



جامعة اللبنانية الدولية
LEBANESE INTERNATIONAL UNIVERSITY

CSIT461

Connecter une base de données MySQL en Java

EL BENANY Mohamed Mahmoud

20/12/2021

Connecter une base de données MySQL en Java

1. **Télécharger et installer la base de données MySQL**
2. **Créer une base de données dans MySQL**
3. **Téléchargez le pilote JDBC et incluez-le dans le chemin de classe**
4. **Connectez-vous avec MySQL**
5. **Tester la connexion JDBC**

1. Télécharger et installer la base de données MySQL

- **WAMP:**

<https://www.wampserver.com/en/download-wampserver-64bits/>

Ou

- **EASYPHP**

<https://www.easyphp.org/download.php>

Ou

- **XAMPP**

<https://www.apachefriends.org/fr/download.html>

Ou

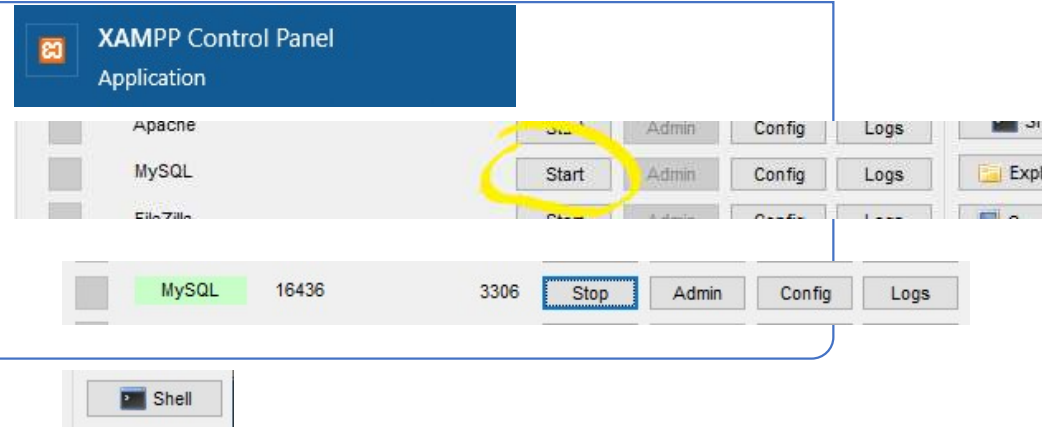
- **MySQL Server**

<https://dev.mysql.com/downloads/mysql/>

XAMPP

On a installé XAMP:

1. Lancer xampp control panel
2. Start MySQL
3. Start Shel
4. Se connecter à MariaDB
5. **Créer un base de donnée**



```
De11@DESKTOP-D8D9ALB c:\xampp
# mysql -u root -p
Enter password: ****
Welcome to the MariaDB monito
Your MariaDB connection id is
Server version: 10.4.19-Maria
Copyright (c) 2000, 2018, Ora
Type 'help;' or '\h' for help
MariaDB [(none)]>
```

Connecter une base de données MySQL en Java

1. **Télécharger et installer la base de données MySQL**
2. **Créer une base de données dans MySQL**
3. Téléchargez le pilote JDBC et incluez-le dans le chemin de classe
4. Connectez-vous avec MySQL
5. Tester la connexion JDBC

2. Créer une base de données dans MySQL

On a installé XAMP:

1. Lancer xampp control panel
2. Start MySQL
3. Start Shel
4. Se connecter à MariaDB

Maintenant, nous allons créer une base de données afin de pouvoir tester la connexion.

