



+ Accueil Clients 200 Tutos Catalogue #10000Codeurs

Nous Contacter



Votre première servlet

Accueil / 200 tutos / Nos tutoriaux / j2ee / Votre première servlet

Créez pas à pas votre première servlet. Utilisez l'assistant de création et configuration d'Eclipse. Comprenez la configuration d'une servlet avec les annotations ou en passant par le fichier web.xml.

Dans le prochain tutoriel, [Créez votre première JSP](#)

Objis : spécialiste JAVA depuis 2005

Touchez du doigt notre approche pédagogique (70% de pratique) à travers ce tutoriel.

Comme [nos clients](#), contactez-nous (contact@objis.com) pour vos besoins en [assistance technique](#) ou en [formations](#) sur mesure dans nos locaux, dans vos locaux ou à distance.





+ [Tutoriel projet de création web dynamique avec Eclipse](#) [Accueil](#) [Clients](#) [200 Tutos](#) [Catalogue](#) [#10000Codeurs](#)

Liens utiles

Nous Contacter 
+ de 100 tutoriaux java/jee Objjs

[Vos premiers pas de programmeur JEE](#)

[Objjs, spécialiste formation java depuis 2005](#)

[Catalogue formations Objjs](#)

Objectifs

Codage et configuration d'une Servlet HTTP

Déploiement de la servlet dans un moteur de Servlet Tomcat

Programme

[Partie 1 : Création du squelette de la servlet](#)

[Partie 2 : Configuration de la servlet](#)

[Partie 3 : Codage de la servlet](#)

[Partie 4 : Déploiement de la servlet](#)

[Partie 5 : Refactoring](#)





Nous allons vous faire aimer Java

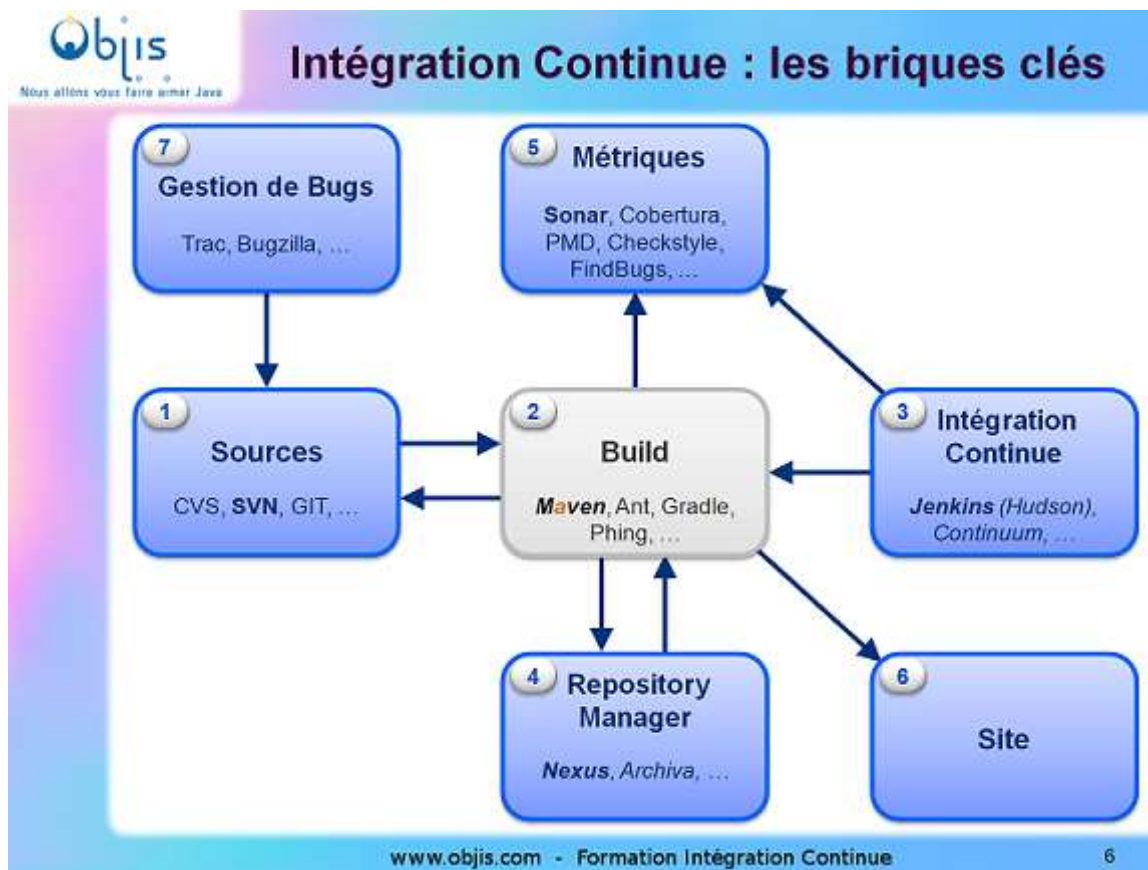
Conclusion : Conclusion

[+ Accueil](#) [Clients](#) [200 Tutos](#) [Catalogue](#) [#10000Codeurs](#)**Durée**

