



Deuxième année Ingénierie Génie Logiciel

Développement Avancé

TP numéro 1 : Servlets, Base de données

(Enseignants de TP : Fatma. NAKBI)

Objectifs: Ce TP a pour but de vous familiariser avec l'API Servlets, vous aborderez ainsi les points suivants :

- Créer un projet d'application Web avec l'IDE Netbeans 8.1.
- Ecrire une Servlet simple renvoyant une page HTML.
- Récupérer les paramètres de la requête et renvoyer ces paramètres dans une page HTML.
- Connexion et manipulation de la base de données à partir d'une Servlet.

Partie 1 : Les Servlets

1-Définition et fonctionnement

- Mécanisme de traitement de requêtes/réponses
- **Servlets HTTP :**
 - ✓ invoquées par des requêtes http
 - ✓ déployables sur un serveur Web
 - ✓ leur résultat est un flux HTML envoyé au navigateur

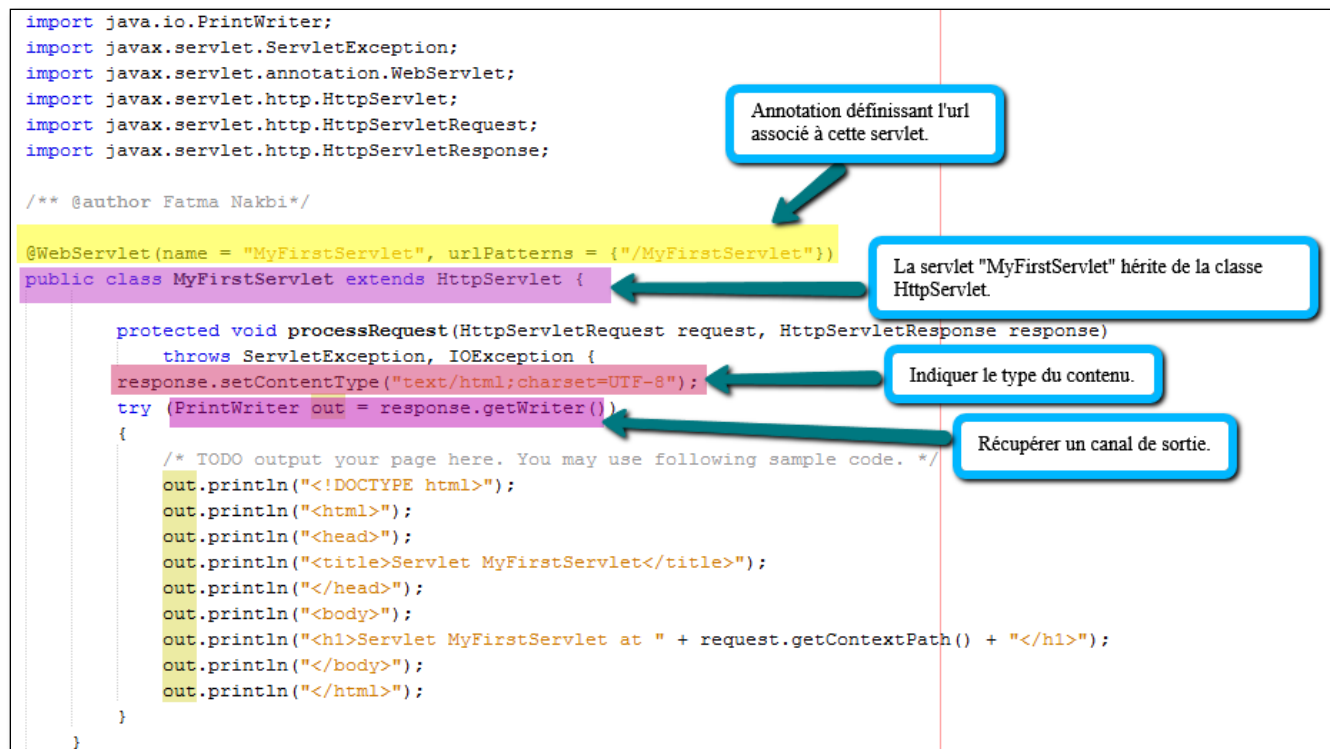


Figure 1.1 : Un exemple de code Java d'une Servlet.

2-Récupération les informations de la requête

Pour récupérer les données de la requête envoyée par le client (formulaire HTML), on utilise la méthode de l'objet request reçu en paramètre de doGet ou doPost :

String getParameter(String nomComposantFormulaire) ;

Exemple:

En code de la page HTML :

```
<input id="nom" name="nom" type="text" />
```

En code de la classe Servlet :

```
String nom = request.getParameter("nom");
```

Pour envoyer une réponse de servlet vers la page HTML ou autre Servlet :

- On positionne le type MIME par : **void setContentType(String type)**

- On récupère le canal de sortie texte par : `PrintWriter getWriter()` sur `response` l'objet `response`.