Pour créer une base de données dans MySQL, utilisez la requête SQL ci-dessous : `create database classbd;`

Création la base de donnée « classbd »

2. Créer une base de données dans MySQL

On a installé XAMP:

1. Lancer xampp control panel
2. Start MySQL
3. Start Shel
4. Se connecter à MariaDB
5. *Création la base de donnée « classbd »*

```
Dell@DESKTOP-D8D9ALB c:\xampp
# mysql -u root -p
Enter password: ****
Welcome to the MariaDB monitor
Your MariaDB connection id is
Server version: 10.4.19-Maria
Copyright (c) 2000, 2018, Ora
Type 'help;' or '\h' for help
MariaDB [(none)]>
```

```
MariaDB [(none)]> create database classbd;
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [(none)]> use classbd;
Database changed
MariaDB [classbd]> _
```

Connecter une base de données MySQL en Java

1. Télécharger et installer la base de données MySQL
2. Créer une base de données dans MySQL
Après avoir créé une base de données, souvenez-vous en car vous l'utiliserez dans la partie connectivité.
3. Téléchargez le pilote JDBC et incluez-le dans le chemin de classe
4. Connectez-vous avec MySQL
5. Tester la connexion JDBC

Connecter une base de données MySQL en Java

1. Télécharger et installer la base de données MySQL
2. Créer une base de données dans MySQL
3. Téléchargez le pilote JDBC et incluez-le dans le chemin de classe

Le pilote JDBC est un fichier JAR fourni par MySQL ; c'est un connecteur qui agit comme un pont entre les applications MySQL et Java.

Pour télécharger le fichier JAR visitez le site officiel de MySQL et placez les fichiers dans le dossier **lib** de votre projet java.

3. Téléchargez le pilote JDBC et placez-le dans le classpath

On va utiliser ECLIPSE:
Téléchargez le pilote JDBC:

Deux méthodes:

1. Projet Java
2. Projet Maven

1. Projet Java

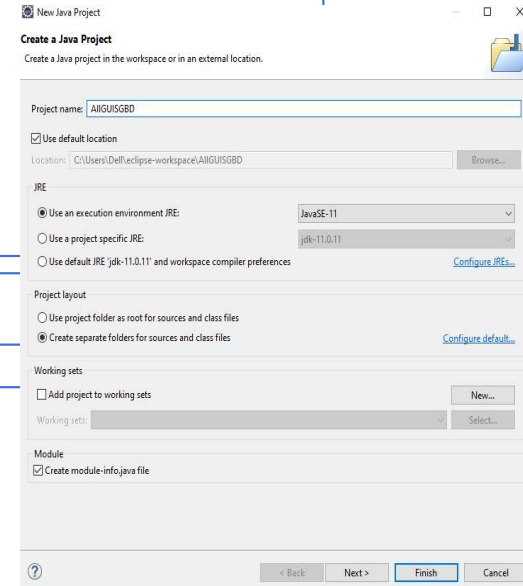
1. Créer un projet *File -> New -> Java Project*.

2. Télécharge le **jar** JDBC de MySQL:

<https://dev.mysql.com/downloads/connector/j/>

• Ou

<https://repo1.maven.org/maven2/mysql/mysql-connector-java/8.0.27/mysql-connector-java-8.0.27.jar>



3. Téléchargez le pilote JDBC et placez-le dans le classpath

1. Projet Java

1. **Créer un projet Java:** *File -> New -> Java Project.* (nom de votre projet **AppSWINGSDBD**)

2. **Télécharge le jar JDBC MySQL:**

<https://dev.mysql.com/downloads/connector/j/>

• **Ou**

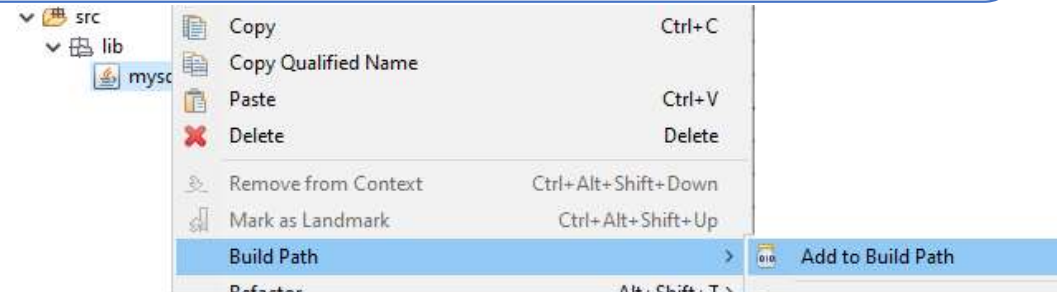
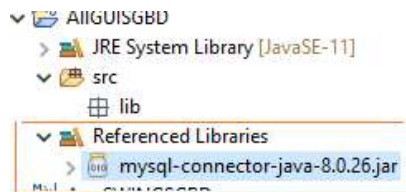
<https://repo1.maven.org/maven2/mysql/mysql-connector-java/8.0.27/mysql-connector-java-8.0.27.jar>

