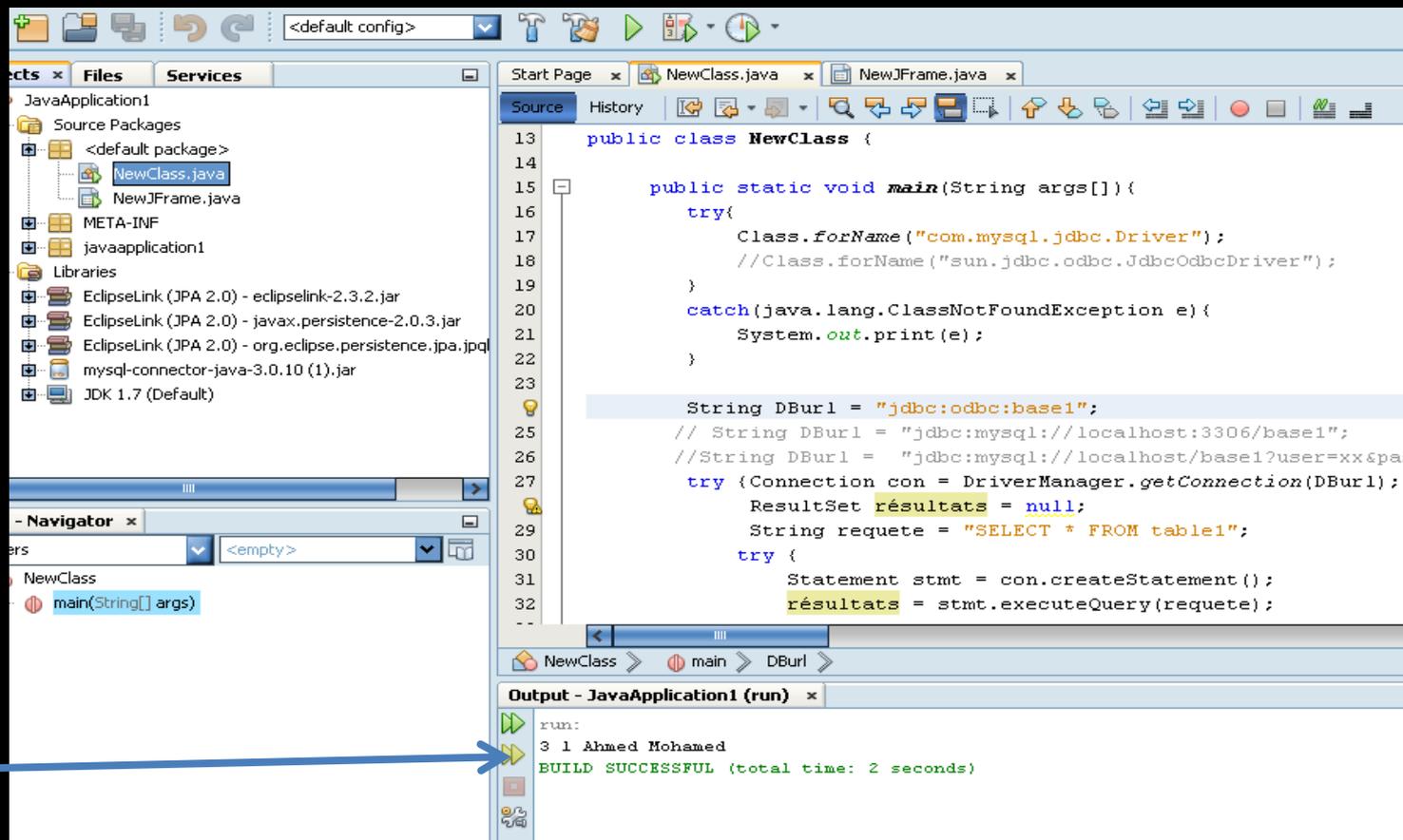


The screenshot displays the Microsoft Access 2007 interface. The title bar indicates the database is 'Base : Base de données (Access 2007)'. The ribbon is set to 'Outils de table' and 'Feuille de données'. The main area shows a table named 'Table1' with the following data:

N°	Code	nom	prénom	Ajouter un nouveau champ
3	1	Ahmed	Mohamed	
* (Nouv.)				