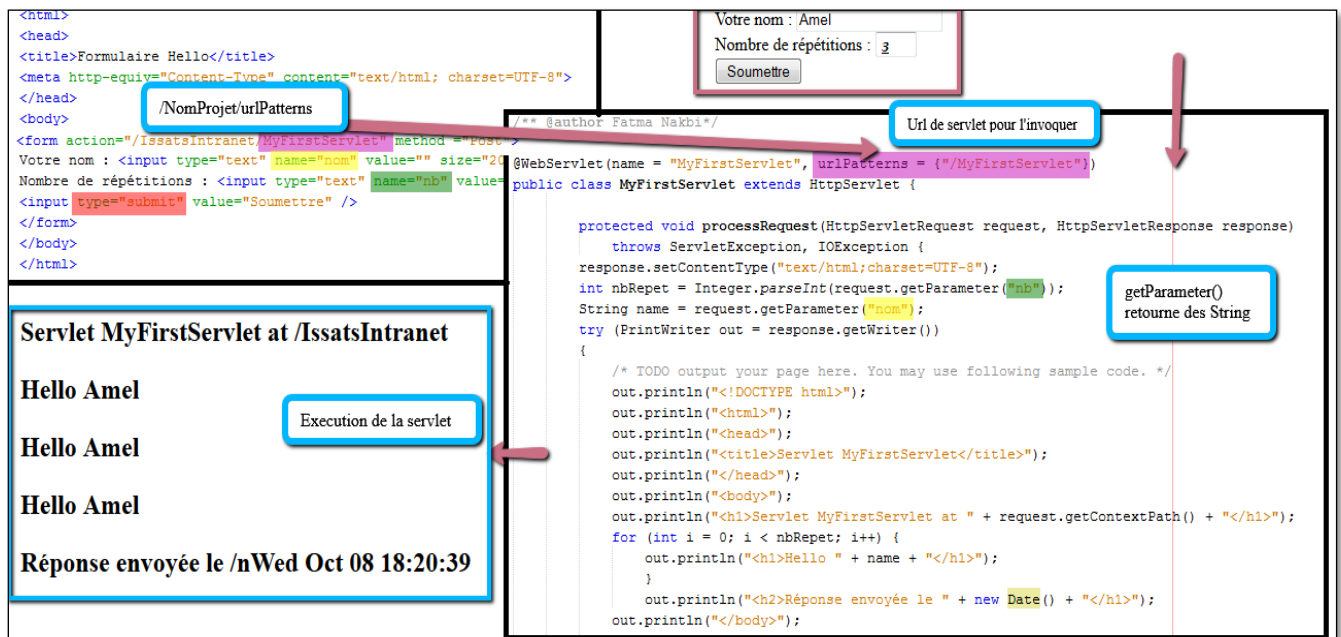


Figure 2.1 : Un exemple de récupération des paramètres à partir d'un formulaire.

3-Exercices :

Exercice 1 : Création d'une Servlet qui récupère les champs d'un formulaire de connexion d'étudiant.

Créer un dossier nommé « **etudiants** » dans le dossier web de l'application Web Java déjà crée en exercice 1 nommée « **IssatIntranet** ».

Créer un fichier HTML nommé « **connexion.html** » dans le dossier « **etudiants** ». Les noms des champs inputs de formulaire de connexion sont « **login** » et « **motDePasse** ». L'URL de l'action du formulaire est « **/IssatIntranet/etudiants/Connexion** ». L'affichage souhaité semble à cette interface à la figure 3.1 suivante.

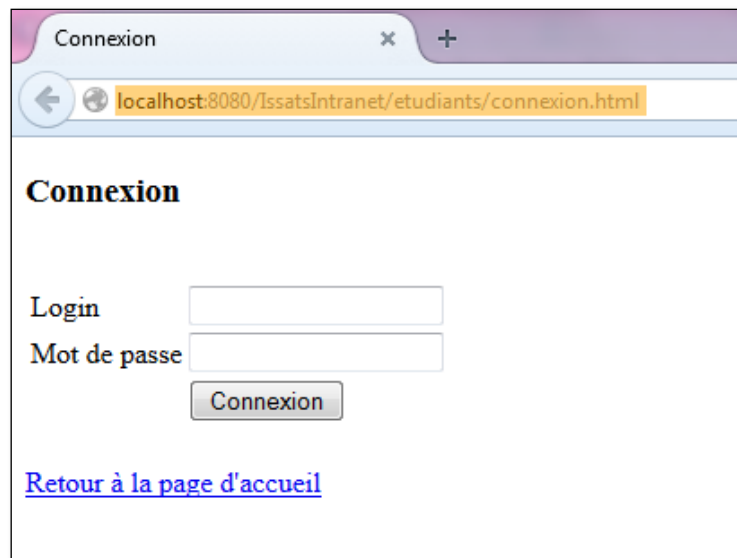


Figure 3.1 : interface de connexion en HTML.

Créer un package java nommé « **etudiants** » dans le projet Web « **IssatsIntranet** ».

