

Tutoriel : Comment faire une connexion à sa base de données en JAVA



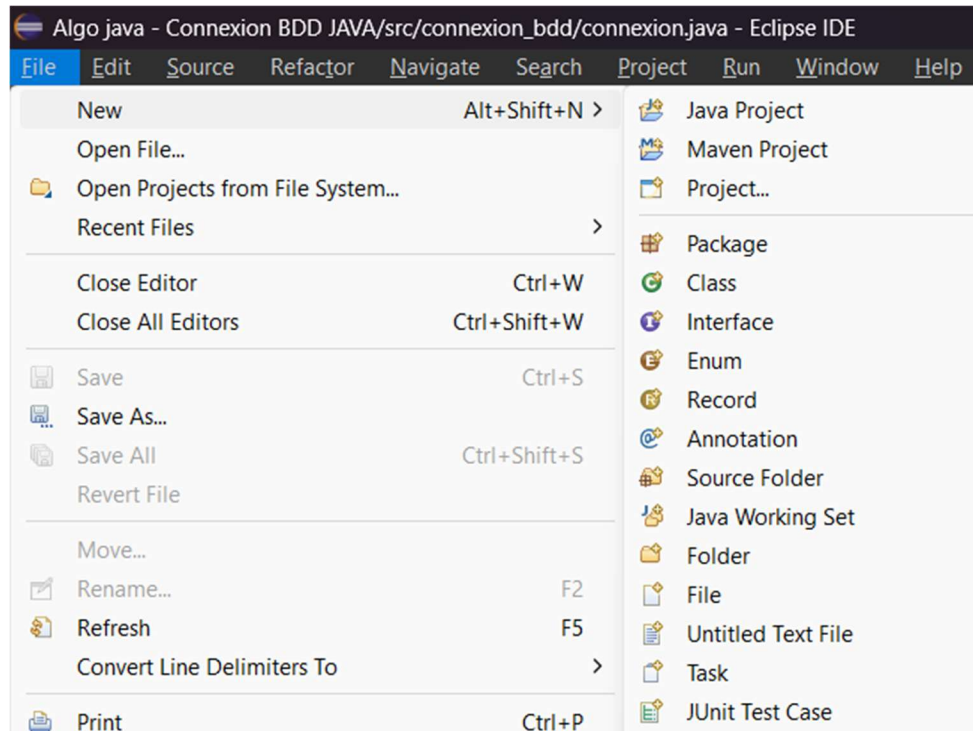
Pour pouvoir suivre ce dossier, vous devez au préalable avoir téléchargé [ECLIPSE IDE](#) ainsi que le serveur SQL [MAMP](#) ou [XAMPP](#) à votre convenance. Le but de ce tutoriel est de comprendre comment fonctionne la connexion à une base de données avec le langage JAVA.

N'allez pas voir la solution dans des vidéos ou en utilisant des IA. Le travail personnel vous permettra de pouvoir reproduire vos travaux et d'améliorer vos compétences.

Passons au tuto :

Étape 1 : Créer votre projet Eclipse et vos dossiers.

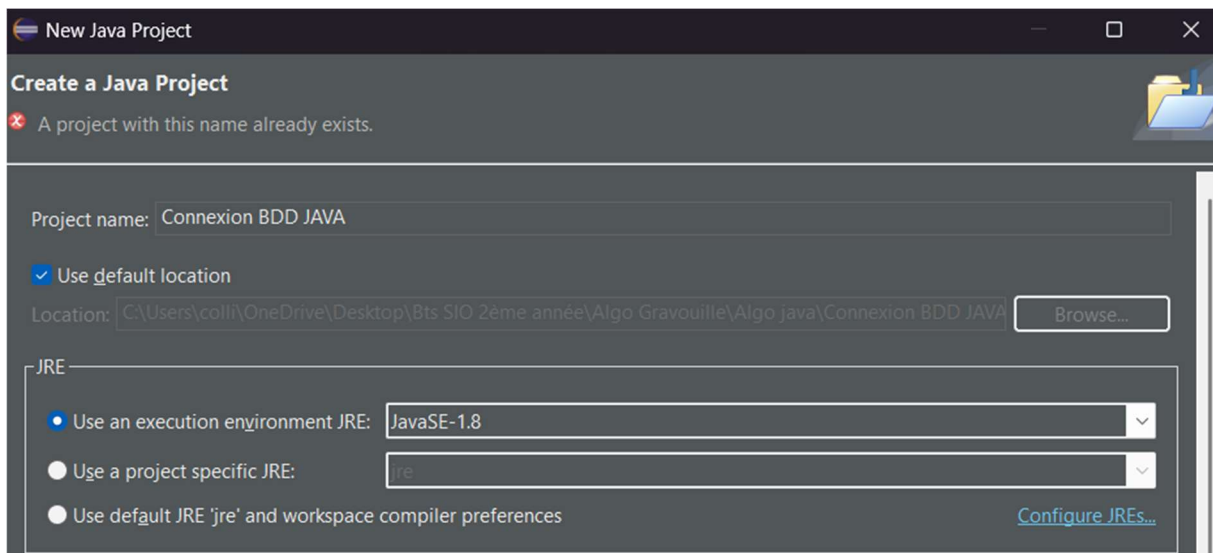
- 1) Ouvrez Eclipse
- 2) Dans le menu Fichier, sélectionnez New → Java Project



