Université Abdelmalek Essaadi

Faculté des Sciences et Techniques - Tanger

Département Génie Informatique

**Filière :**

**« Logiciels et systèmes intelligents »**

**LSI**

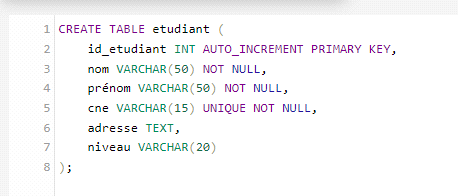
|  |
| --- |
| Lab 3 : Application JEE Distribuée avec EJB |

|  |  |
| --- | --- |
| **Réalisé par :**  Mohamed Amine BAHASSOU | **Encadré par :**  Prof. Lotfi EL AACHAK |

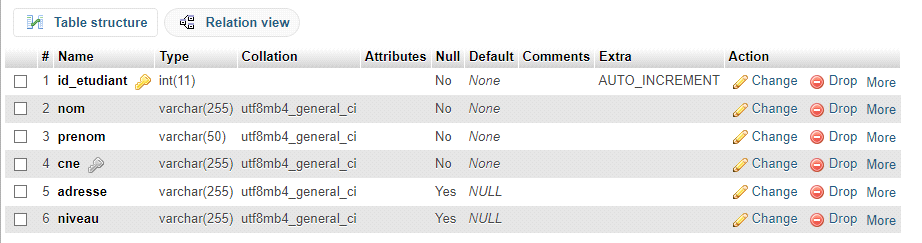
**Etape 1 : Création du base du Données :**



Après on va ajouter la table étudiant avec la commande sql :



Pour avoir une table comme ca :



**Etape 2 : Création du projet EJB Provider**

**On a crée une projet EJB dans intellij avec les depedencies suivants :**



Dans ce projet il nécessaire d’implementer EJB et servlet et aussi hibernate et mysqlconnector pour la connection au base du données

Et voici la persistence utilisé dans ce projet :



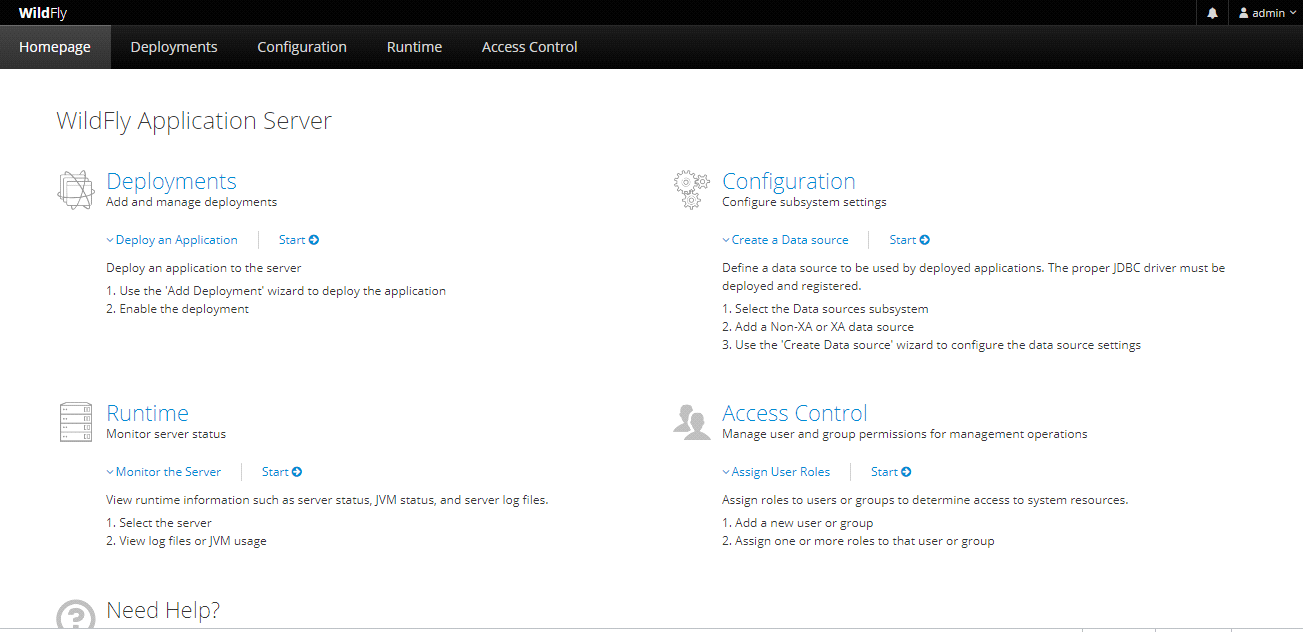
On a MySQLDS comme data source avec le nom du persistence ‘’Etudiant’’ aussi il faut connecter cette datasource dans WildFly pour qu’il connu tout d’abord il faut initilaiser la platforme Admin du wildfly avec cette commande :



