

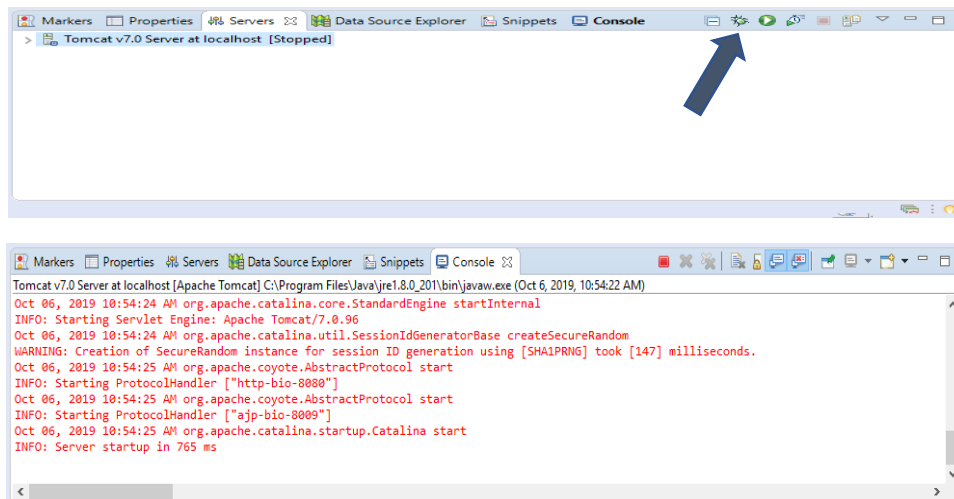
TP 1 : Prise en main de l'environnement JEE et Servlets

1) Création d'un projet JEE

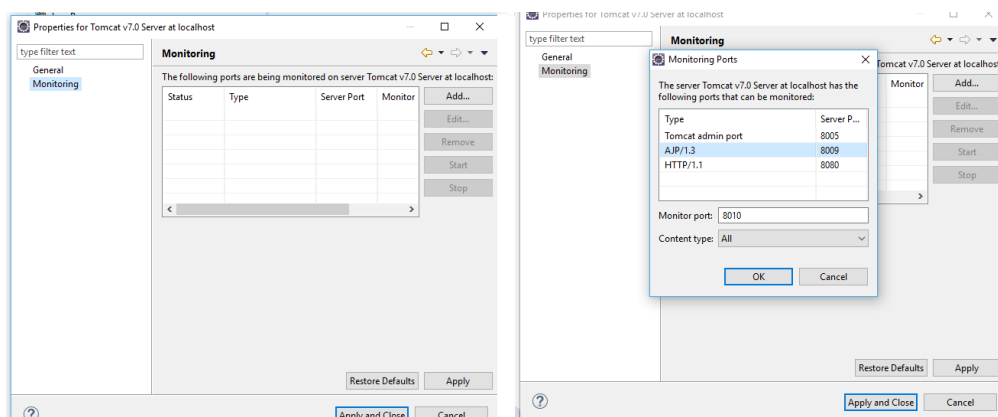
1.1) Apache Tomcat

Sur vos machines, un serveur d'applications Tomcat est installé. Ce serveur possède un conteneur JEE permettant la réalisation de projets JEE.

- Tout d'abord, vérifiez que Tomcat est correctement installé sur vos machines et configuré avec Eclipse



Note : Il se peut que Oracle ou Skype tourne déjà sur le port 8080. Pour pallier au problème, modifier le port utilisé par Tomcat dans le fichier de configuration du serveur Tomcat (<Répertoire Tomcat>/conf/server.xml). Ou via l'interface de l'IDE Eclipse.

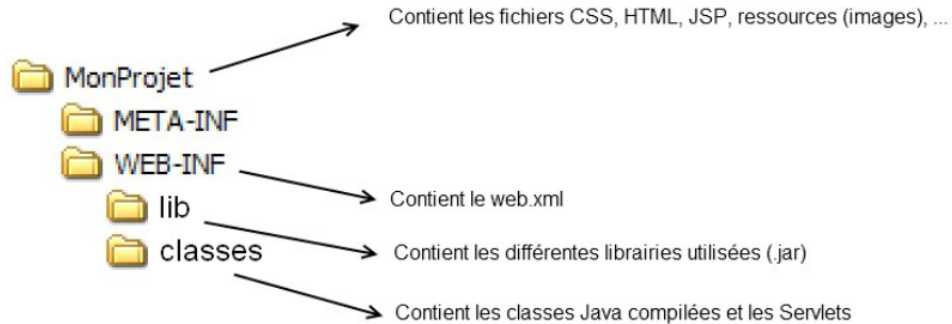


Premier projet JEE

- Créez un nouveau projet JEE : MonProjet contenant un debut.html faisant un Hello World. Il faut créer un nouveau dossier MonProjet contenant le projet JEE. On y mettra les sources Java (servlets et classes) ainsi que les fichiers HTML/JSP/CSS dans leur phase de développement.