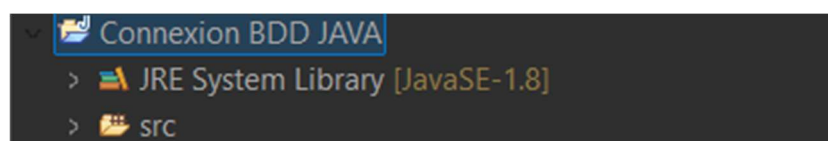
3) Donnez un nom à votre projet.

Vous pouvez nommer votre projet comme bon vous semble, pour ma part j'ai choisi Connexion BDD JAVA.

Vérifiez bien que vous utilisez JavaSE-1.8.



Cliquez ensuite sur le bouton Finish, pour créer le projet. Vous devriez à ce moment-là voir un projet Java vide qui ressemble à ceux-ci :

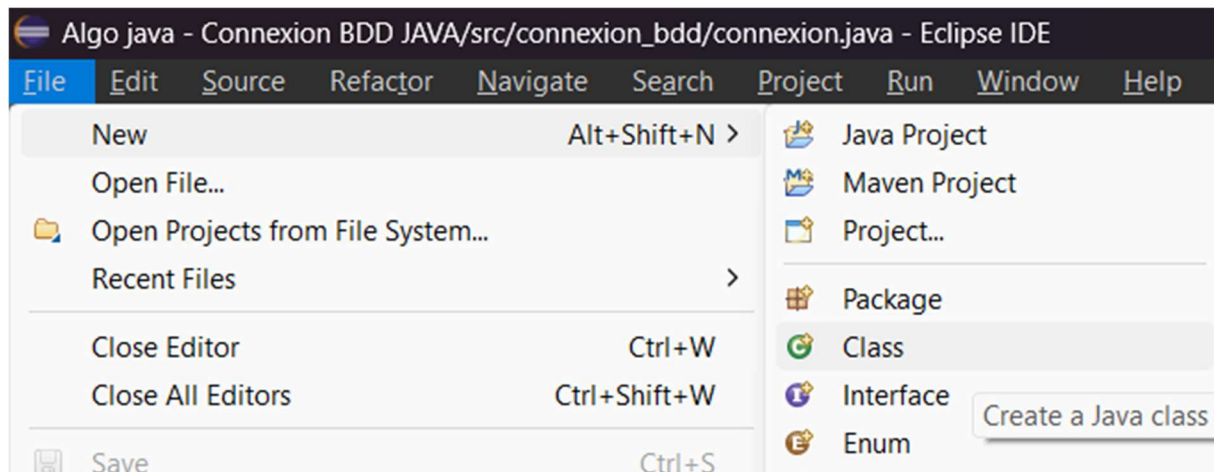


4) Créer un package et une classe Java

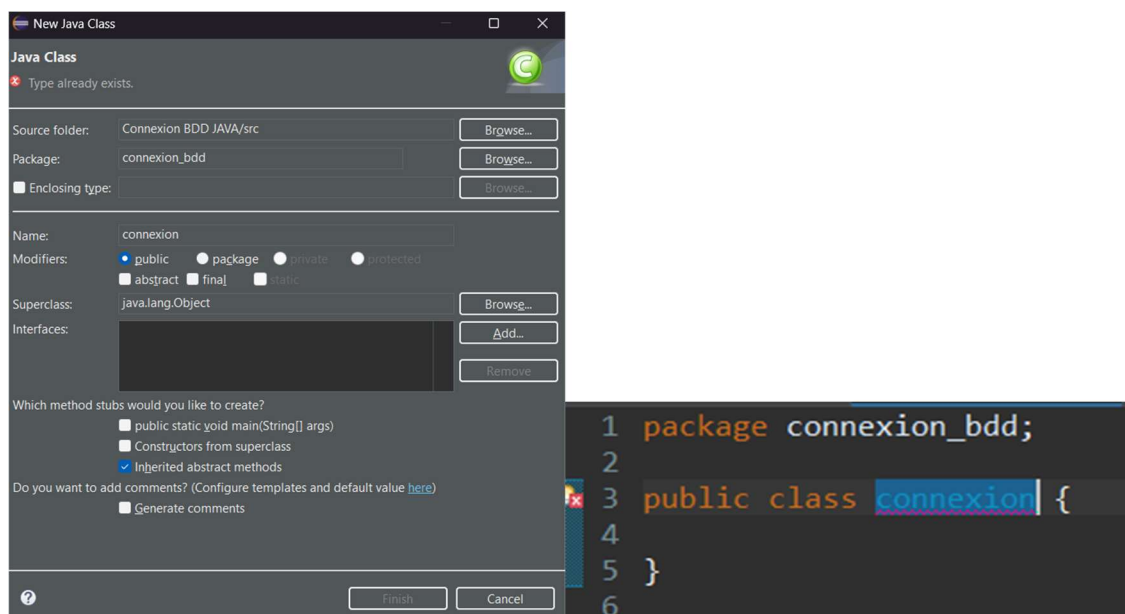
Choisissez depuis le menu **File -> New -> package**

et puis donnez un nom à votre package **connexion_bdd** par exemple

Choisissez ensuite depuis le menu **File -> New -> Class**



Donnez ensuite un nom à la classe **connexion** par exemple. Vous obtenez un projet de ce type :



Étape 2 : Téléchargement et installation du driver JDBC pour MySQL.

