

TP 1 – Prise en main de l'environnement JEE dans Apache Tomcat

Exercice 1 : Manipulation du serveur Apache Tomcat

Sur vos machines, installez un **serveur d'applications Tomcat**. Ce serveur possède un **conteneur JEE** permettant la réalisation du projet Java EE.

- Tout d'abord, vérifiez que Tomcat est correctement installé sur vos machines
- Lancez Tomcat startup.bat dans le répertoire **TOMCAT_HOME/bin**
- Etudiez la configuration de Tomcat dans le fichier **server.xml** du répertoire **TOMCAT_HOME/conf** et définir :
 - La balise principale
 - Les balises filles et le nombre de leurs apparitions
 - Les attributs obligatoires et optionnels.
 - La balise **connector** et ces **attributs**
- Etudier le fichier « **tomcat-users.xml** » qui se trouve dans le répertoire **TOMCAT_HOME/conf** et définir :
 - La balise principale ainsi que ces attributs
 - Les balises **role**, **user** et leurs **attributs**
 - Activer un compte utilisateur
- Entrer au site local du Tomcat (voir figure ci-dessous)

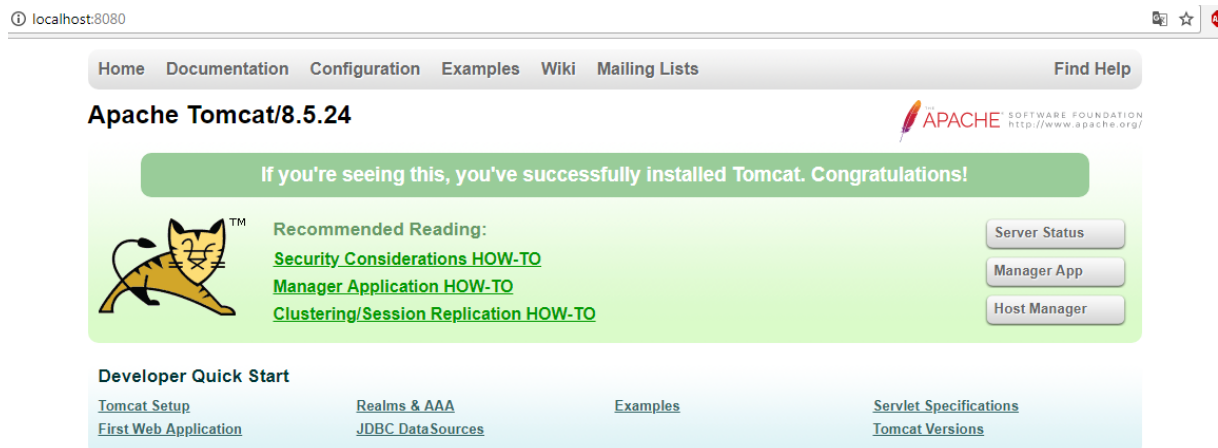


Figure 1. Site d'administration d'Apache Tomcat

- Ajouter les balises ci-dessous dans le fichier tomcat-users.xml

```
<role rolename="manager-gui"/>
<role rolename="admin-gui"/>
<user username="manager" password="manager" roles="manager-gui"/>
<user username="admin" password="admin" roles="admin-gui"/>
```

- Etudier le « Server Status », « Manager App » et « Host Manager »
- Etudier les différents projets d'exemples déjà installés sur le serveur.

Exercice 2 :

Soit la Servlet suivante :

```
public class MaServlet extends HttpServlet
```

```
{
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException
    {
        response.setContentType("text/html"); // indique que le contenu de la page à
            envoyer au client est du HTML
        PrintWriter out = response.getWriter(); // indique le flux récupéré de sortie
            (qui sera envoyé au client)
        out.println( // J'écris le fichier HTML à envoyer au client dans mon flux

            ... (à compléter) ...

        )
    }
}
```

- 1- Compléter la Servlet ci-dessus qui renvoie un tableau de 10 cases par 10 selon le modèle suivant :

M(1,1)	M(1,2)	M(1,3)	M(1,4)	M(1,5)	M(1,6)	M(1,7)	M(1,8)	M(1,9)	M(1,10)
M(2,1)	M(2,2)	M(2,3)	M(2,4)	M(2,5)	M(2,6)	M(2,7)	M(2,8)	M(2,9)	M(2,10)
M(3,1)	M(3,2)	M(3,3)	M(3,4)	M(3,5)	M(3,6)	M(3,7)	M(3,8)	M(3,9)	M(3,10)
M(4,1)	M(4,2)	M(4,3)	M(4,4)	M(4,5)	M(4,6)	M(4,7)	M(4,8)	M(4,9)	M(4,10)
M(5,1)	M(5,2)	M(5,3)	M(5,4)	M(5,5)	M(5,6)	M(5,7)	M(5,8)	M(5,9)	M(5,10)
M(6,1)	M(6,2)	M(6,3)	M(6,4)	M(6,5)	M(6,6)	M(6,7)	M(6,8)	M(6,9)	M(6,10)
M(7,1)	M(7,2)	M(7,3)	M(7,4)	M(7,5)	M(7,6)	M(7,7)	M(7,8)	M(7,9)	M(7,10)
M(8,1)	M(8,2)	M(8,3)	M(8,4)	M(8,5)	M(8,6)	M(8,7)	M(8,8)	M(8,9)	M(8,10)
M(9,1)	M(9,2)	M(9,3)	M(9,4)	M(9,5)	M(9,6)	M(9,7)	M(9,8)	M(9,9)	M(9,10)
M(10,1)	M(10,2)	M(10,3)	M(10,4)	M(10,5)	M(10,6)	M(10,7)	M(10,8)	M(10,9)	M(10,10)

- 2- Changer le type du sortie mime pour envoyer la réponse vers un fichier excel.