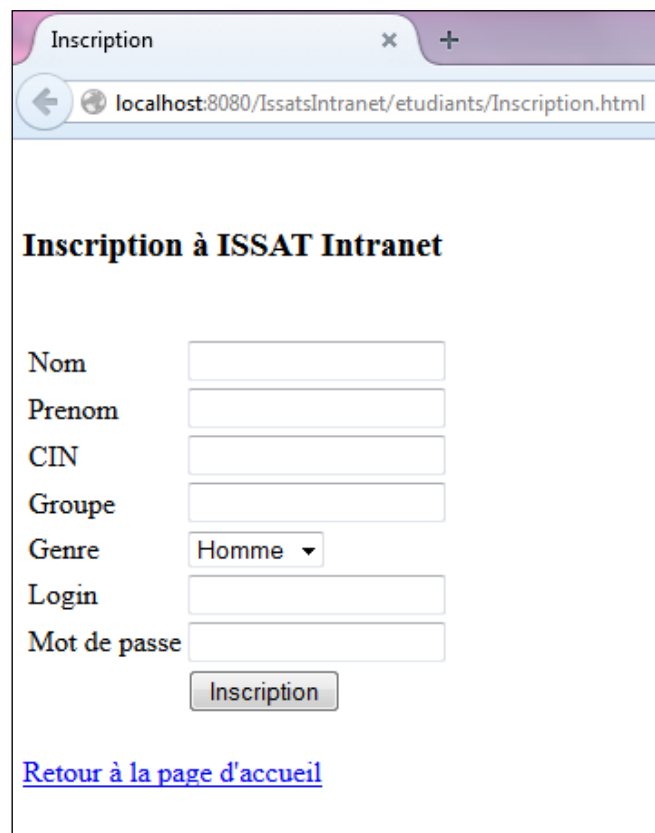
Créer une Servlet nommée « **Connexion.java** » dans le package « **etudiants** ». La servlet « Connexion » permet **de récupérer le contenu** des champs de formulaire envoyés de la page HTML connexion.html, puis **d'afficher les valeurs de login et mot de passe**. L'affichage souhaité semble à cette interface à la figure 3.2 à la page suivante.



Figure 3.2 : Résultat souhaité de la Servlet Connexion.

Exercice 2 : Création d'une Servlet qui récupère les champs d'un formulaire d'inscription d'étudiant.

Créer un fichier HTML nommé « **inscription.html** » dans le dossier « **etudiants** » dans le dossier web. Les noms des champs inputs de formulaire d'inscription sont « **nom** », « **prenom** », « **cin** », « **groupe** », « **genre** » { **Homme**, **Femme** }, « **login** » et « **motDePasse** ». L'URL de l'action du formulaire est « **/IssatIntranet/etudiants/Inscription** ». L'affichage souhaité semble à cette interface à la figure 3.3 suivante.



The screenshot shows a web browser window with the title 'Inscription'. The address bar displays 'localhost:8080/IssatsIntranet/etudiants/Inscription.html'. The main content area features the heading 'Inscription à ISSAT Intranet'. Below the heading, there is a form with the following fields: 'Nom' (text input), 'Prenom' (text input), 'CIN' (text input), 'Groupe' (text input), 'Genre' (dropdown menu with 'Homme' selected), 'Login' (text input), and 'Mot de passe' (text input). A button labeled 'Inscription' is positioned below the 'Mot de passe' field. At the bottom of the form, there is a blue underlined link that reads 'Retour à la page d'accueil'.

Figure 3.3 : interface d'inscription en HTML.

Créer une Servlet nommée « **Inscription.java** » dans le package « **etudiants** ». La servlet « **Inscription** » permet **de récupérer le contenu des champs de formulaire** envoyés de la page HTML inscription.html, puis **d'afficher les valeurs** de nom, prénom, CIN, groupe, genre, login et mot de passe. L'affichage souhaité semble à cette interface à la figure 3.5 à la page suivante.