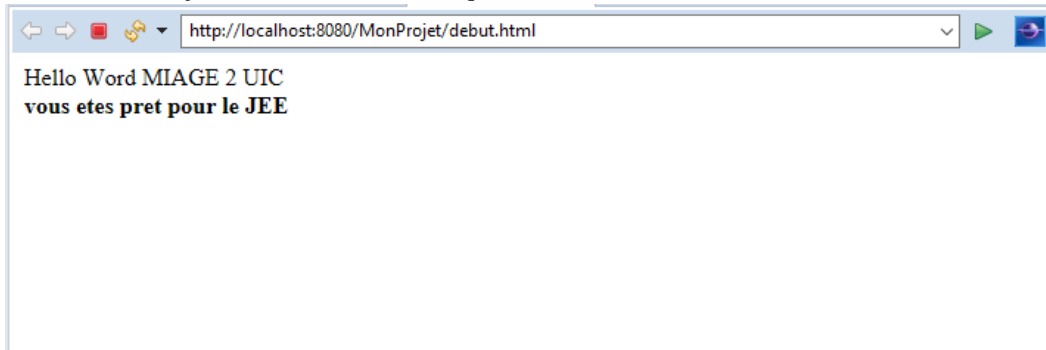
Lorsque le projet est fonctionnel dans le répertoire de développement, on peut passer à la phase de déploiement sur le serveur.

Pour cela, allez dans le répertoire de Tomcat puis dans le dossier webapps (contenant les projets JEE déployés sur le serveur). Dans le conteneur web (Tomcat), la hiérarchie de dossiers suivante doit être respectée pour que Tomcat puisse charger le projet JEE :



Le strict minimum étant le dossier WEB-INF et le fichier web.xml (sauf depuis la v.6 où le web.xml n'est plus obligatoire). Ici on doit rajouter un fichier debut.html dans le répertoire racine du projet.

- Lancez votre Projet sur Tomcat et vérifiez qu'il fonctionne correctement.



- Configurez votre Projet pour que la page de démarrage par défaut soit votre *debut.html*
Il faut configurer le fichier web.xml en ajoutant les lignes suivantes :

```
<web-app>
  <welcome-file-list>
    <welcome-file>debut.html</welcome-file>
  </welcome-file-list>
</web-app>
```

Ainsi, lorsque vous lancez votre site : <http://localhost:8080/MonProjet/> sans spécifier de fichier, Tomcat lance le fichier debut.html par défaut.

Note : Il n'existe pas d'annotation remplaçant ceci, le fichier web.xml est obligatoire si l'on veut définir une url par défaut

1.2) Ma première Servlet

Pour l'instant, nous avons réussi à créer une page Web statique et à la lancer sur notre serveur Tomcat. C'est bien mais ce n'est pas du Web dynamique. Pour faire cela, nous allons créer notre première **Servlet** et allons essayer de la faire fonctionner sur notre site Web. Une Servlet étant une classe Java, il faudra écrire son code dans un fichier .java puis le compiler.

- Créez une Servlet affichant : Bonjour le monde.
Suivez les étapes suivantes :
 1. Commencer d'abord par créer le fichier java : **MaServlet.java**



2. On récupère le flot de sortie : **PrintWriter**, et on écrit dedans le fichier HTML qui sera généré par le serveur.
 3. Ensuite on doit configurer le fichier web.xml qui se charge de faire le mapping entre une URL et notre Servlet.
 4. Ainsi, on peut ensuite accéder à notre servlet par l'URL `http://localhost:8080/MonProjet/Hello`
Attention : **Lorsque le fichier web.xml est modifié, il faut recharger le Projet par Tomcat**
- Améliorer le programme en écrivant une nouvelle version qui permet de s'affranchir du fichier web.xml dans des cas très simples comme celui-ci (il reste obligatoire si l'on veut définir davantage de propriétés avancées).
Suivez les étapes suivantes :
 1. Définir le mapping de la Servlet par une annotation (**@WebServlet**) directement dans le code de la servlet.
 2. Cette annotation contient l'**url-pattern** correspondant à l'url qui permettra d'appeler la servlet sur le serveur.