30 min

[Nous Contacter](#)

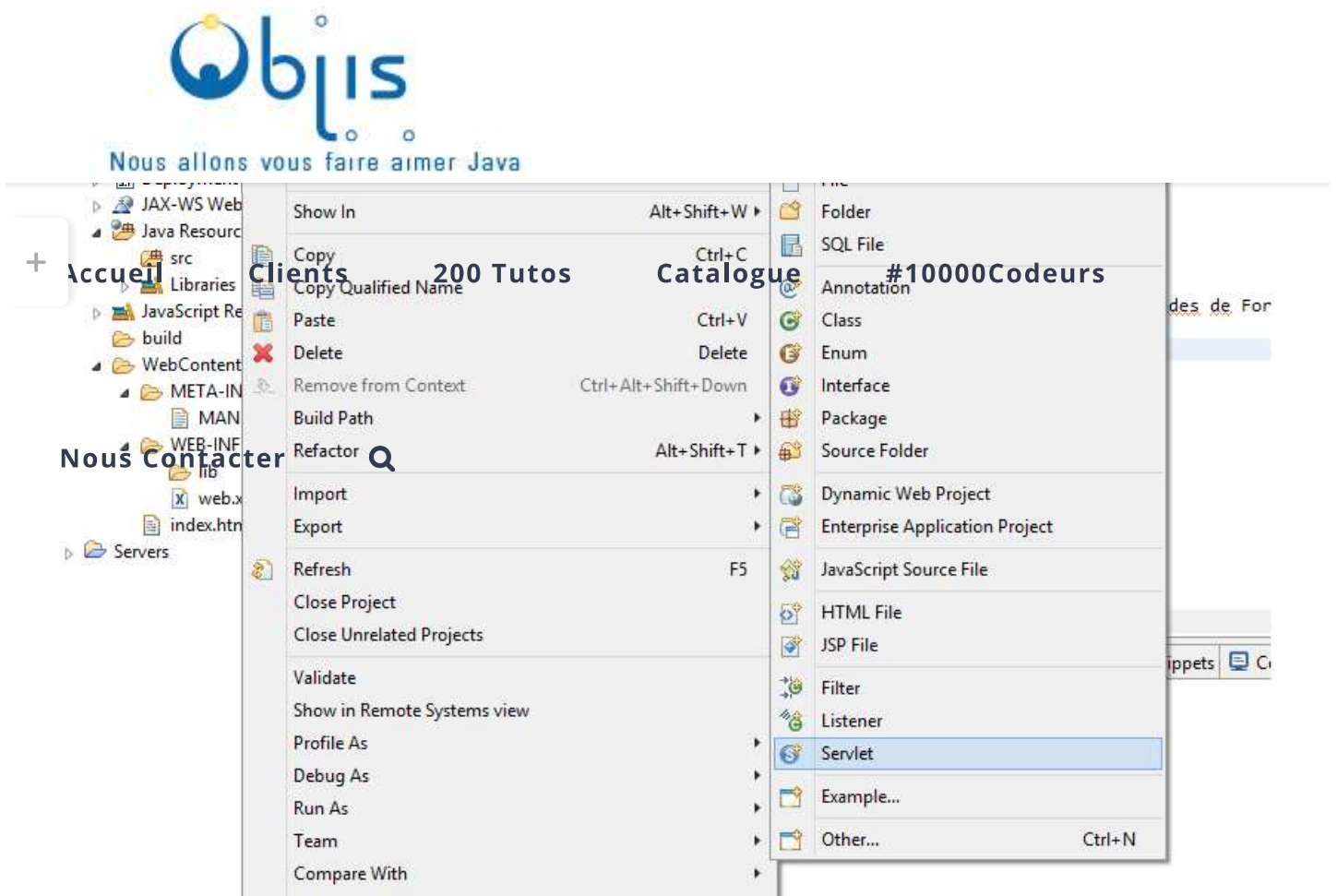
PUBLICITE : votre formation 'Intégration continue' avec Objis