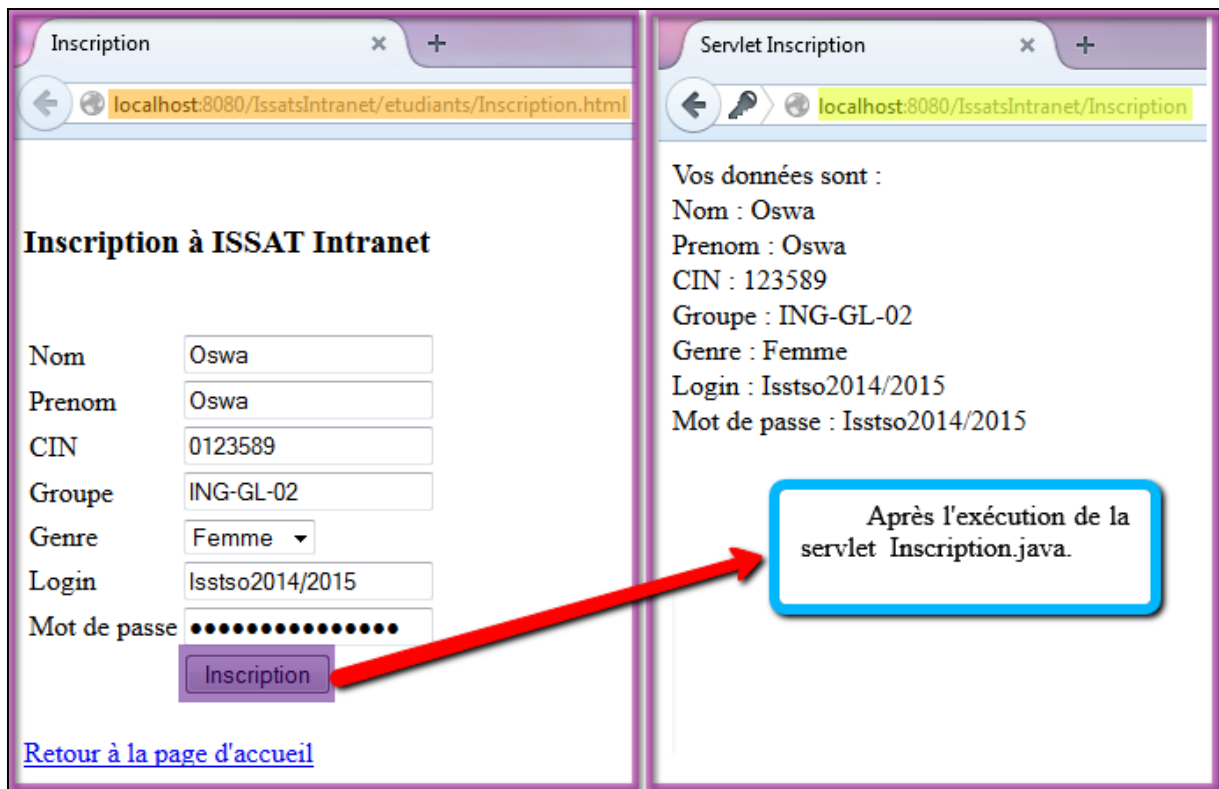


Figure 2.5 : Résultat souhaité de la Servlet Inscription.

La structure de notre application Web est présentée à la figure 2.5.

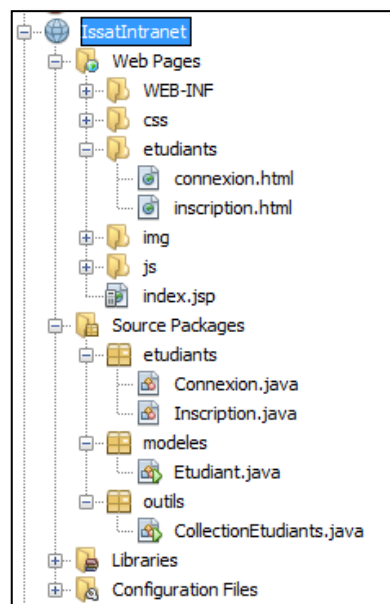


Figure 3.5 : Structure de projet Web IssatIntranet sous Netbeans 8.1.

Partie 2 : Servlets et base de données

Partie MySQL : Installation, création de base et configuration du driver

1. Installer Workbench de MySQL 5.x.

2. Création de la base:

1. créer une base de données nommée « issat_intranet » sous l'interface graphique de MySQL Workbench (voir figure 1.1).

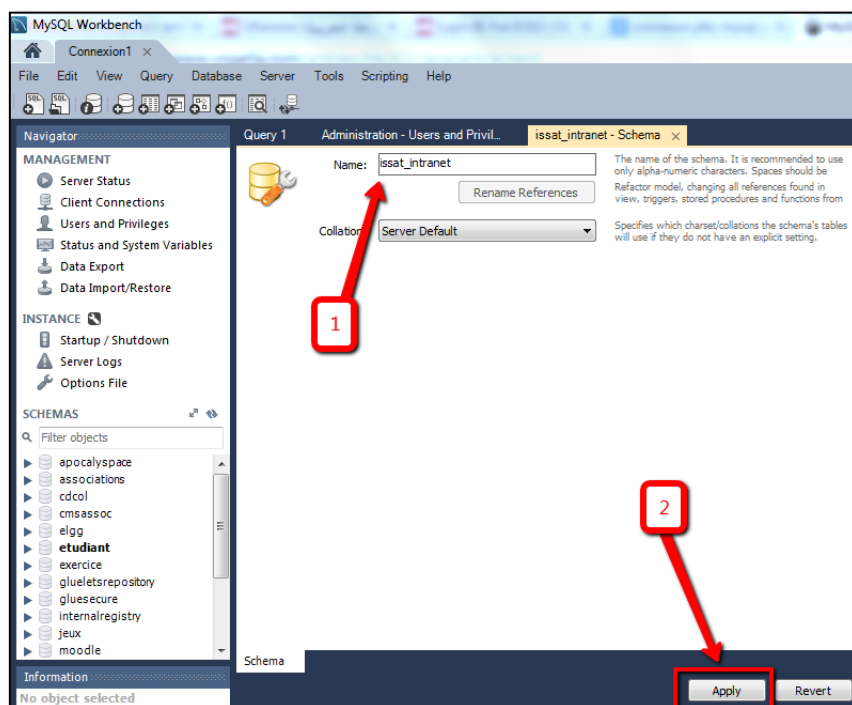


Figure 1.1 : création d'une base de données sur MySQL Workbench.