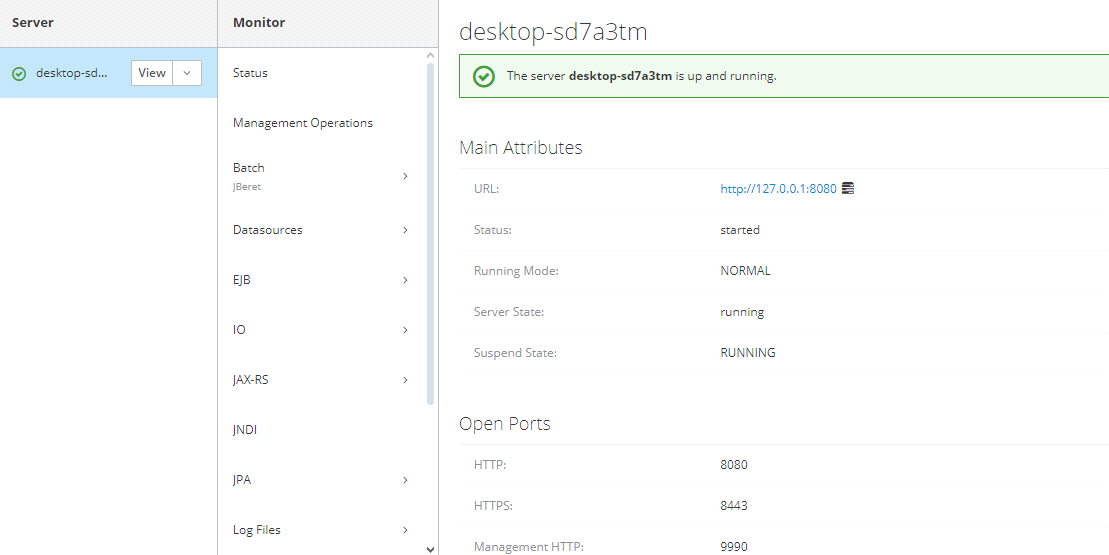
Après on peut accéder la platforme admin en utilisant le lien localhost:9990



Et dans la platforme on saisi l’username et password ici c’est admin pour les deux et on aura l’accès :