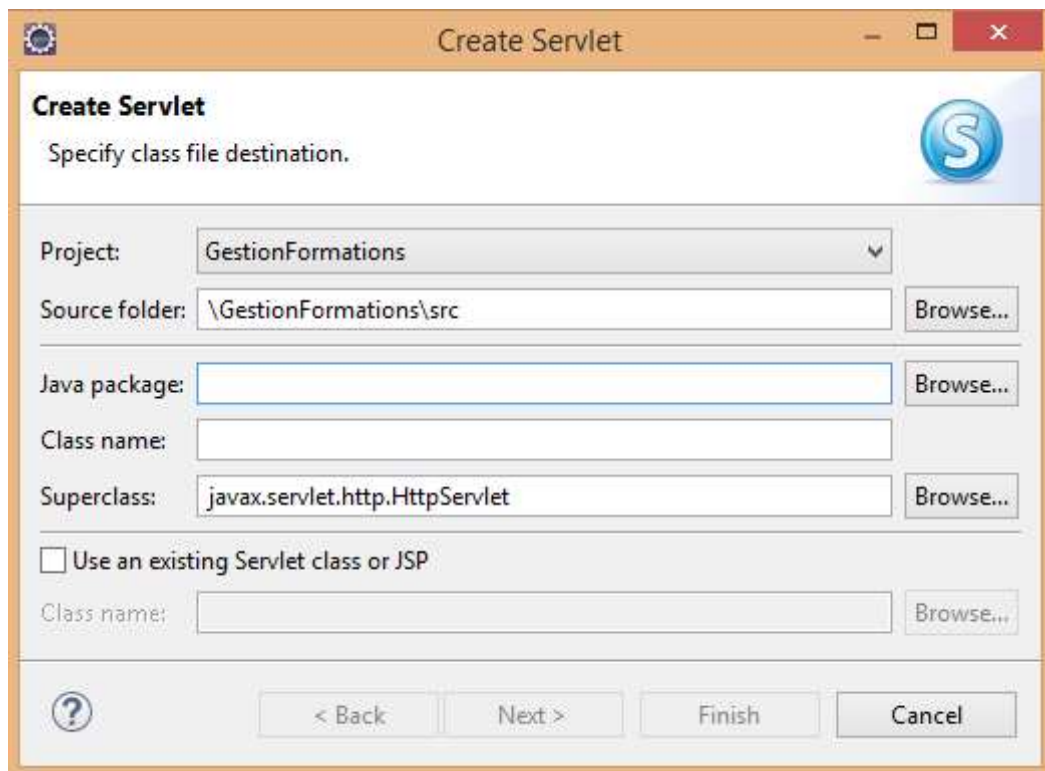
Partie 1 : Création du squelette de la servlet

A partir d'Eclipse, faire menu File → New → Servlet





L'écran suivant apparaît



Ajouter le package *com.objis.gestionformations* ainsi que la classe *MaServlet*



Nous allons vous faire aimer Java

+ Accueil

Clients

200 Tutos

Catalogue

#1000000 de users

Nous Contacter

Project: GestionFormations

Source folder: \GestionFormations\src

Java package: com.objjs.gestioninformations

Class name: MaServlet

Superclass: javax.servlet.http.HttpServlet

☐ Use an existing Servlet class or JSP

Class name: MaServlet

?

< Back

Next >

Finish

Cancel

Question

En utilisant l'API JEE 7 , Expliquez ce qu'est **javax.servlet.http.HttpServlet** et citez 3 méthodes .

Expliquez les méthodes *init*, *destroy* et *service*

Ensuite cliquez sur le bouton Next





Nous allons vous faire aimer Java

+

Accueil

Clients

200 Tutos

Catalogue

#10000Codeurs

Nous Contacter

Q

Name: MaServlet

Description:

Initialization parameters:

Name	Value	Description

Add... Edit... Remove

URL mappings:

/MaServlet

Add... Edit... Remove

☐ Asynchronous Support

?