Communication avec une BD access (9) le programme Java(7)

Exemple : affichage sous java netbeans



The screenshot displays the NetBeans IDE interface. On the left, the Project Explorer shows a project named 'JavaApplication1' with source packages and libraries. The central editor shows the code for 'NewClass.java':

```
13 public class NewClass {
14
15     public static void main(String args[]) {
16         try{
17             Class.forName("com.mysql.jdbc.Driver");
18             //Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
19         }
20         catch(java.lang.ClassNotFoundException e){
21             System.out.print(e);
22         }
23
24         String DBurl = "jdbc:odbc:base1";
25         // String DBurl = "jdbc:mysql://localhost:3306/base1";
26         //String DBurl = "jdbc:mysql://localhost/base1?user=xx&pas
27         try {Connection con = DriverManager.getConnection(DBurl);
28             ResultSet résultats = null;
29             String requete = "SELECT * FROM table1";
30             try {
31                 Statement stmt = con.createStatement();
32                 résultats = stmt.executeQuery(requete);
33             }
34         }
35     }
36 }
```

The Output window at the bottom shows the execution results:

```
run:
3 1 Ahmed Mohamed
BUILD SUCCESSFUL (total time: 2 seconds)
```

A blue arrow points from the text 'Affichage Des éléments de la table' to the output window.

Affichage
Des
éléments
de la table

Communication avec une BD access (9) le programme Java(8)

Travail demandé 1:

- 1) Construire une BD sous access;
- 2) Rédiger un programme Java qui fait la connexion, et affiche le contenu de l'une des tables;
- 2) Utiliser le netbeans, pour développer une interface graphique pour afficher les différents éléments de la tables table1

Partie II

Communication avec une BD MySQL(1)

MySQL

1. Installer le **MySQL**;
 2. Vous pouvez utiliser le **WampServer**;
(<http://www.commentcamarche.net/download/telecharger-34055064-wampserver>)
-
1. Créer une BD; voir la figure suivante

Communication avec une BD MySql(2)

MySql (exemple d'une BD)

Nom de la BD

Une table

des données

The screenshot shows the phpMyAdmin interface for a MySQL database named 'base1'. The table 'table1' is selected, and its structure is displayed. The table has three columns: 'code', 'Nom', and 'Prénom'. The data is displayed in a table format with two rows:

code	Nom	Prénom
1	Ahmed	Mohamed
2	leila	mourad

The interface also shows a SQL query editor with the following query:

```
SELECT * FROM 'table1' LIMIT 0, 30
```

Navigation and action buttons are visible at the top, including 'Afficher', 'Structure', 'SQL', 'Rechercher', 'Insérer', 'Exporter', 'Importer', and 'Opérations'. The left sidebar shows the database 'base1' and the table 'table1' selected.

Communication avec une BD MySql(3)

MySql (exemple d'une BD)

Ajouter un autre utilisateur (un username, un mot de passe) et des privilèges d'accès

Nom de la BD

Ajouter un Utilisateur avec des privilèges d'accès

Utilisateur	Client	Mot de passe	Privilèges globaux	«Grant»	Action
<input type="checkbox"/> ammar	%	Oui	USAGE	Non	
<input type="checkbox"/> root	127.0.0.1	Non	ALL PRIVILEGES	Oui	
<input type="checkbox"/> root	:::1	Non	ALL PRIVILEGES	Oui	
<input type="checkbox"/> root	localhost	Non	ALL PRIVILEGES	Oui	
<input checked="" type="checkbox"/> x	x	Oui	ALL PRIVILEGES	Oui	
<input type="checkbox"/> xx	localhost	Oui	ALL PRIVILEGES	Oui	

Tout cocher / Tout décocher

Ajouter un utilisateur

Communication avec une BD MySql (4)