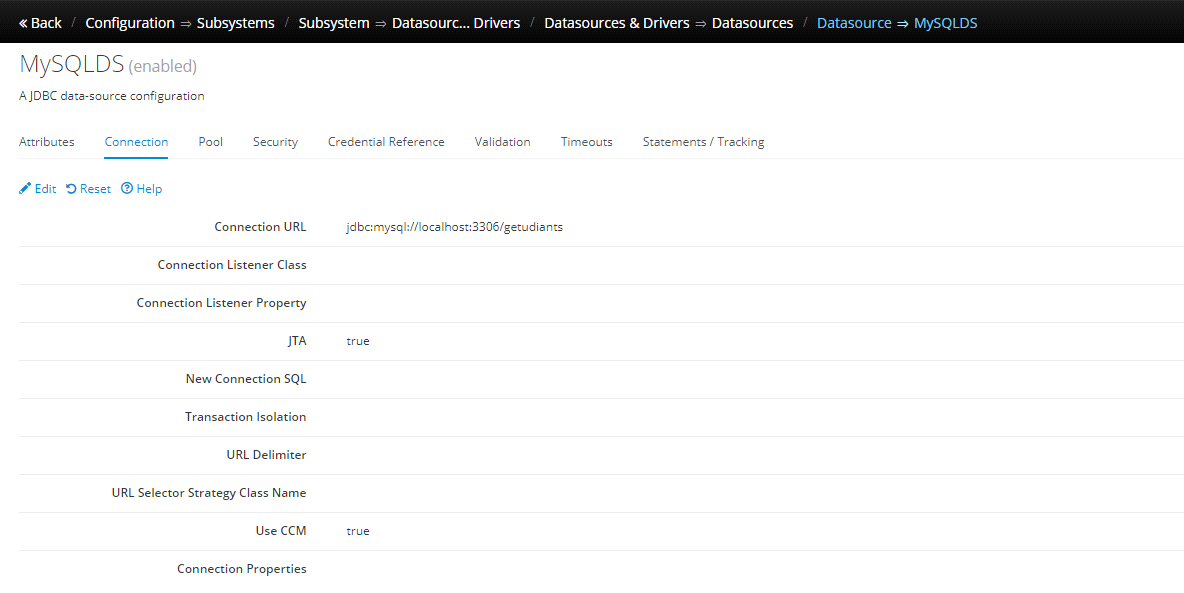


Homepage du wildfly admin control panel:

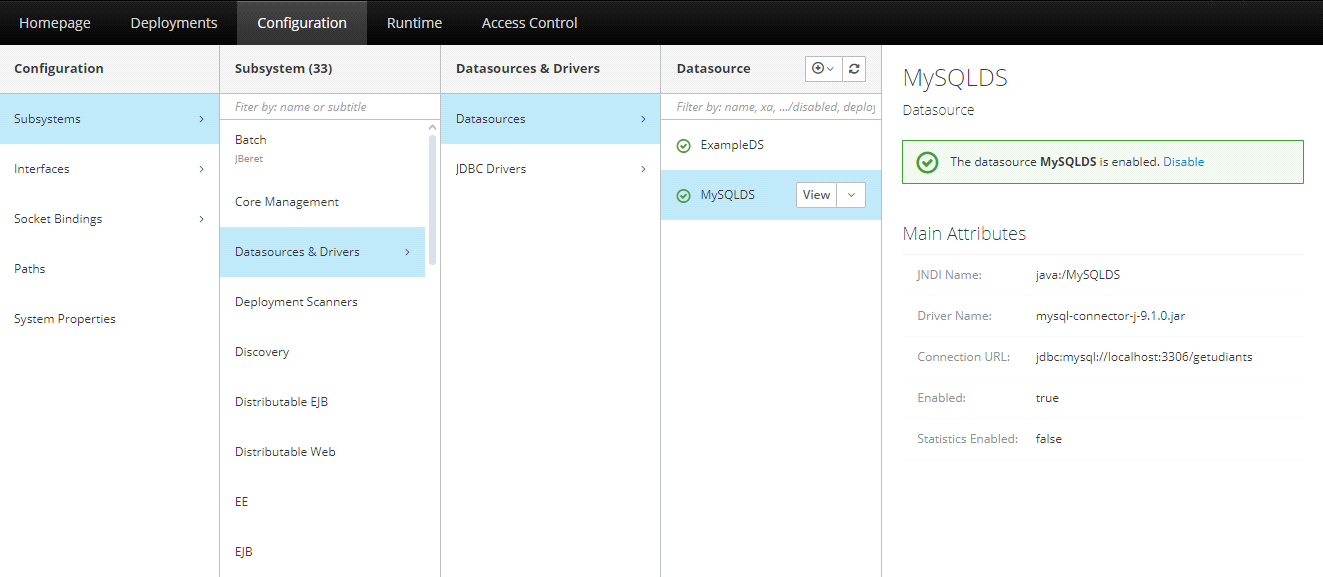


Lorsqu’on a pas des erreurs il doit afficher comme ca

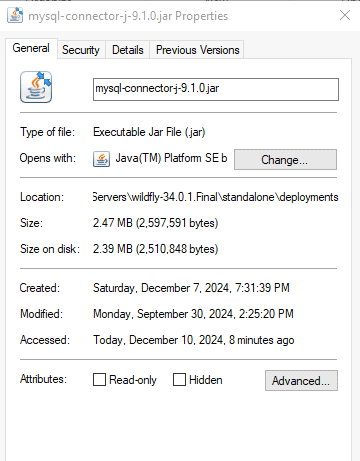
Et on va ajouter le datasource MySQLDS :