< Back

Next >

Finish

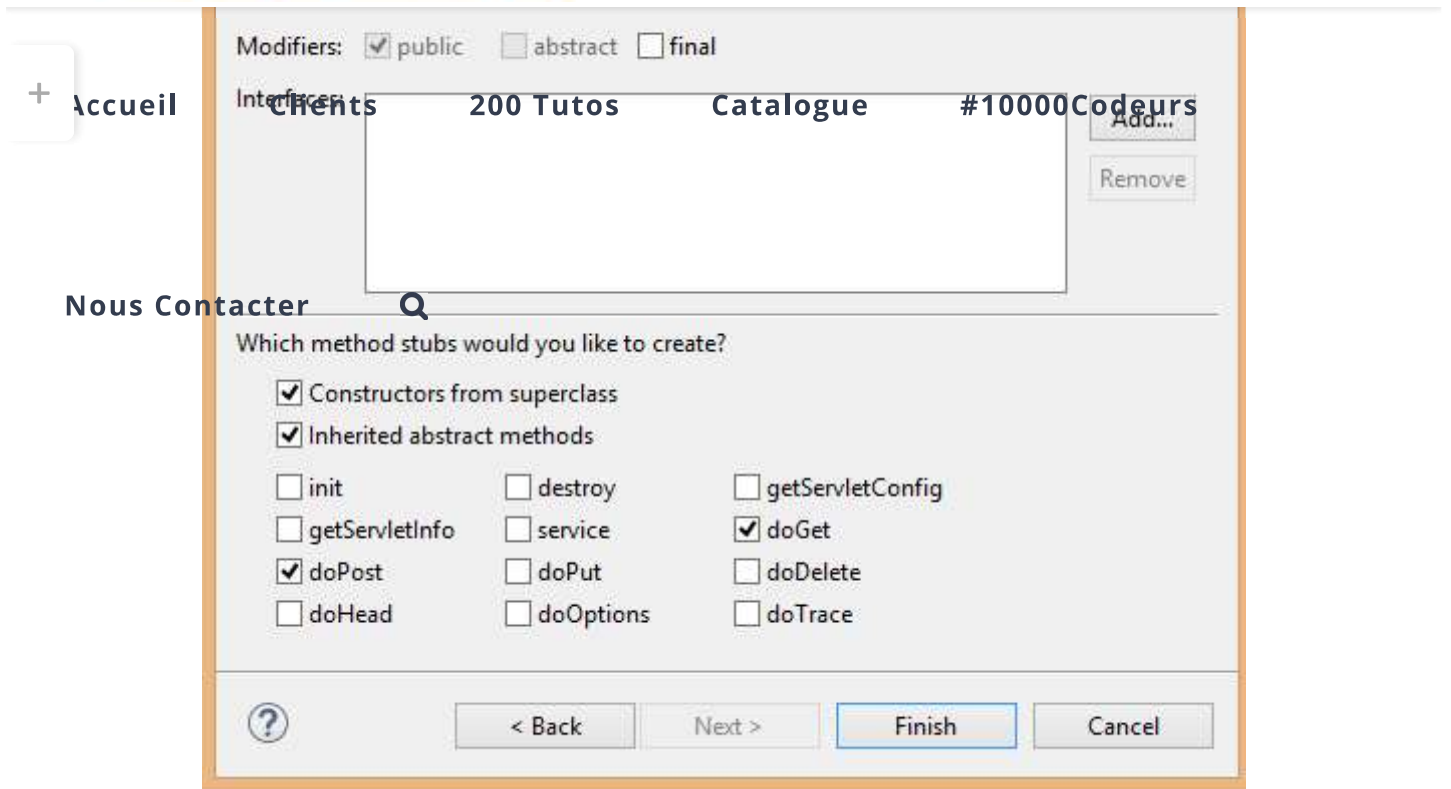
Cancel

Question

A quoi servent la section "URL Mappings"

Cliquez sur le bouton Next pour avoir l'écran suivant





En Cliquant sur le bouton Finish, Eclipse génère le code suivant

```

1 package com.objis.gestionformations;
2
3 import java.io.IOException;
4
5 /**
6  * Servlet implementation class MaServlet
7  */
8 @WebServlet("/MaServlet")
9 public class MaServlet extends HttpServlet {
10     private static final long serialVersionUID = 1L;
11
12     /**
13      * @see HttpServlet#HttpServlet()
14      */
15     public MaServlet() {
16         super();
17         // TODO Auto-generated constructor stub
18     }
19
20     /**
21      * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
22      */
23     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
24         // TODO Auto-generated method stub
25     }
26
27     /**
28      * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
29      */
30     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
31         // TODO Auto-generated method stub
32     }
33 }
34
35
36
37
38
39
40

```




+ Expliquez les paramètres de ces méthodes
[Accueil](#) [Clients](#) [200 Tutos](#) [Catalogue](#) [#10000Codeurs](#)

En utilisant [l'API JEE 7](#), expliquez e qu'est **HttpServletRequest** ? **HttpServletResponse** ?

Proposez 3 méthodes .