2. Copier le script SQL suivant à la fenêtre qui semble à celle de la figure 2.1

```
use issat_intranet;
create table etudiants
(
  cin int primary key,
  nom varchar(30) not null,
  prenom varchar(30) not null,
  groupe varchar(30) null,
  genre tinyint(1) null default 0,
  login varchar(30) unique,
  motDePasse varchar(255) not null
);
```

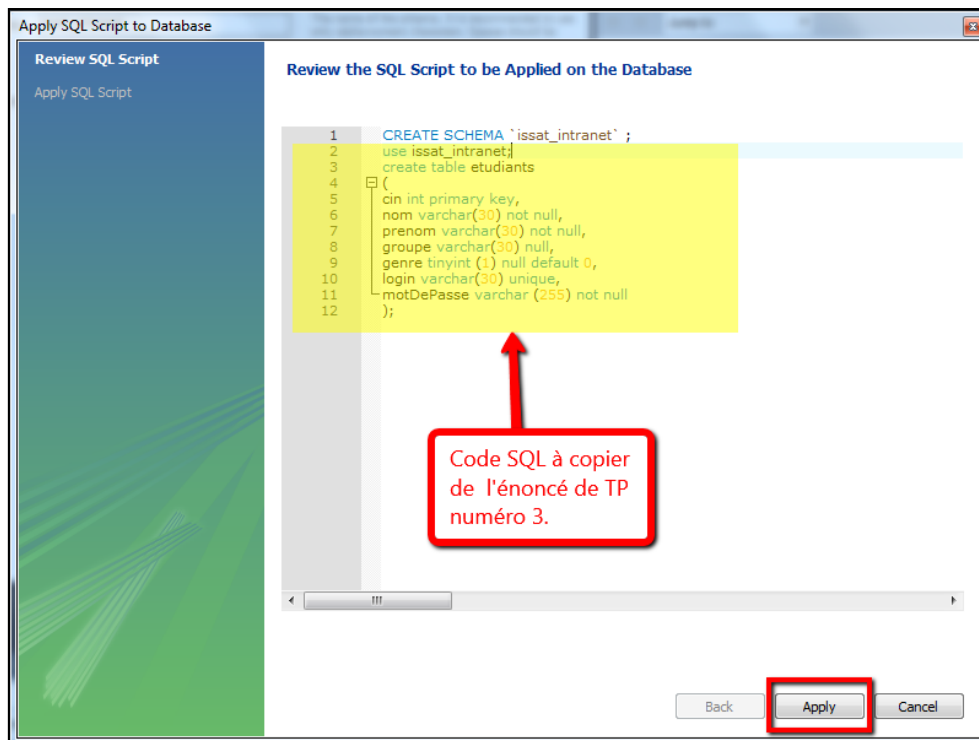


Figure 2.1 : Ajout de script de création de la table « etudiants ».

3. Valider le script copié, il faut avoir un résultat avec succès comme le résultat de la figure 3.1.

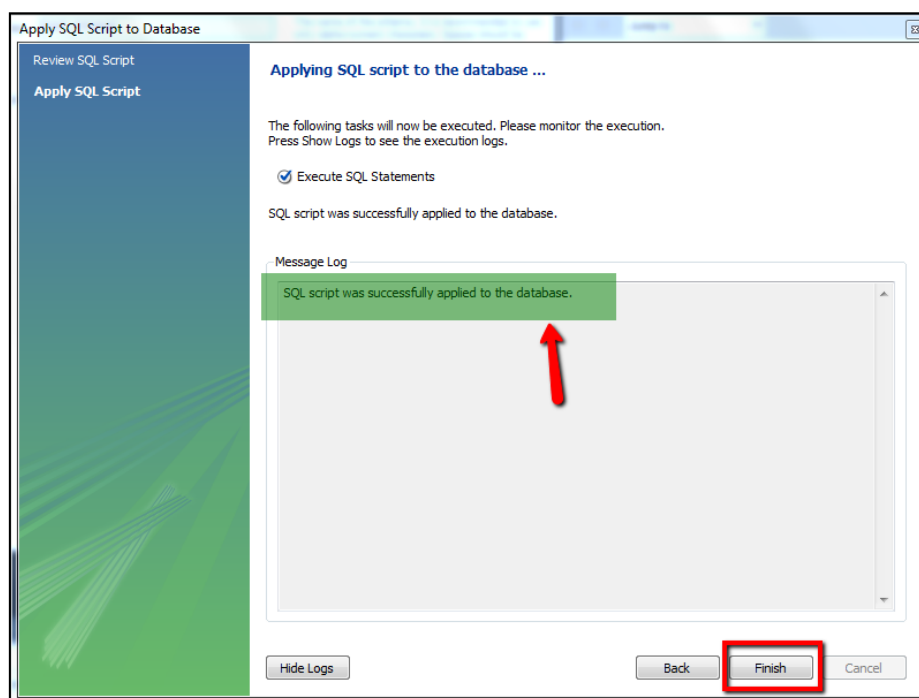


Figure 3.1 : Code SQL exécuté avec succès.