Après l’ajout on peut voir que il existe dans les datasources et on va tester la connection du base du données et il va donner en success :



aussi il faut ajouter une deploiement nécessaire qui est mysqlconnector dans ce dossier :



Lorsqu’il est deployé avec success il doit afficher créer ce fichier :



**Etape 3 : creation du classe “Etudiant” :**

Voici notre classe etudiant utilisé :



on a utilisé des annotations JPA et aussi des annotations du lombok pour simplifier l’appel des getters,setters,constructors

**Etape 4 : creation du classe du Type Session Bean**

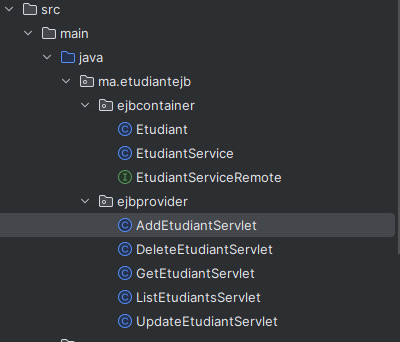
Tout d’abord on va créer l’interface remote qui comme ca :

on appelé tous les méthodes cruds qu’on va utiliser et voici le service qui va l’implémenter :

dans cette partie on a definir tous les méthodes CRUDs qui vont manipuler la base du données connecté ‘’getudiants’’ en utilisant EntityManager aussi on a appelé l’annotation @Stateless qui est nécessaire pour le projet EJB

**Etape 5 : Création du Projet Web pour consommer l’EJB**

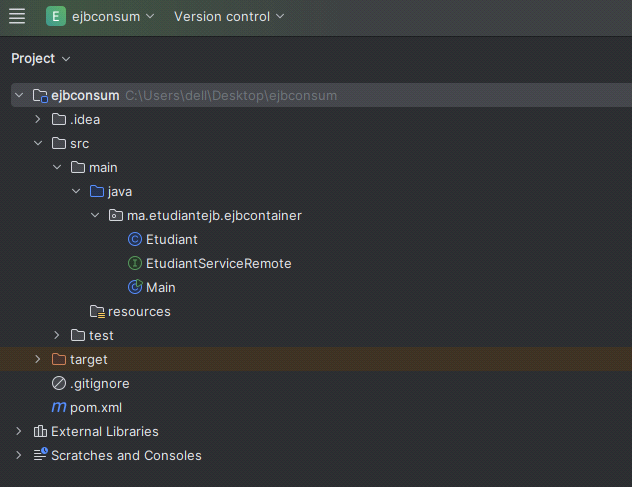
Tout d’abord dans le provider il faut créer les servlets nécessaires pour réaliser les taches CRUDs différents :



Voici une exemple du servlet comme AddEtudiantServlet :

dans cette servlet on a connecter avec service EJB avec une méthode post pour l’ajoute si on utilise Postman ou bien une autre maniére pour tester, et aussi il va créer une utilisateur par défaut qui est Dupont Jean pour remplir la BDD initialement

pour la structure du projet web voici la structure que j’ai fait :

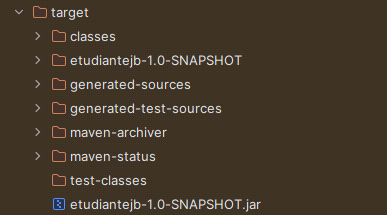


J’appelle juste la classe Remote et Entity d’Etudiant et voici le Main utilisé pour tester les fonctionnalités :

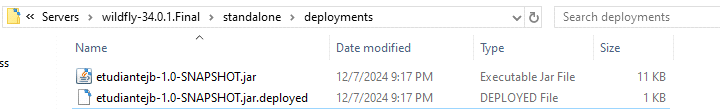


dans cette parite il va appeler l’EJB du projet provider qui est cité dans le context lookup donc il va utilisé le datasource qui est MySQLDS, mais tout d’abord il faut deployer le projet provider dans WildFly admin control panel pour qu’il doit utiliser :

