

Architecture Web

Travaux Pratiques

Accès au service :

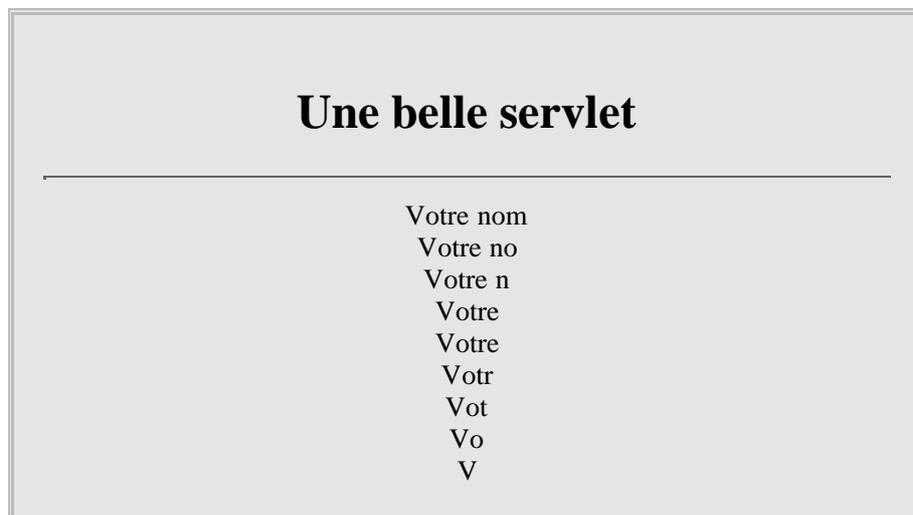
Connectez-vous sous windows en tant que guest/password sur le domaine INFO-nn. Un serveur Apache-Tomcat tourne sur chaque poste sur le port 8080, on peut donc s'y connecter avec l'URL: <http://localhost:8080/>

Si vous êtes à la recherche de documentation: Consultez la rubrique [Références](#)

Le corrigé des exercices 1 à 6 est disponible !

Exercice 1 – Réalisation et déploiement d'une servlet simple

1. Réalisez une servlet qui génère la page HTML suivante:



La ligne horizontale est réalisée avec la balise `<hr>`, on peut centrer du texte à l'aide de `<div align=center>... </div>`.

Vous pourrez avoir besoin d'indiquer le répertoire d'accès à J2EE pour compiler le fichier .class de votre servlet. Le fichier à lier est `C:\Program files\Java\Sun\lib\j2ee.jar`.

2. Une fois la servlet programmée, il faut la *déployer* sur le serveur. Pour cela, elle doit faire part d'une application web normalisée qui se trouvera dans le répertoire *webapps* de tomcat.

Format normalisé d'une application web JavaEE

`*.html,`
`*.jsp ...` fichiers visibles depuis le client: html, images, javascript, css...

WEB-INF/classes/	répertoire contenant toutes les classes Java de l'application – en particulier les servlets – et les ressources associées, à l'exception de celles qui ont été combinées sous forme de fichiers JAR.
WEB-INF/lib/	contient les JAR nécessaires à l'application (bibliothèques...).
WEB-INF/web.xml	décrit le déploiement de l'application: les différents servlets qui la composent, comment les initialiser et y accéder, certaines contraintes de sécurité, etc.

Créez votre application suivant ce format (dans le répertoire de votre choix). Un exemple simpliste de fichier `web.xml` vous est fourni.

Une fois l'application web mise sous cette forme normalisée, la méthode la plus flexible pour l'installer sur le serveur consiste à l'encapsuler dans un fichier `.war` (*web archive*). Ce fichier est une archive JAR et peut être créé via la commande: `c:\Program Files\Java\jdk1.5.0_07\jar.exe -cvf ../toto.war WEB-INF` depuis la racine de votre application.

Ensuite, vous pouvez l'uploader sur le serveur depuis l'"interface manager" de tomcat (accessible sur <http://localhost:8080/manager/html>)

Exercice 2 – Servlet avec formulaire

1. Réalisez une nouvelle servlet ayant pour apparence:

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

- Nom: Threepwood
- Prénom: Guybrush
- Code postal: 75027
- Mèl: threepwood@monkey.c

Below the fields are two buttons: "Envoyer" and "Remettre les données originales".

Pour le déploiement de votre servlet, vous pouvez utiliser la même application web que pour l'exercice 1.

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 "@"

Une page montrera alors le résultat de ces vérifications à l'utilisateur.

Exercice 3 – Page JSP simple

Refaites l'exercice 1, mais cette fois à l'aide d'une page JSP plutôt que d'une servlet. Examinez la servlet générée à partir de votre JSP et comparez avec votre propre code.

Exercice 4 – Servlet en entrée, JSP en sortie

Reprendre l'exercice 2 en réalisant les sorties à l'aide de pages JSP. Les données doivent toujours être contrôlées par une servlet après appui du bouton "Envoi".

Exercice 5 – Gestion de session

Réaliser une application WEB comportant un ensemble d'URL correspondant aux commandes:

- initialiser une session utilisateur sur le serveur
- afficher le formulaire de l'exercice 2 (la validation du formulaire par l'utilisateur ajoute les données à la session)
- visualiser l'ensemble des données de la session courante
- détruire la session

Les pages peuvent être réalisées indifféremment à l'aide de servlets ou de JSP.

Exercice 6 – Taglib

- Programmer des tags qui, utilisés à la place de `<table>` et `<tr>` (ligne de tableau), affichent 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

- Déployez votre taglib.

- Écrivez une JSP en montrant le fonctionnement.
-

Références

J2EE 1.4 Tutorial	http://java.sun.com/j2ee/1.4/docs/tutorial/doc/
Java EE 5 Tutorial	http://java.sun.com/javaee/5/docs/tutorial/doc/index.html
Guide Tomcat	http://localhost:8080/tomcat-docs/appdev/
JavaEE	http://java.sun.com/javaee/
JavaEE API	http://java.sun.com/javaee/5/docs/api/
Servlets	http://java.sun.com/products/servlet/
JSP	http://java.sun.com/products/jsp/
JavaBeans	http://java.sun.com/products/javabeans/
EJB	http://java.sun.com/products/ejb/