2) Servlet avec formulaire

Implémenter une application qui permet à un étudiant de l'UIIC d'entrer son login et mot de passe dans un formulaire HTML, puis d'afficher un message d'accueil après la soumission du nom. Lorsque l'étudiant fournit son login et mot de passe. La servlet effectue des contrôles élémentaires sur leur intégrité puis affiche un message de ce style « Bienvenue à l'UIIC votre login est : et votre mot de passe est »



3) Edition d'une page HTML via Servlet

1. Réalisez une nouvelle servlet ayant pour apparence

The screenshot shows a web form with the following fields and buttons:

- Nom: Threepwood
- Prénom: Guybrush
- Code postal: 75027
- Miel: threepwood@monkey.c
- Buttons: Envoyer, Remettre les données originales

2. Lorsque l'utilisateur soumet ces données, votre servlet effectuera des contrôles élémentaires sur leur intégrité:

| | |
|-------------|--|
| Nom, prénom | que des caractères alphabétiques |
| code postal | 5 chiffres |
| Adresse | 2 chaînes arbitraires délimitées par "@" |

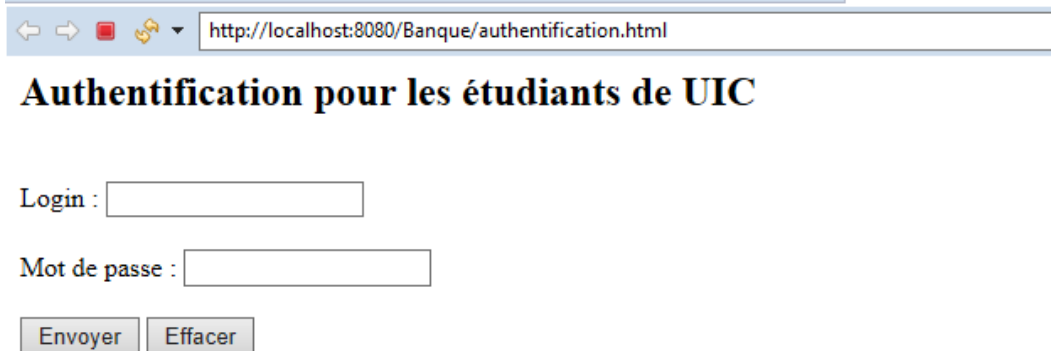
3. Une page montrera alors le résultat de ces vérifications à l'utilisateur.
4. Editer une table html qui, utilisés les balises <table> et <tr> (ligne de tableau), et affiche un tableau où la couleur de fond des lignes est alternée comme suit :

Un joli tableau

| | |
|---------|--|
| ligne 1 | C'est une ligne dans un tableau |
| ligne 2 | C'est aussi une ligne |
| ligne 3 | C'en est encore une. décidément |
| ligne 4 | Mais il y a que des lignes dans ce tableau |
| ligne 5 | En même temps, c'est un tableau |
| ligne 6 | Faut s'attendre à ce genre de choses |

4) Traitement servlet et redirection

1. Ecrire une page html qui retourne au client web la page suivante :



On construira un formulaire dont le code pourra être :

```
1=<html>
2=<head>
3 <title>Authentification pour SMB111</title>
4 <meta charset="UTF-8">
5 </head>
6=<body>
7   <h2>Authentification pour les étudiants de UIC</h2>
8=   <form action="AuthentificationServlet" method="post">
9     <br />Login : <input type="text" name="Login"><br /> <br />Mot
10    de passe : <input type="password" name="motDePasse"><br /> <br />
11    <input type="submit" value="Envoyer" /> <input type="reset"
12    value="Effacer" />
13  </form>
14 </body>
15 </html>
```

Dans un tel formulaire lorsque l'utilisateur clique sur le bouton Envoyer, une connexion est faite sur la servlet dont l'URL se termine par **AuthentificationServlet**. Cette servlet vérifie si le couple



(nom de login, mot de passe) est correct. On pourra prendre le couple (uic111, uic111pw). Si le couple est correct, alors l'utilisateur sera redirigé vers la page **operationbancaire.html**.

authentification réussie

Votre compte bancaire contient : 300.0 euro

Vous voulez faire
un virement
ou un retrait

de euro

Les informations concernant l'opération bancaire sont envoyées à la servlet **OperationBancaireServlet**. Si l'utilisateur choisit un virement alors le solde sera incrémenté dans le cas contraire le solde est décrémenté.

Si ce couple n'est pas correct on doit avoir une page comme :

échec à l'authentification

Il est : Wed Dec 09 15:59:31 CET 2009