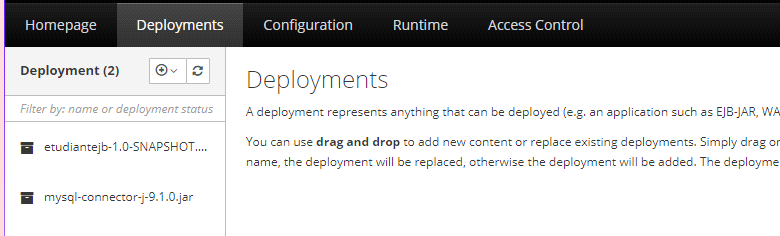
après le build du maven on va voir le fichier target pour avoir le jar :



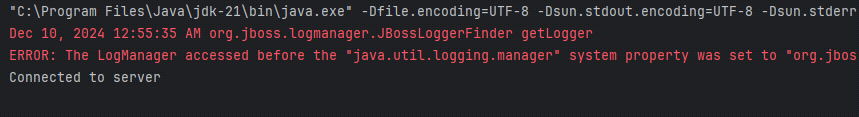
Ce jar on va le copier coller dans le dossier du deployments du wildfly



Et on peut voir que il est déployer avec succes :

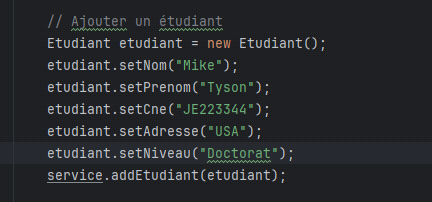


Maintenant pour tester notre web project on va juste lancer le server Wildfly dans project provider, aussi il est important du wildfly d’etre aussi commencé dans le cmd avec ‘’standalone.bat’’

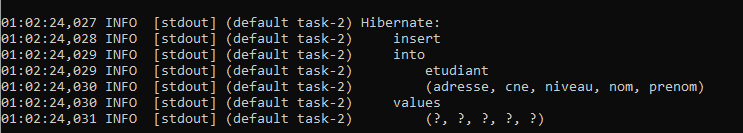


Et on va tester quelque méthodes dans notre web projet pour vérifier la fonctionnement

Pour l’ajout :



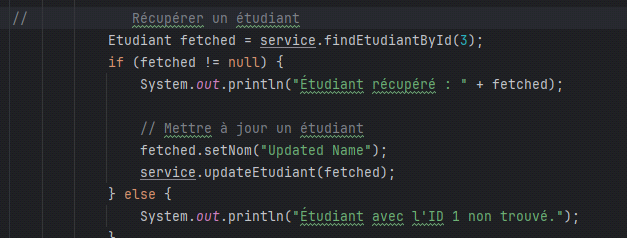
On voir dans le serveur wildfly dans cmd :



Et dans la BDD :



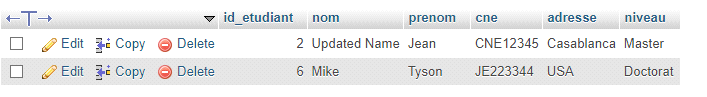
Pour récuperer et modifier :



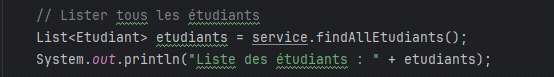
Voici la récuperation :



Et voici la modificaiton fait :



Pour lister tous :



Voici la résultat obtenu :

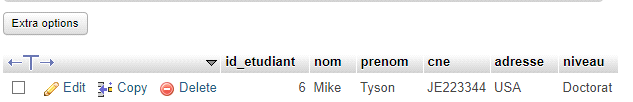


Il affiche encore les infos mais il est très longue pour capturer tous dans le terminal

Pour supprimer :



Base du données après suppression :



Donc tous les méthodes CRUD sont disponibles dans notre web projet et il fonctionne bien on peut implémenter ces méthodes CRUDs plus encore dans une interface web qui est simple