1) Téléchargement et installation du driver JDBC pour et installation du driver JDBC pour MySQL.

Pour pouvoir connecter Java avec mysql, il nécessaire, télécharger et installer « mysql connector » for Java. Ce dernier est téléchargeable depuis

l'adresse: <https://dev.mysql.com/downloads/file/?id=480091>

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

Login »
using my Oracle Web account

Sign Up »
for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

No thanks, just start my download.

ORACLE © 2024 Oracle

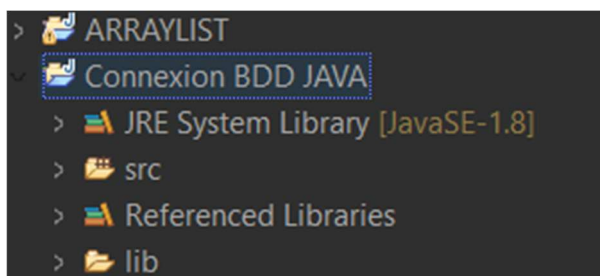
[Privacy / Do Not Sell My Info](#) | [Terms of Use](#) | [Trademark Policy](#) | [Préférences en matière de cookies](#)

Cette page, vous invite à créer un compte et à vous connecter. Mais vous pouvez surpasser cette étape en cliquant sur le lien en bas :

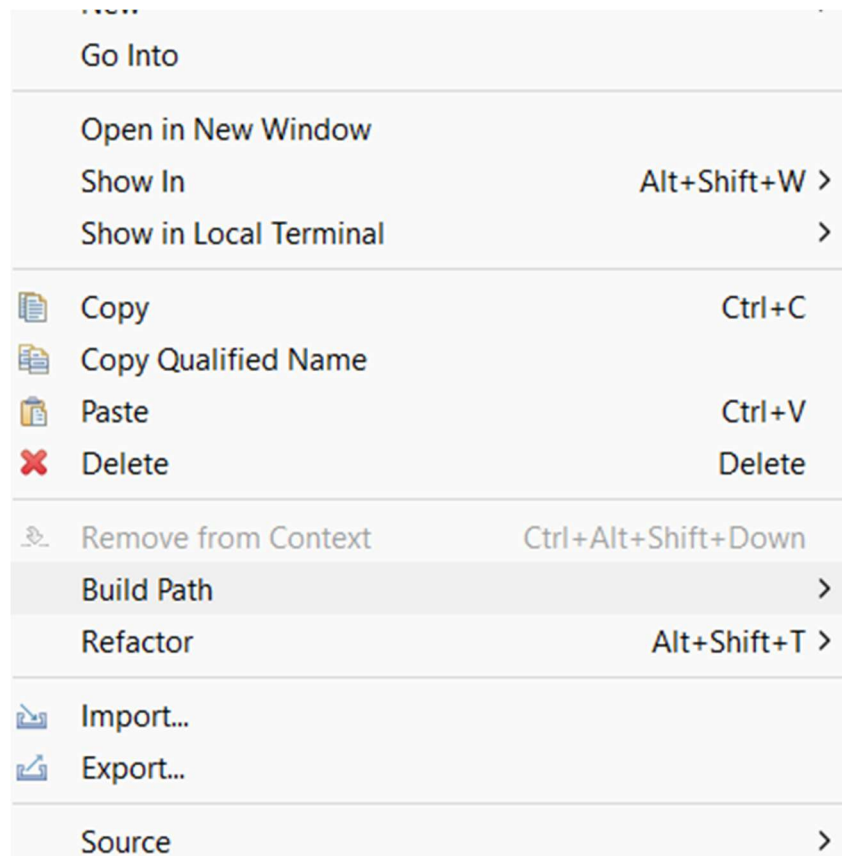
[No thanks, just start my download](#)