3. **Créer un dossier lib dans votre projet**

4. **Placer le `mysql-connector-java-***.jar` dans ce dossier lib**

5. **Placer le JDBC dans le classpath : Cliquer à Droite sur `mysql-connector-java-***.jar` sur et sélection **Build Path -> Add to Build Path****

(Une nouvelle librairie a été ajouté à notre projet: Referenced librairies)



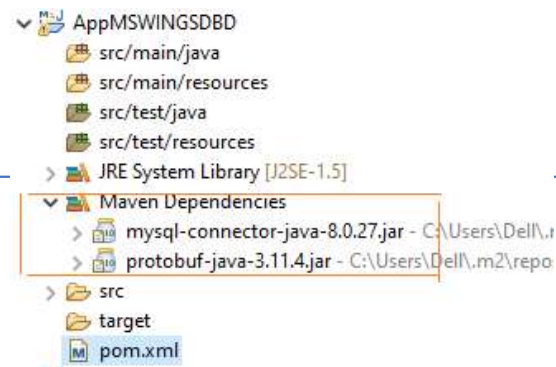
3. Téléchargez le pilote JDBC et placez-le dans le classpath

2. Projet Maven

1. Créer un projet Maven *File -> New -> Maven Project.* (nom de votre projet **AppMSWINGSDBD**)
2. Connecter à l'internet
3. Copier la dépendance dans le fichier **POM.xml** de votre projet de JDBC de MySQL:
<https://mvnrepository.com/artifact/mysql/mysql-connector-java/8.0.27>

```
<dependency>  
  <groupId>mysql</groupId>  
  <artifactId>mysql-connector-java</artifactId>  
  <version>8.0.27</version>  
</dependency>
```

(Une nouvelle librairie a été ajouté
à notre projet: Referenced librairies)



New Maven project

Configure project

Artifact

Group Id: AppSW

Artifact Id: AppMSWINGSDBD

Version: 0.0.1-SNAPSHOT

Packaging: jar

Name:

3. Téléchargez le pilote JDBC et placez-le dans le classpath

2. Projet Maven

pom.xml:

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>AppSW</groupId>
  <artifactId>AppMSWINGSDBD</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <dependency>
      <groupId>mysql</groupId>
      <artifactId>mysql-connector-java</artifactId>
      <version>8.0.27</version>
    </dependency>
  </dependencies>
</project>
```

4. Connectez-vous avec MySQL

Après avoir terminé les procédures ci-dessus, écrivez le code Java pour la connectivité.

Ici, nous avons utilisé **la méthode** `class.forName()` pour charger le **JDBC Driver**, que nous avons téléchargé depuis le site officiel de MySQL.

Pour MySQL:

```
DRIVER = "com.mysql.cj.jdbc.Driver"
```

La méthode `getConnection()` permet de passer la chaîne de connexion(URL, User, Mdp) :

Pour MySQL:

```
URL = "jdbc:mysql://localhost:3306/ ";
```

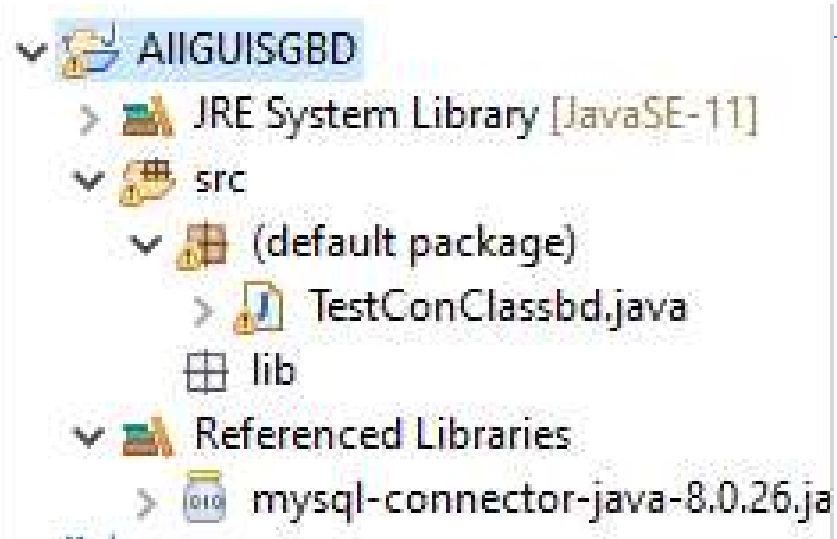
```
USER : "root";
```

```
Mot de passe= "";
```

Cette chaîne est utilisée pour authentifier l'utilisateur et fournir l'accès aux utilisateurs autorisés uniquement.

4. Connectez-vous avec MySQL

Créer un classe **TestConClassbd** dans votre projet:



4. Connectez-vous avec MySQL

```
import java.sql.Connection;
import java.sql.DriverManager;
public class TestConClassbd{
public static void main(String[] args) {
try {
    Class.forName("com.mysql.jdbc.Driver");
    Connection
    con=DriverManager.getConnection("jdbc:mysql://localhost:3306/classbd","root","");
    System.out.println("Connecter");
}
catch(Exception e)
{ System.out.println(e);
  System.err.println("Connecter");

