

Utilisation de Tomcat & MySQL

Sous Win2000NT et WinXP

Je suppose que le serveur web Tomcat et la base de données MySQL sont installés correctement.

Références

<http://127.0.0.1:8080/tomcat-docs/index.html>

Plus les liens de Stéphane que je rappelle ici :

<http://www.eomss.net/LRDB/1r110/main.php?page=tutorial>

<http://www.apl.jhu.edu/~hall/java/Servlet-Tutorial/Servlet-Tutorial-Intro.html>

<http://scitec.uwichill.edu.bb/cmp/online/CS36L/tutorials/TomcatIntroduction/TomcatIntroIndex.html>

<http://www.onjava.com/pub/a/onjava/2001/03/15/tomcat.html>

Tomcat

Introduction

Dans cette partie je vous présente quelques notions et concepts liés au serveur web Tomcat (StandAlone). Je fais une synthèse de documents, donc tous les concepts ne sont pas présentés.

Dans tous le document je considère que la variable d'environnement `$CATALINA_HOME` désigne le répertoire d'installation de Tomcat et la variable `$JAVA_HOME` désigne le répertoire où se trouve le JDK. Pour vérifier si ces variables d'environnement existent, faire :

Clique bouton droit sur Poste de Travail, choisir propriétés, onglet Avancé, variables d'environnement

Si `$CATALINA_HOME` n'existe pas, la créer en cliquant sur Nouveau et remplir les champs Nom et Valeur. Mettre le chemin complet vers le répertoire de Tomcat

Exemple : Nom : CATALINA_HOME
 Valeur : E:\Program Files\Apache Tomcat 4.0 (il se peut que votre répertoire soit différent)

Si `$JAVA_HOME` n'existe pas profitez en pour la définir.

L'organisation du répertoire `$CATALINA_HOME` est la suivante :

- `\bin` : contient les scripts pour démarrer, relancer et arrêter le serveur. Normalement, on devrait pas y mettre les doigts.
- `\conf` : contient 3 fichiers de configurations xml. Ultérieurement on modifiera le fichier `server.xml` qui permettra de définir le répertoire racine de notre application web (webapp)
- `\logs` : contient les fichiers logs du serveur.
- `\webapps` : contient les différentes applications web disponible sur le serveur. C'est dans ce répertoire que l'on travaillera. Les applications web sont chargés automatiquement au démarrage de Tomcat. Il faut seulement respecter les règles de déploiement de la webapp.
- `\work` : répertoire temporaire pour les webapps. Ce répertoire est intéressant lorsqu'on veut déboguer une page jsp. Il suffit de trouver le servlet correspondant.

Pour faciliter le déploiement des applications web, celles ci doivent respecter un certain standard. L'application web doit être installée dans le répertoire `$CATALINA_HOME\webapps`. Je nomme `$CATALINA_HOME\webapps\projet` le répertoire de l'application web « monprojet ». Pour vérifier le standard,

L'application web doit respecter l'organisation suivante dans son répertoire

`$CATALINA_HOME\webapps\projet` :

- *.html, *.jsp, etc... : ce sont les fichiers HTML, les pages JSP et autres qui prennent place à la racine de `$CATALINA_HOME\webapps\projet`.
- `\WEB-INF\web.xml` : ce fichier permet de décrire un servlet unique comme point d'entrée de la webapp. Il n'est pas obligatoire.
- `\WEB-INF\classes` : ce répertoire contient toutes les classes java utilisées par la webapp.
- `\WEB-INF\lib` : ce répertoire est utilisée pour placer les fichiers .jar qui contiennent des classes java.

Il existe 4 techniques de déploiement possibles, je vous en présente seulement une :

- La première consiste à copier le contenu du répertoire de l'application « monprojet » dans le répertoire `$CATALINA_HOME\webapps\projet`, puis de redémarrer le serveur Tomcat. L'application est alors disponible à l'URL suivante : <http://localhost:8080/projet> si vous n'avez pas redéfinis le répertoire racine et le port d'écoute du serveur Tomcat

Pour redémarrer le service, faire :

Démarrer – Paramètres – Panneau de configuration – Outils d'administration – Services
Sélectionnez Apache Tomcat et appuyez sur le bouton « Redémarrer le service ».

ATTENTION : selon les versions utilisées pour le serveur web Apache Tomcat, il peut être nécessaire de modifier le fichier `$CATALINA_HOME\conf\web.xml`. Vérifier que les lignes suivantes ne sont pas en commentaires :

```
<!-- The mapping for the invoker servlet -->
<servlet-mapping>
  <servlet-name>invoker</servlet-name>
  <url-pattern>/servlet/*</url-pattern>
</servlet-mapping>
```

Si ces lignes sont commentées, il faut les décommenter.

Pour les autres méthodes consultez la page suivante : <http://127.0.0.1:8080/tomcat-docs/appdev/deployment.html>

Je m'arrête ici pour l'introduction, car à ce stade on peut commencer à travailler. Je décrirai dans un document livrable l'organisation de notre application web.

Utilisation

Création d'un servlet.

Lire le fichier `RequestInfo.java`,
Compilez le fichier `RequestInfo.java`

ATTENTION : pour compiler, mettez à jour la variable `$CLASSPATH` pour qu'elle puisse aller chercher le fichier `servlet.jar`. Ce fichier se trouve dans le répertoire `$CATALINA_HOME\common\lib`.

Créer le répertoire suivant `$CATALINA_HOME\webapps\test\WEB-INF\classes`
Copier `RequestInfo.class` dans `$CATALINA_HOME\webapps\test\WEB-INF\classes`
Redémarrer le service Apache Tomcat

Allez à l'URL suivante: <http://127.0.0.1:8080/test/servlet/RequestInfo>

=> Le résultat donne des informations sur la requête que vous venez de faire.

Création d'une page JSP

Lire le fichier `date.jsp`
Copier `date.jsp` dans `$CATALINA_HOME\webapps\test`

Allez à l'URL suivante: <http://127.0.0.1:8080/test/date.jsp>

=> La date d'aujourd'hui s'affiche.

Tomcat + MySQL

On va créer une page jsp qui fait appel à la base de données et affiche les résultats en html

- Créer au moins une table

Lancer le client mysql.exe dans le répertoire \$MYSQL_HOME\bin

Se connecter à la base :connect test

Lancer le script d'initialisation : source sql.txt

- Créer la page jsp

Copier le fichier DBtest.jsp dans le répertoire \$CATALINA_HOME\webapps\test

Allez à l'URL suivante: <http://127.0.0.1:8080/test/DBtest.jsp>

=> Le contenu de la base s'affiche sous forme d'un document html.

Et voilà le tour est joué.

Debug

Juste un mot pour les procédures de debug.

Comme chacun sait une page JSP est compilé à la volée comme un servlet.

Pour les pages JSP, il est possible d'accéder au servlet généré par Tomcat dans le répertoire \$CATALINA_HOME\work\localhost.

Exemple : Pour la webapp « test » Les fichiers se nomment :

\$CATALINA_HOME\work\localhost\test\date\$jsp.java

et

\$CATALINA_HOME\work\localhost\test\DBtest\$jsp.java