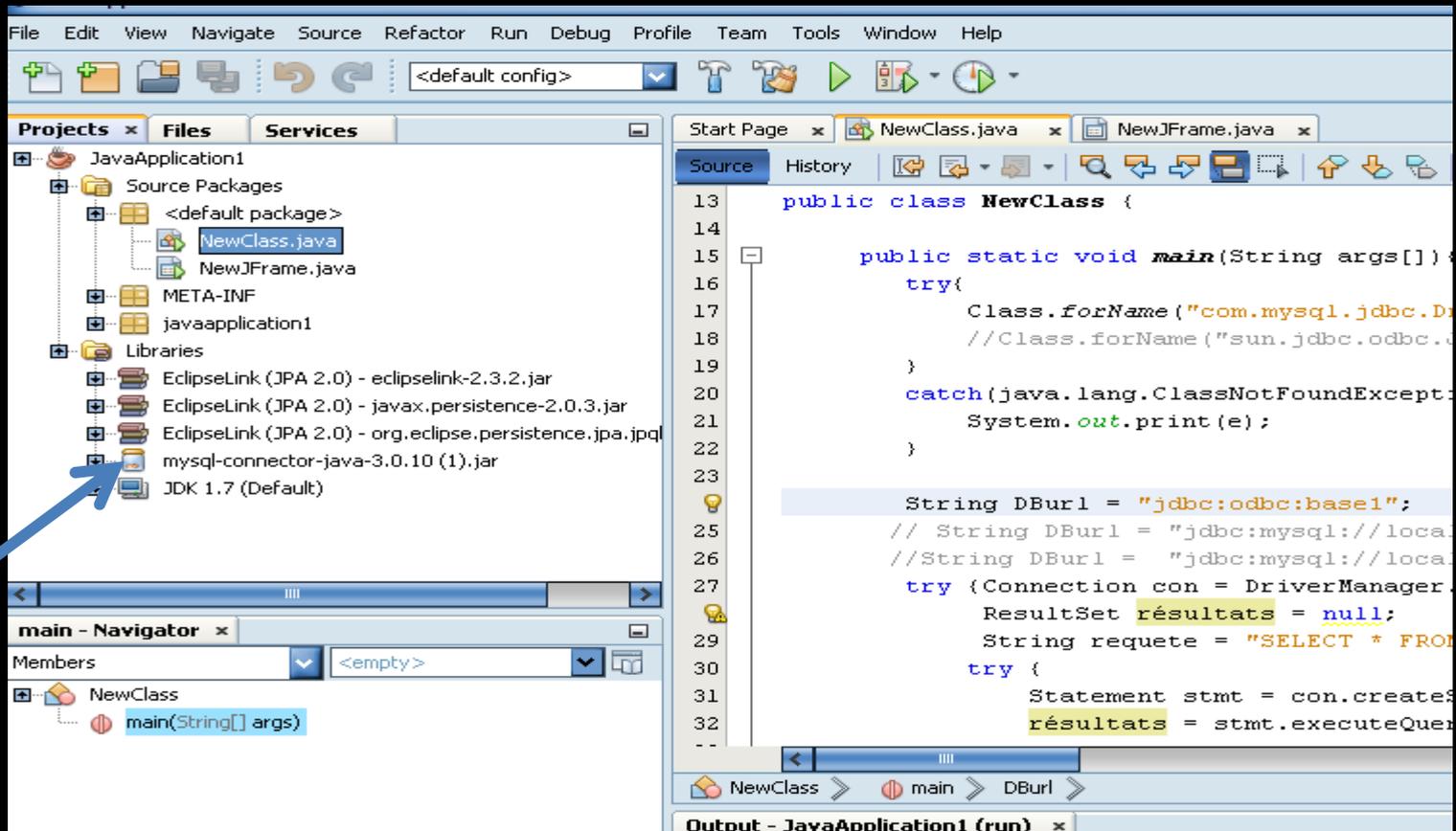
Le pilote mysql (1)

1. Ajouter le pilote de connexion MySql à netbeans:
mysql-connector-java-3.0.10-ga-bin.jar
3. À télécharger et à ajouter comme library dans le netbeans; (voir la figure suivante)
4. Maintenant, on peut avoir:

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

Ceci permet de charger le pilote par le programme java, pour avoir accès à MySql

Communication avec une BD MySql(5) le pilote mysql(2)



