

# TP

## Sujet de TP 1: Création d'un système de connexion en Java EE

### Objectif de TP :

Nous voulons réaliser un système de connexion basique utilisant Java EE, en appliquant les notions que nous avons abordés depuis le 07 Octobre jusqu'au 18 Octobre à savoir : **JavaBean**, **JSP**, **Servlet** etc. Pour être proche du monde réel , on va profiter pour ajouter des ressources statiques telles que **CSS** pour le style, **JavaScript** pour les vérifications simples, et quelques **images** pour l'esthétique. Ce TP n'utilise pas de base de données et vise à vous familiariser avec la communication entre les composants Java EE

### Prérequis :

- Connaissances de base en Java, JSP, et Servlet
- Notions de CSS et JavaScript.
- Un IDE tel que Eclipse
- Serveur Tomcat

Créer un système de connexion en Java EE nommé **lfntiTp**, Dans cette application ,créer 3 fichiers **jsp** respectivement nommés **index.jsp**, **insription .jsp** et **bienvene.jsp**. Dans **index.jsp** mettez un formulaire de connexion avec 2 champs nom et mot de passe , dans **inscription.jsp**, un formulaire pour inscription des utilisateurs qui contient 2 champs nom et mot de passe et dans **bienvenue.jsp** qui affiche bienvenue à l'utilisateur au cas où il s'est connecté puis ce fichier doit être placé dans **web-inf** et les 2 autres dans la racine du projet. Ensuite , créer 2 packages : le premier contient un fichier **bean** qui contient les noms et mot de passe des utilisateurs et le deuxième contient 3 **servlets** .Le premier servlet nommé **Inscription.java** contenant le code de vérification des utilisateurs provenant du fichier **inscription.jsp** ; le second nommé **Connexion.java** qui vérifie la connexion de l'utilisateur s'il est déjà inscrit et le dernier nommé **Deconnexion.java** qui contient le code de deconnection .En plus de tout ce que je viens de donner ,ajouter tes propres notions Css et Js pour que tout le code soit une application réelle en vérifiant les champs ; les redirections et ajouter des notions de session ou quelque notions des objets implicite de jsp ou expression langage, Voici la hiérarchie du projet :

## Structure du Projet :

### 1. Créer un projet Java EE appelé IfntiTp

### 2. Organisation des dossiers :

- **src/** : pour les classes Java.
- **Webapp/** : pour les fichiers **JSP**, **CSS**, **JS**, et les images.
- **WEB-INF/** : pour le fichier web.xml., **bienvenue.jsp** pour cacher ce fichier

## Structure du projet

### IfntiTp/