4. Vérifier que la base et la table « étudiants » sont bien créés (figure 4.1).

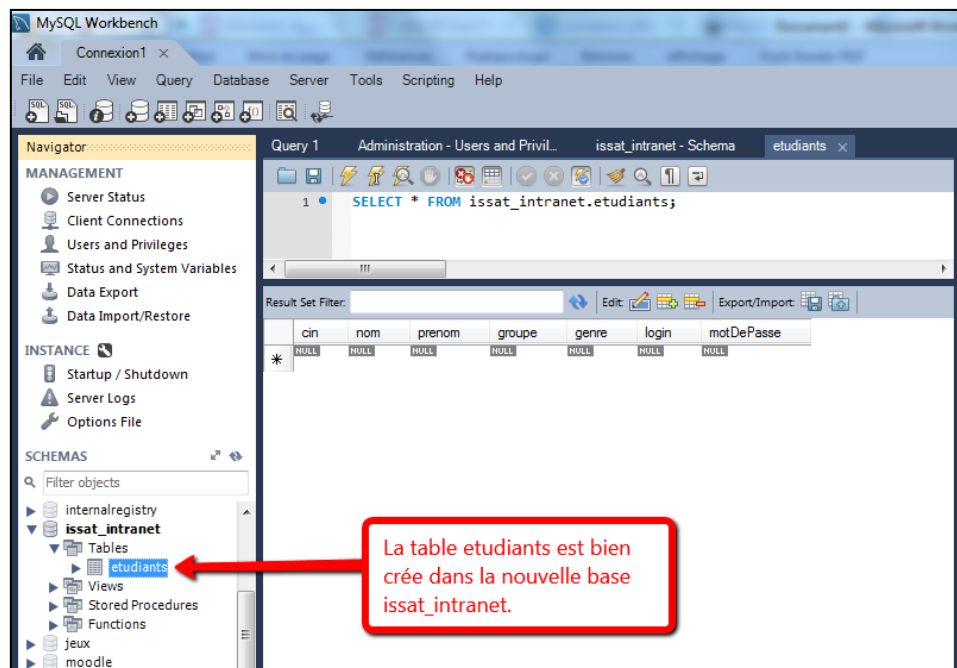


Figure 4.1 : L'interface de la table « étudiants ».

5. configuration de Driver JDBC sous Netbeans :

Ajouter le fichier .jar **mysql-connector-java-5.0.8-bin.jar** à la liste des bibliothèques de projet « IssatIntranet ».

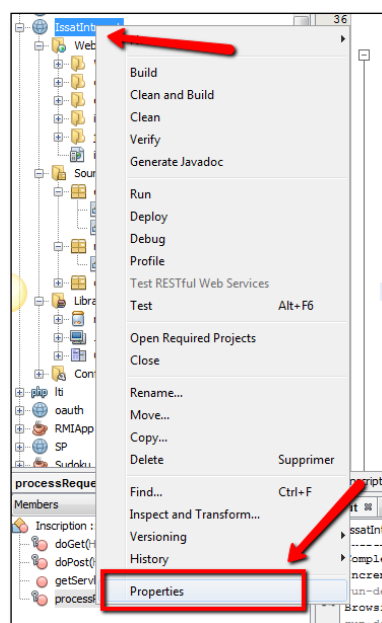


Figure 5.1 : Propriété de projet « IssatIntranet ».

Ajouter le JAR en cliquant le bouton « Add Jar/Folder »

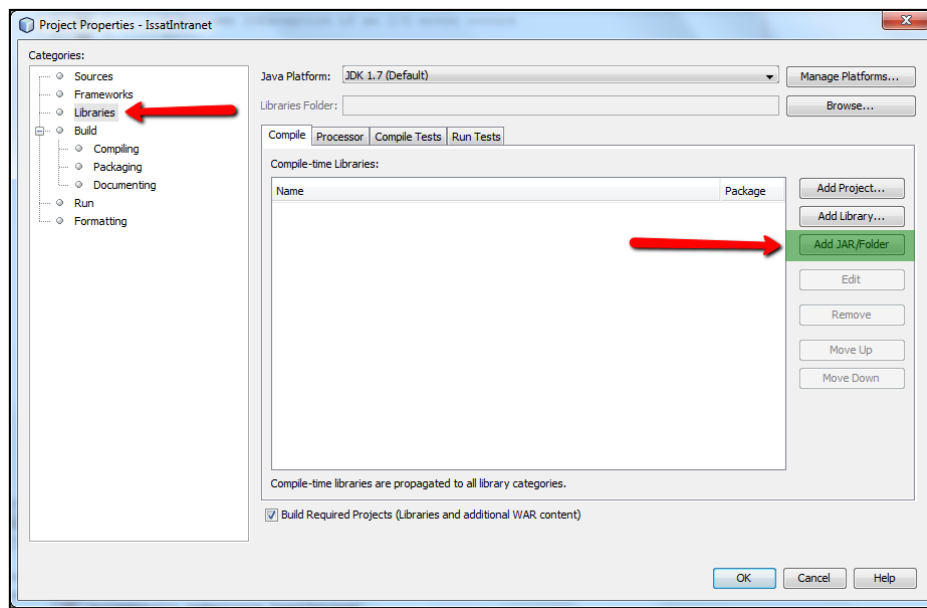


Figure 5.2 : Ajout de fichier JAR (partie 1).