The screenshot shows the Eclipse IDE interface. In the Project Explorer on the left, the 'Libraries' section is expanded, and a blue arrow points to the 'mysql-connector-java-3.0.10 (1).jar' library. The main editor displays the source code of 'NewClass.java' with the following content:

```
13 public class NewClass {
14
15     public static void main(String args[]) {
16         try{
17             Class.forName("com.mysql.jdbc.Driver");
18             //Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
19         }
20         catch(java.lang.ClassNotFoundException e){
21             System.out.print(e);
22         }
23
24         String DBurl = "jdbc:odbc:base1";
25         // String DBurl = "jdbc:mysql://localhost:3306/";
26         //String DBurl = "jdbc:mysql://localhost:3306/";
27         try {Connection con = DriverManager.getConnection(DBurl, "root", "root");
28             ResultSet résultats = null;
29             String requete = "SELECT * FROM";
30             try {
31                 Statement stmt = con.createStatement();
32                 résultats = stmt.executeQuery(requete);
33             }
34         }
35     }
36 }
```

Le voilà
ajouté ici

Communication avec une BD MySQL(6) le programme Java (1)

- 1) Charger le pilote mysql;
- 2) Se connecter à la BD;
- 3) Faire des manipulation;

Communication avec une BD MySQL(7) le programme Java (2)

Charger le pilote

```
try{
    Class.forName("com.mysql.jdbc.Driver");
}
catch(java.lang.ClassNotFoundException e){
    System.out.print(e);
}
```

Communication avec une BD MySQL(8) le programme Java (3)

Connexion à la BD (localement)

```
String DBurl =  
    "jdbc:mysql://localhost/base1?user=xx&password=x";  
  
try {  
    Connection con = DriverManager.getConnection(DBurl);  
}  
  
catch( java.sql.SQLException e){System.out.print(e);
```

Communication avec une BD MySQL(9) le programme Java (4)

Exécution d'une requête

```
ResultSet résultats = null;
```

```
String requete = "SELECT * FROM table1";
```

```
try {
```

```
    Statement stmt = con.createStatement();
```

```
    résultats = stmt.executeQuery(requete);
```

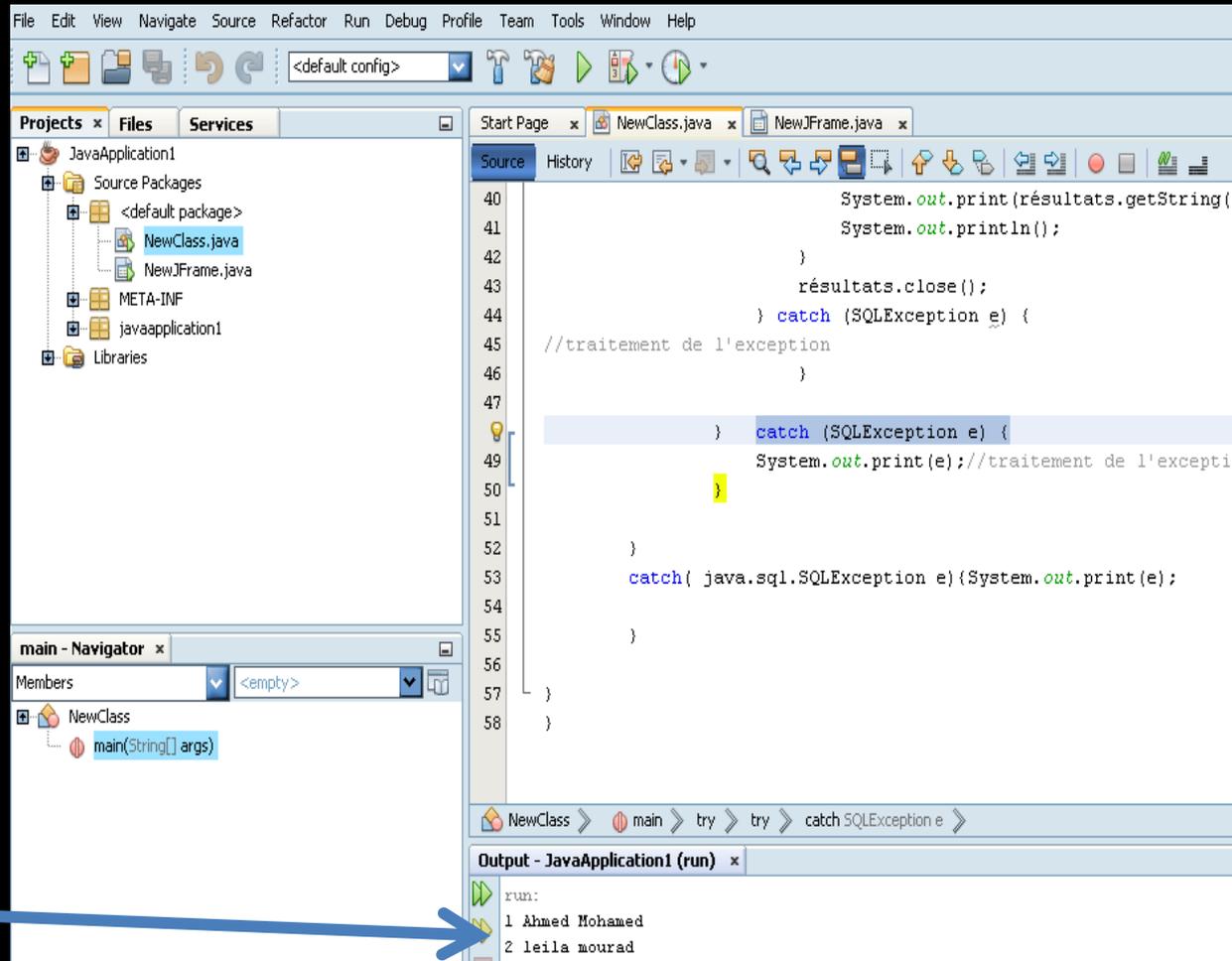
```
}
```

```
catch (SQLException e) {}
```