Après avoir téléchargé l'archive, décompressez la et vous allez trouver à l'intérieur un fichier jar nommé : **mysql-connector-java-5.1.47-bin.jar**

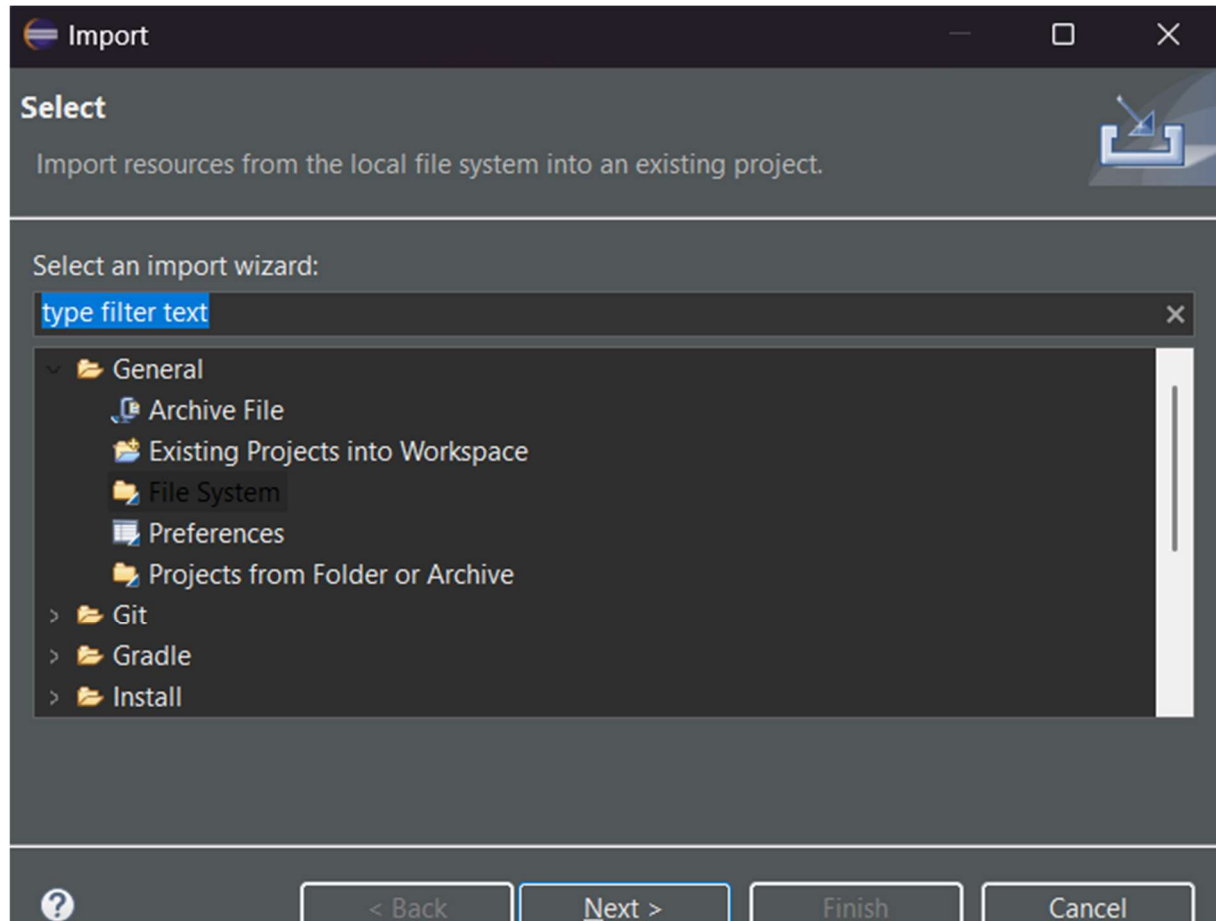
Sélectionnez votre projet sur Eclipse, et cliquez sur le menu **File ->New -> Folder**, pour créer un nouveau dossier, donnez le un nom **lib** par exemple :