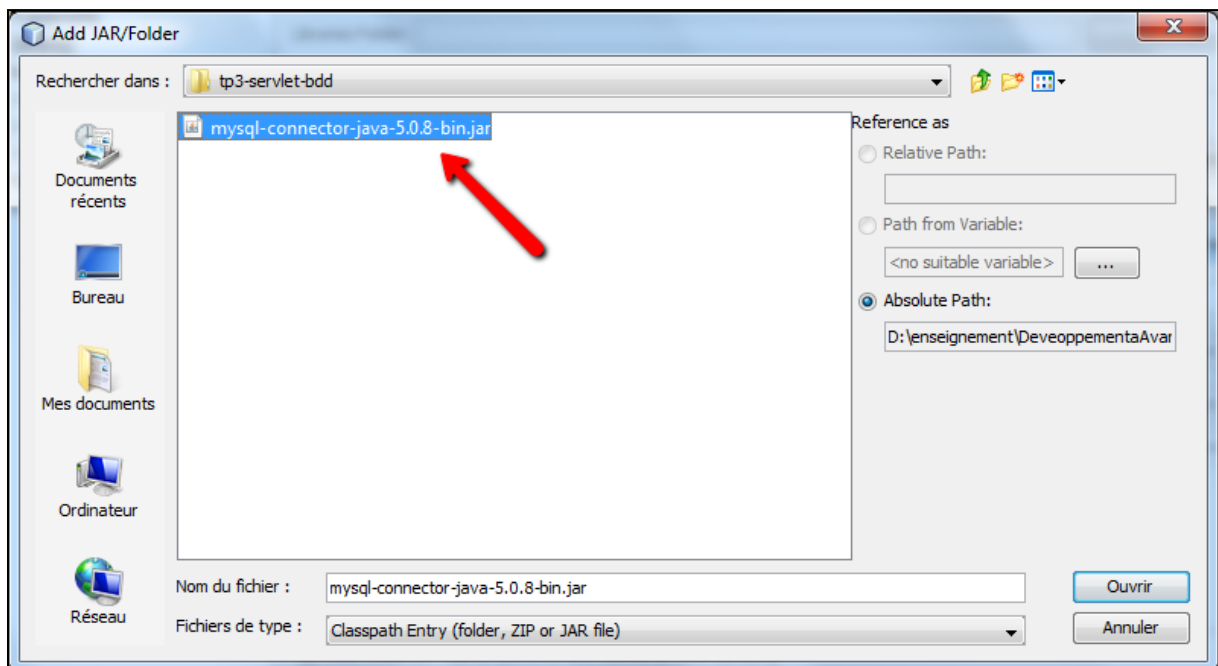


Figure 5.3 : Ajout de fichier JAR (partie 2).

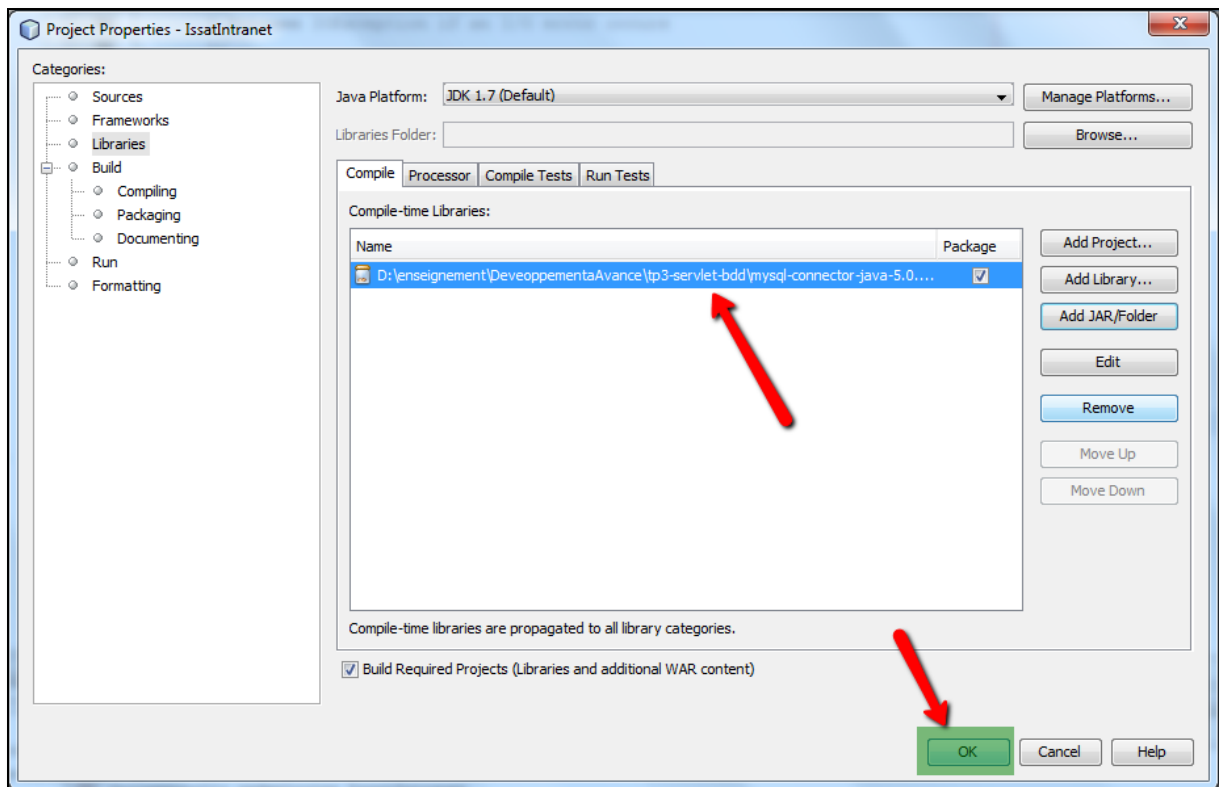


Figure 5.4 : Ajout de fichier JAR (partie 3).