}
}
}
```


5. Tester la connexion JDBC

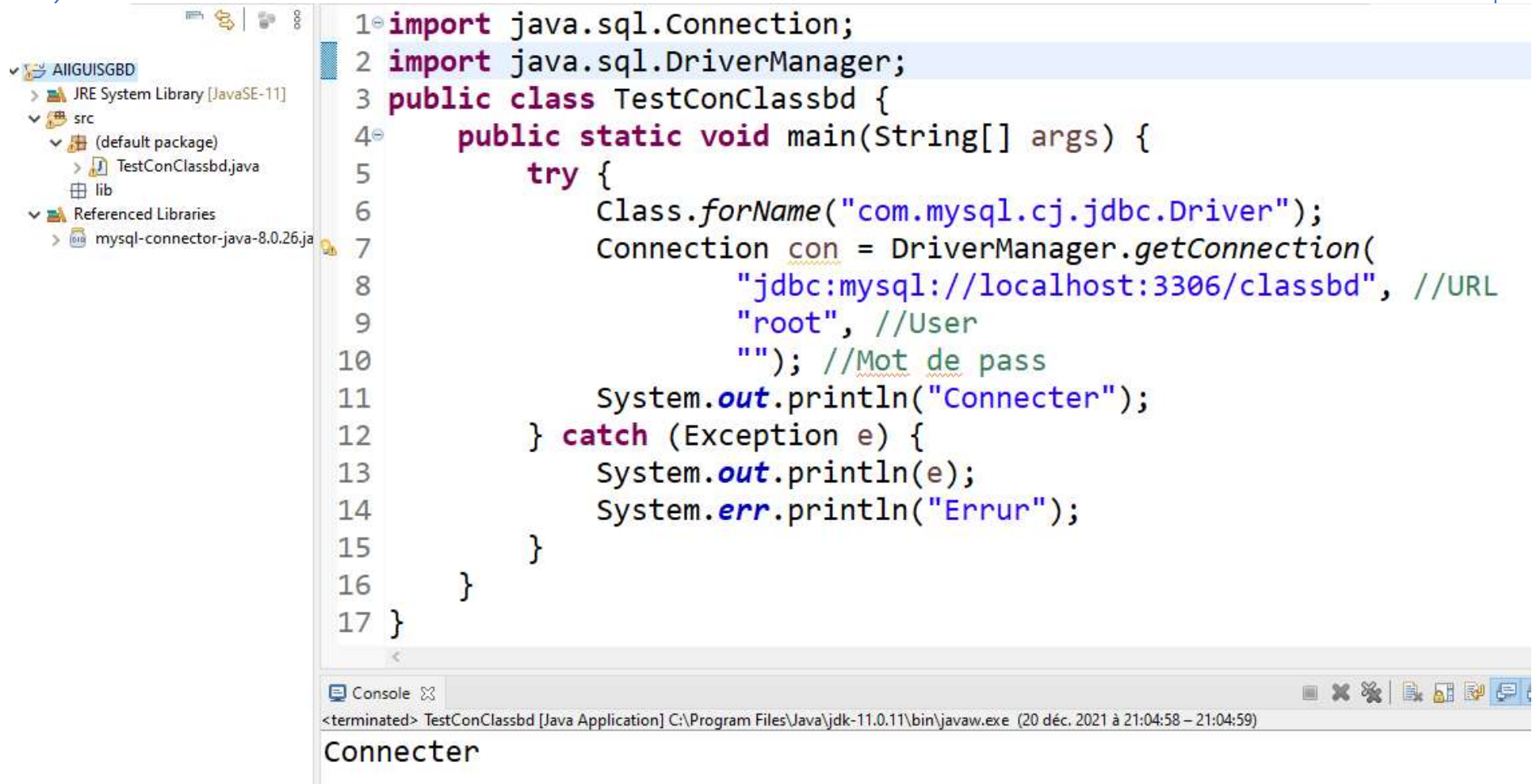
Tester la connexion JDBC

Après avoir écrit le code, il ne vous reste plus qu'à l'exécuter.

Si le code fonctionne correctement, vous obtiendrez la sortie suivante sur la console :