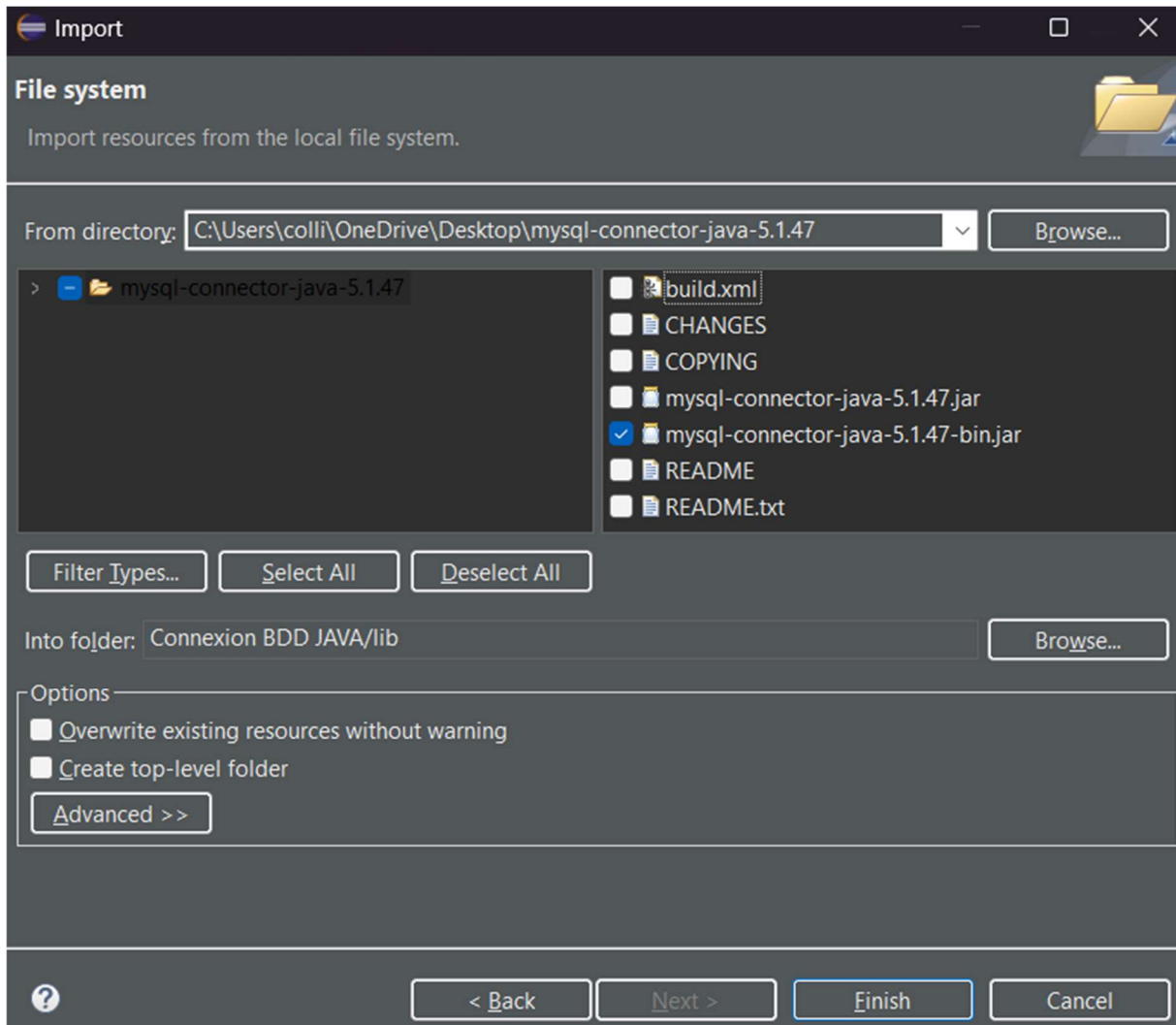
Maintenant à l'aide de la souris, faites un **clic droit sur le dossier lib** et pointez sur **le menu import**:



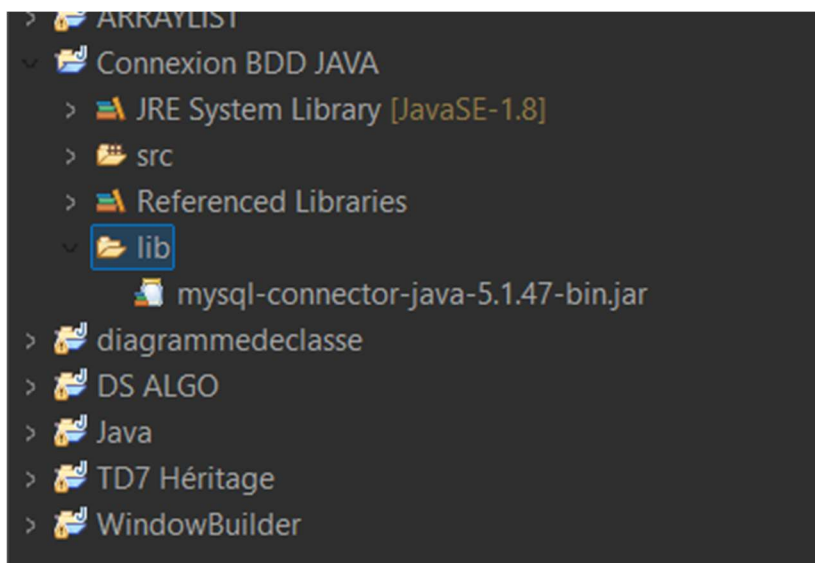
2) Ensuite depuis le menu General, choisissez le sous menu File System:



Continuez ensuite votre navigation jusqu'à ce que vous trouviez le fichier **mysql-connector-java-5.1.47-bin.jar** , sélectionnez le et cliquez sur le bouton **Finish** :

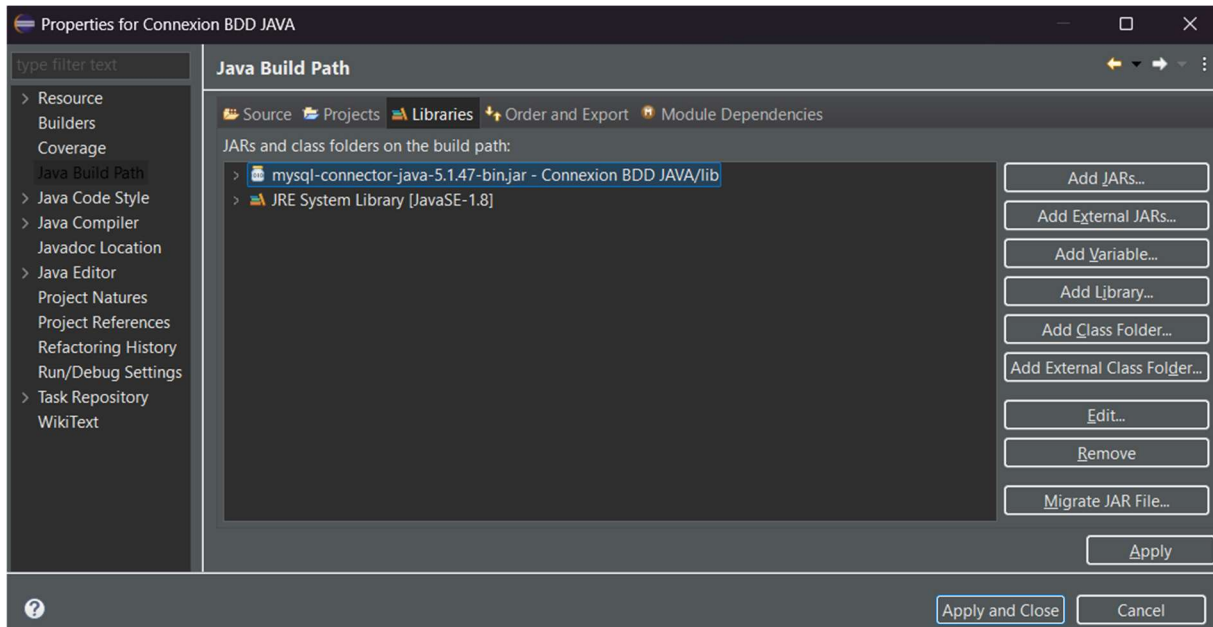


Le fichier est maintenant visible sur l'explorateur de fichier d'Eclipse :