Exercice 3 :

- 1- Ecrire une Servlet qui nous permet de retourner les différentes informations sur l'URL, citées ci-dessous, En utilisant les différentes méthodes des interfaces HttpServletRequest et HttpServletResponse :

- Le nom du serveur (`serverName`)
- Le port du serveur d'écoute (`serverPort`)
- Le nom de l'application hébergeant la servlet (`contextPath`)
- Le chemin de la servlet (`servletPath`)
- Le type de la requête http utilisée (`method`)
- Les paramètres passés dans l'URL (`queryString`)
- L'URL utilisée pour contacter la servlet (`requestURL`)
- L'adresse locale (`localAddr`)
- Le nom local (`localName`)
- Le port local (`localPort`)
- L'adresse IP locale (`remoteAddress`)
- Le nom de l'hôte local (`remoteHost`)

- 2- Créer un formulaire en html/jsp (deux zones de texte et deux boutons submit et reset)

- Récupérer les données écrites dans les zones de texte après qu'on clique sur « submit »
- Récupérer la longueur du texte (contentLength), en utilisant la méthode « get » ou post, conclure
- Ecrire le code source suivant et conclure :

```
Enumeration<String> headerNames = request.getHeaderNames();
out.println("<H2><center>Les entetes provenant du client</center></H2>");
out.println("<table border=2 bordercolor=red>");
out.println("<tr><th bgcolor=pink>Header Name</th><th bgcolor=pink>Header Value</th></tr>");
while (headerNames.hasMoreElements()) {
    String name = headerNames.nextElement();
    String value = request.getHeader(name);
    out.println("<tr><th>" + name + "</th><th>" + value + "</th></tr>");
}
out.println("</table>");
```

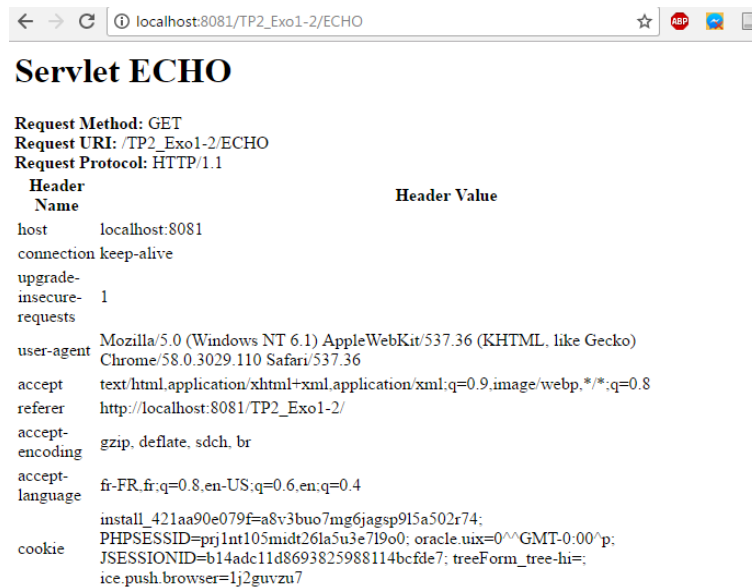
3- Soit le code source suivant :

```
protected void service(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException
{
    PrintWriter out = response.getWriter();

    Enumeration<String> NomsParam = request.getParameterNames();
    while(NomsParam.hasMoreElements()) {
        String NomParam = (String) NomsParam.nextElement();
        out.println("NomParameter : " + NomParam + "<br>");

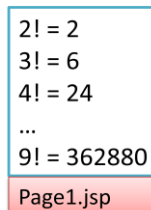
        String[] ValeursParam = request.getParameterValues(NomParam);
        if (ValeursParam.length == 1) {
            String ValeurParam = ValeursParam[0];
            if(ValeurParam.length() == 0) {
                out.println("Aucune valeur selectionnée <br>");
            }
            else
                out.println(ValeurParam + "<br>");
        }
        else {
            out.println("<ul>");
            for(int i=0; i< ValeursParam.length; i++)
            {
                out.println("<li>" +
ValeursParam[i]+ "</li><br>");
            }
            out.println("</ul>");
        }
    }
}
```

Ecrire une servlet qui permet d'afficher les informations sur le client comme la figure affichée ci-dessous :



Exercice 4 :

- 1- Créer une page web jsp permettant d'afficher le factoriel des nombres entre 0 et 9
- 2- Générer dans cette page une méthode qui me permet d'afficher le factoriel des nombres entre 0 et n (choisi par l'utilisateur)



- 3- Modifier votre code pour calculer le factoriel d'un nombre lu au clavier



Exercice 5 :

On veut réaliser une page d'ouverture de session avec :

- Un titre de page « page d'accueil »
- Une indication d'introduction de : Nom d'utilisateur
- Une indication d'introduction de : Mot de passe
- Un bouton d'ouverture de session

La page jsp intitulée « entree.jsp » permet de valider l'entrée en consultant une base de données en mysql nommée « mabase.sql », un login « login » et un mot de passe « password ».

Cette base contient une table des utilisateurs nommée « users » et qui a la forme suivant (id, login, password, firstname, lastname)

Si le nom d'utilisateur et le mot de passe est valide, alors une page est générée avec le titre « Bienvenu » et affiche le nom et prénom du propriétaire

Sinon un message d'erreur est affiché et une invitation de relance de la page d'accueil.