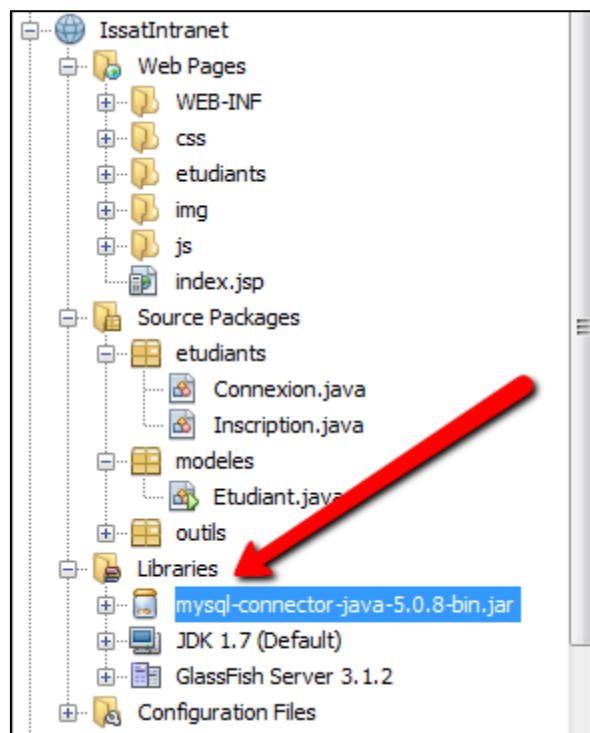


Figure 5.5 : Structure de projet « IssatIntranet ».

Partie 3 : Exercice numéro 1 :

1. Créer une méthode nommée « sauvegarder » dans la classe « Etudiant » qui permet de sauvegarder un étudiant dans la base. La signature de la méthode est la suivante :

```
public Boolean sauvegarder() ;
```

2. Appeler alors cette méthode dans la Servlet Inscription pour enregistrer l'étudiant à la base de données. Tester si l'étudiant est enregistré ou non via la variable de retour booléen de la méthode sauvegarder().

Partie 4 : Exercice numéro 2 :

1. Créer une méthode statique dans la classe Etudiant nommée connexion qui prend en paramètre un login et un mot de passe. Cette méthode permet de consulter la base de données et retourne un Etudiant qui possède le login et le mot de passe données, s'il n'existe pas elle retourne un étudiant vide. La signature de la méthode est :

```
public static Etudiant connexion(String login, String motDePasse);
```

2. Appeler la méthode connexion dans la Servlet Connexion pour consulter la base et retourner l'Etudiant correspondant. Afficher le nom et le prénom de l'étudiant dans la servlet Connexion en cas de réussite de connexion, affiché « connexion impossible » en cas d'échec.

Annexe 1 : Connexion à une base MySQL en Java.

Si la base de données est nommée « issat_intranet », l'utilisateur MySQL est « root » et le mot de passe est « root » alors le code de connexion en utilisant le driver JDBC est :

```
1 Class.forName("com.mysql.jdbc.Driver");  
2 Connection conn = DriverManager.getConnection(  
3 "jdbc:mysql://localhost/issat_intranet?"  
4 + "user=root&password=root");
```

Figure 1 : Connexion à une base MySQL avec un driver JDBC.

Annexe 2 : Une requête SELECT.

Pour un SELECT en cas des lignes multiple en résultat :

```
1 Statement st = conn.createStatement();
2 ResultSet r=st.executeQuery("SELECT * FROM issat_intranet.etudiants; ");
```

Parcourir de résultat d'un SELECT avec plusieurs lignes en résultat :

```
4 while (r.next())
5 {
6     String lenom = r.getString("nom");
7     // votre traitement...
8 }
```

Parcourir de résultat d'un SELECT qui retourne un seul enregistrement ou ligne :

```
10 if(r.first())
11 {
12     String lenom = r.getString("nom");
13     // votre traitement...
14 }
```

Annexe 3 : Une requête INSERT ou UPDATE.

En cas de sauvegarde d'un nouveau enregistrement dans la table « etudiants »

```
1 PreparedStatement reqSauvegarder = conn.prepareStatement(
2     "insert into issat_intranet.etudiants values (?, ?, ?, ?, ?, ?, ?)");
3     reqSauvegarder.setInt(1, 10001000);
4     reqSauvegarder.setString(2, "unNom");
5     reqSauvegarder.setString(3, "unPrenom");
6     reqSauvegarder.setString(4, "groupe1");
7     reqSauvegarder.setBoolean(5, true);
8     reqSauvegarder.setString(6, "unLogin");
9     reqSauvegarder.setString(7, "unMotDePasse");
10    Boolean ok = reqSauvegarder.executeUpdate()==0;
11    connect.close();
```