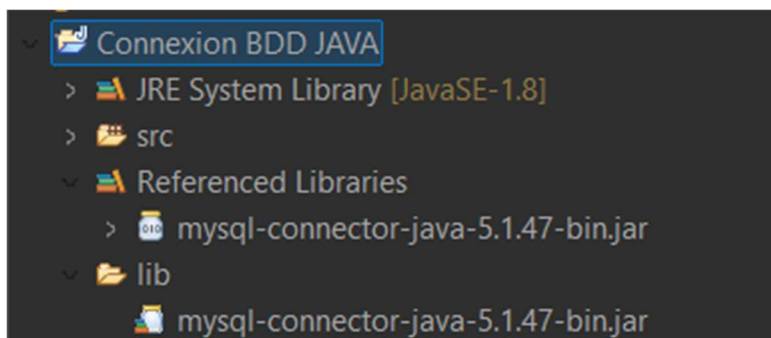
- 3) Veuillez ensuite informer Eclipse du fichier : **mysql-connector-java-5.1.47-bin.jar**, afin de le prendre en compte :

A l'aide du bouton droit de la souris sur le **Votre dossier principal** de votre projet Eclipse, cliquez sur **Properties** et choisissez ensuite le menu **Java Build Path**:



Sélectionnez ensuite l'archive jar qui se trouve dans le dossier lib du projet Eclipse.

Si les étapes sont correctement exécutés vous devriez voir l'archive jar dans **Referenced Libraries**:

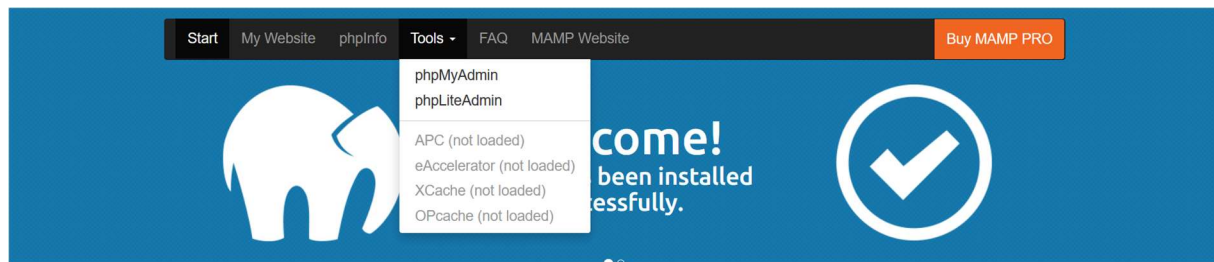


Étape 3 : Connexion de Java avec Mysql en utilisant le driver JDBC

- 1) Lecture et affichage des données.

En utilisant votre serveur local SQL (MAMP ou XAMPP) vous allez pouvoir créer votre base de donnée.

Ouvrez votre logiciel et cliquez sur votre site web. Ensuite allez dans TOOLS et cliquez sur phpmyadmin :



Vous allez ensuite pouvoir créer votre base de donnée. Nommez la connexionbddjava.

Après cela allez dans la section SQL de votre bdd et tapez ce code :

```
CREATE TABLE connexionbddjava.students ( id INT(10) NOT NULL AUTO_INCREMENT , nom
VARCHAR(50) NOT NULL , prenom VARCHAR(50) NOT NULL , dateN DATE NOT NULL , email
VARCHAR(50) NOT NULL , PRIMARY KEY (id)) ENGINE = InnoDB;
```

Votre table students devrait apparaître.

Retournez dans votre SQL et insérez maintenant des données avec ce code :

```
INSERT INTO `students` (`id`, `nom`, `prenom`, `dateN`, `email`) VALUES
(1, 'Dupont', 'Jean', '2000-05-15', 'jean.dupont@example.com'),
(2, 'Martin', 'Claire', '1998-08-22', 'claire.martin@example.com'),
(3, 'Bernard', 'Paul', '1995-12-05', 'paul.bernard@example.com'),
(4, 'Petit', 'Lucie', '2002-01-30', 'lucie.petit@example.com'),
(5, 'Rousseau', 'Julie', '1999-07-12', 'julie.rousseau@example.com'),
(6, 'Lemoine', 'Marc', '2001-11-17', 'marc.lemoine@example.com'),
(7, 'Fontaine', 'Sophie', '1997-04-10', 'sophie.fontaine@example.com'),
(8, 'Durand', 'Thomas', '2003-03-28', 'thomas.durand@example.com'),
(9, 'Morel', 'Emma', '2000-09-09', 'emma.morel@example.com'),
(10, 'Laurent', 'Nicolas', '1996-06-18', 'nicolas.laurent@example.com');
```

Votre table est créé et complétée.

2) Connexion à la base de donnée en JAVA.