```
Connecter
```

5. Tester la connexion JDBC: Java Project



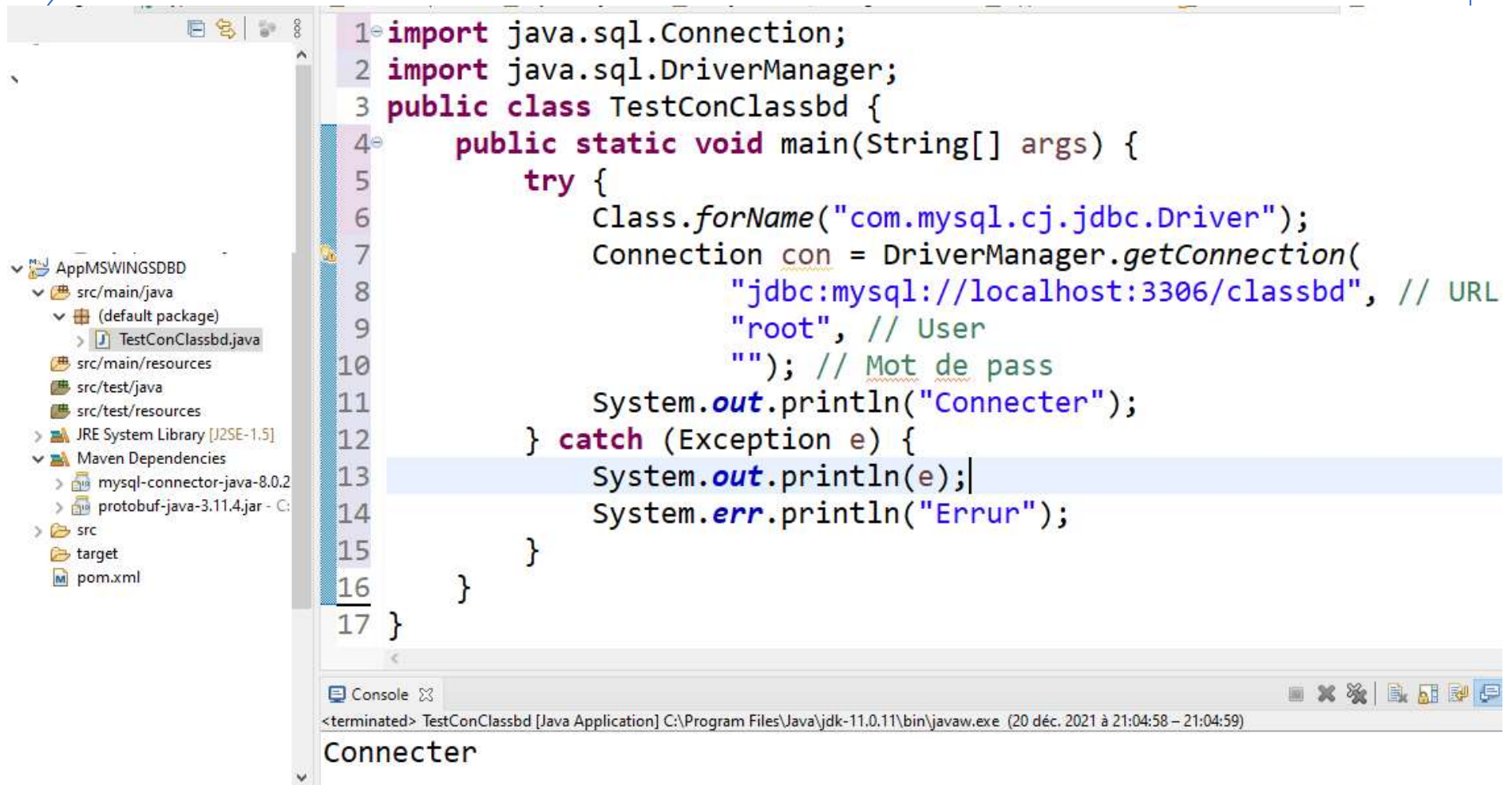
```
1 import java.sql.Connection;
2 import java.sql.DriverManager;
3 public class TestConClassbd {
4     public static void main(String[] args) {
5         try {
6             Class.forName("com.mysql.cj.jdbc.Driver");
7             Connection con = DriverManager.getConnection(
8                 "jdbc:mysql://localhost:3306/classbd", //URL
9                 "root", //User
10                ""); //Mot de pass
11             System.out.println("Connecter");
12         } catch (Exception e) {
13             System.out.println(e);
14             System.err.println("Errur");
15         }
16     }
17 }
```

Console

<terminated> TestConClassbd [Java Application] C:\Program Files\Java\jdk-11.0.11\bin\javaw.exe (20 déc. 2021 à 21:04:58 – 21:04:59)

Connecter

5. Tester la connexion JDBC: Maven Project



The screenshot displays an IDE interface. On the left, a project tree shows a Maven project named 'AppMSWINGSDBD' with a 'pom.xml' file and source directories. The main editor shows a Java file 'TestConClassbd.java' with the following code:

```
1 import java.sql.Connection;
2 import java.sql.DriverManager;
3 public class TestConClassbd {
4     public static void main(String[] args) {
5         try {
6             Class.forName("com.mysql.cj.jdbc.Driver");
7             Connection con = DriverManager.getConnection(
8                 "jdbc:mysql://localhost:3306/classbd", // URL
9                 "root", // User
10                ""); // Mot de pass
11             System.out.println("Connecter");
12         } catch (Exception e) {
13             System.out.println(e);
14             System.err.println("Error");
15         }
16     }
17 }
```

Below the code editor, the console window shows the output of the program:

```
<terminated> TestConClassbd [Java Application] C:\Program Files\Java\jdk-11.0.11\bin\javaw.exe (20 déc. 2021 à 21:04:58 - 21:04:59)
Connecter
```