Communication avec une BD MySql(11)

Le programme Java (6)

Exemple de résultats



The screenshot shows an IDE with the following components:

- Projects View:** Shows a project named 'JavaApplication1' with source packages, including '<default package>', 'NewClass.java', and 'NewJFrame.java'.
- Source Editor:** Displays the code for 'NewClass.java'. The code includes a try-catch block for handling a 'SQLException'. The output of the program is visible in the 'Output' window at the bottom.
- Output Window:** Shows the results of the program execution: 'run: 1 Ahmed Mohamed' and '2 leila mourad'.

```
40         System.out.print(résultats.getString(
41         System.out.println();
42     }
43     résultats.close();
44     } catch (SQLException e) {
45 //traitement de l'exception
46     }
47
48     } catch (SQLException e) {
49         System.out.print(e);//traitement de l'excepti
50     }
51
52     }
53     catch( java.sql.SQLException e){System.out.print(e);
54
55     }
56
57 }
58 }
```

Output - JavaApplication1 (run) x

```
run:
1 Ahmed Mohamed
2 leila mourad
```

Contenu de la table
affichée

Communication avec une BD MySql(12)

le programme Java (7)

Travail demandé 2:

- 1) Construire une BD sous MySql (WampServer);
- 2) Rédiger un programme Java qui fait la connexion (localement), et affiche le contenu de l'une des tables;
- 3) Rédiger un programme Java qui fait la connexion (à distance), et affiche le contenu de l'une des tables;

Ici vous devez utiliser l'adresse comme suit, en ajoutant le port et bien sur l'url de la machine où la base est installée:

```
String DBurl = "jdbc:mysql://localhost:3306/base1";
```

- 4) Utiliser le netbeans, pour développer une interface graphique pour afficher les différents éléments de la tables table1

Références

Remarque:

1) Vous pouvez voir ce lien pour plus d'information:

[file:///G:/TP%20BDA%20M1/D%C3%A9veloppons%20en%20Java%20-%20JDBC%20\(Java%20DataBase%20Connectivity\).htm](file:///G:/TP%20BDA%20M1/D%C3%A9veloppons%20en%20Java%20-%20JDBC%20(Java%20DataBase%20Connectivity).htm)

Bon courage