Maintenant, tout est près, vous pouvez donc commencer à communiquer le langage Java avec les bases de données MySql. Pour cela veuillez suivre les étapes suivantes :

1- Charger le driver mysql

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


2- Créer un objet connection en utilisant la classe Connection:

```
Connection con=DriverManager.getConnection("jdbc:mysql://localhost/school","root","");
```

3 - Créer un état de connexion

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

4 - Créer et exécuter une requete sql en utilisant la classe ResultSet

```
ResultSet res=stm.executeQuery("select * from students");
```

5 - parcourir les données de la table res

Les données sont maintenant récupérées sur l'objet *res* pour accéder à la valeur d'un champ on utilise les méthodes `getInt(numéro de la colonne)`, `getDouble(numéro de la colonne)`, `getString(numéro de la colonne)`... selon le type du champ de la table (int, double, String...).

Exemple pour la 2ème colonne qui contient les noms du type *String* on utilise la syntaxe :

```
res.getString(2);
```

Et afin de parcourir les données successivement une après l'autre du début de la sélection jusqu'à la fin de la sélection, on utilise la boucle `while` :

```
while (res.next()){  
System.out.println("id :"+res.getInt(1)+" nom : "+res.getString(2)  
//Cet exemple affiche les données de la 1ère colonne et les données de la 2ème colonne}
```

6 - Fermer la connection

```
con.close();
```

Et voici le code final attendu et sécurisé :

```
1 connexion.java x
2 public class connexion {
3     public static void main(String[] args) {
4         try {
5
6             Class.forName("com.mysql.jdbc.Driver");
7
8
9             Connection con = DriverManager.getConnection("jdbc:mysql://localhost/connexionbddjava", "root", "root");
10
11
12             Statement stm = con.createStatement();
13
14             String query = "INSERT INTO students (nom, prenom, dateN, email) VALUES (?, ?, ?, ?)";
15             PreparedStatement pstmt = con.prepareStatement(query);
16             pstmt.setString(1, "Robert");
17             pstmt.setString(2, "Smith");
18             pstmt.setString(3, "2000-05-15");
19             pstmt.setString(4, "robert@gmail.com");
20             pstmt.executeUpdate();
21
22             ResultSet res = stm.executeQuery("SELECT * FROM students");
23
24             while (res.next()) {
25                 int id = res.getInt(1);
26                 String nom = res.getString(2);
27                 String prenom = res.getString(3);
28                 String dateN = res.getString(4);
29                 String email = res.getString(5);
30
31                 System.out.println(id + " | " + nom + " | " + prenom + " | " + dateN + " | " + email);
32             }
33
34             res.close();
35             stm.close();
36             con.close();
37
38         } catch (Exception e) {
39             e.printStackTrace();
40         }
41     }
42 }
```

!!! ATTENTION !!!

Pour la connexion à votre base de donnée (sur mon image la ligne 12) adaptez votre nom de base de donnée. De plus, le mot de passe diffère selon votre serveur SQL.

Pour MAMP :

```
Connection con = DriverManager.getConnection("jdbc:mysql://localhost/nomdevotrebdd",  
"root", "root");
```

Pour XAMPP :

```
Connection con = DriverManager.getConnection("jdbc:mysql://localhost/nomdevotrebdd",  
"root", "pas de mot de passe");
```


📄 Browse 📁 Structure 📄 SQL 🔍 Search 🧩 Insert 📄 Export 📄 Import 🛡️ Privileges 🔑

✅ Showing rows 0 - 10 (11 total, Query took 0.0010 seconds.)

`SELECT * FROM `students``

☐ Show all | Number of rows: 25 ▾ | Filter rows: | Sort by key: None ▾

+ Options
⬅️ ➡️

				id	nom	prenom	dateN	email
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	1	Dupont	Jean	2000-05-15	jean.dupont@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	2	Martin	Claire	1998-08-22	claire.martin@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	3	Bernard	Paul	1995-12-05	paul.bernard@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	4	Petit	Lucie	2002-01-30	lucie.petit@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	5	Rousseau	Julie	1999-07-12	julie.rousseau@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	6	Lemoine	Marc	2001-11-17	marc.lemoine@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	7	Fontaine	Sophie	1997-04-10	sophie.fontaine@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	8	Durand	Thomas	2003-03-28	thomas.durand@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	9	Morel	Emma	2000-09-09	emma.morel@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	10	Laurent	Nicolas	1996-06-18	nicolas.laurent@example.com
<input type="checkbox"/>	✎ Edit	📄 Copy	🗑 Delete	11	Robert	Smith	2000-05-15	robert@gmail.com

⬆️ ☐ Check all | With selected: ✎ Edit 📄 Copy 🗑 Delete 📄 Export

Félicitations !! Vous savez désormais faire un connexion locale de votre base de donnée en JAVA.