[Nous Contacter](#) 

Partie 2 : Configuration de la servlet

Expliquez le code de la ligne 13 : Annotation **@WebServlet(« /MaServlet »)**

```
10 /**
11  * Servlet implementation class MaServlet
12  */
13 @WebServlet("/MaServlet")
14 public class MaServlet extends HttpServlet {
```

Partie 3 : Codage de la servlet

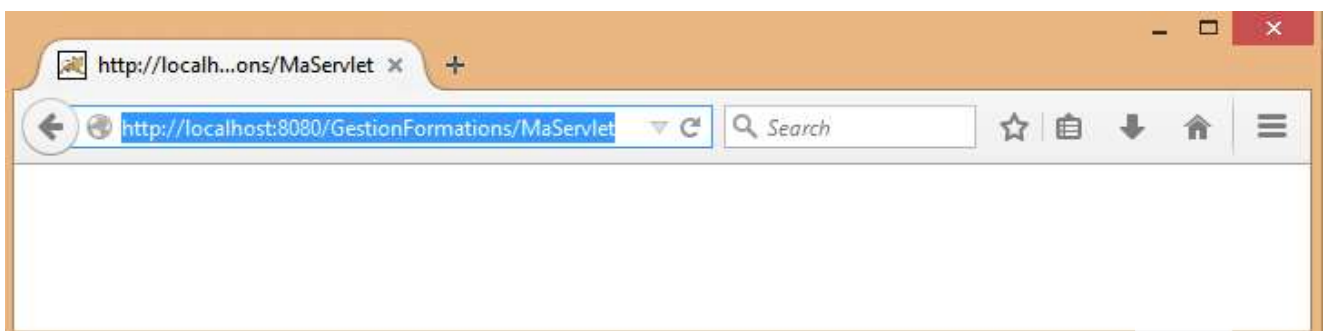
Ajouter le code suivant comme implémentation de la méthode *doGet()*

```
/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    System.out.println("ICI DANS LA SERVLET");
}
/**
```

Le codage de la servlet est terminé. Reste à déployer .

Partie 4 : Déploiement de la servlet

Quel est l'effet de l'URL suivante : <http://localhost:8080/GestionFormations/MaServlet> ?





Nous allons vous faire aimer Java

```
dec. 16, 2014 3:29:07 PM org.apache.coyote.AbstractProtocol start
INFOS: Starting ProtocolHandler ["ajp-nio-8009"]
déc. 16, 2014 3:29:07 PM org.apache.catalina.startup.Catalina start
INFOS: Server startup in 406 ms
ICI DANS LA SERVLET
```

+ [Accueil](#) [Clients](#) [200 Tutos](#) [Catalogue](#) [#10000Codeurs](#)

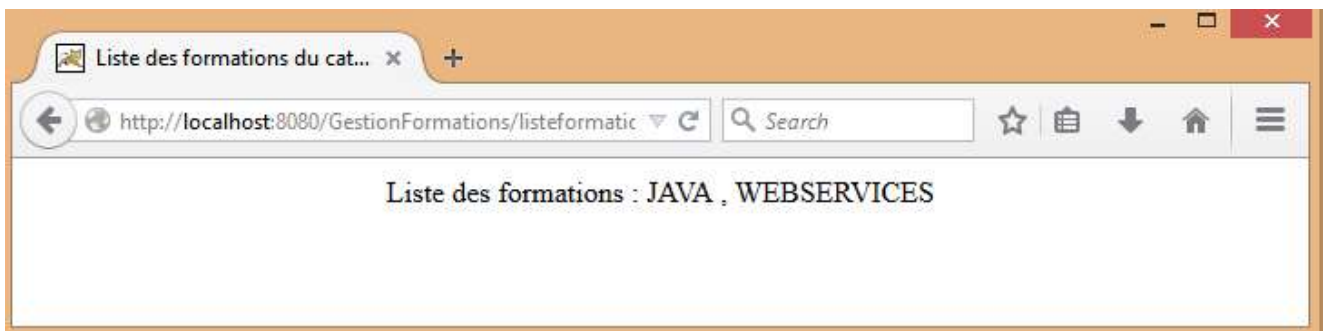
Expliquez !!!

Nous Contacter



Partie 5 :Refactoring

Modifiez la configuration et le code de la Servlet afin d'avoir le résultat suivant



ASTUCE : Analysez le code suivant et expliquez :

```
30- protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
31-     response.setContentType("text/html");
32-     PrintWriter out = response.getWriter();
33-     out.println("<HTML>");
34-     out.println("<HEAD><TITLE>Liste des formations du catalogue</TITLE></HEAD>");
35-     out.println("<BODY>");
36-     out.println("<CENTER>Liste des formations : JAVA , WEBSERVICES </CENTER>");
37-     out.println("</BODY>");
38-     out.println("</HTML>");
39- }
```

Partie 6 : Configuration de la servlet à travers le fichier web.xml

La même configuration que nous avons fait en utilisant les annotations

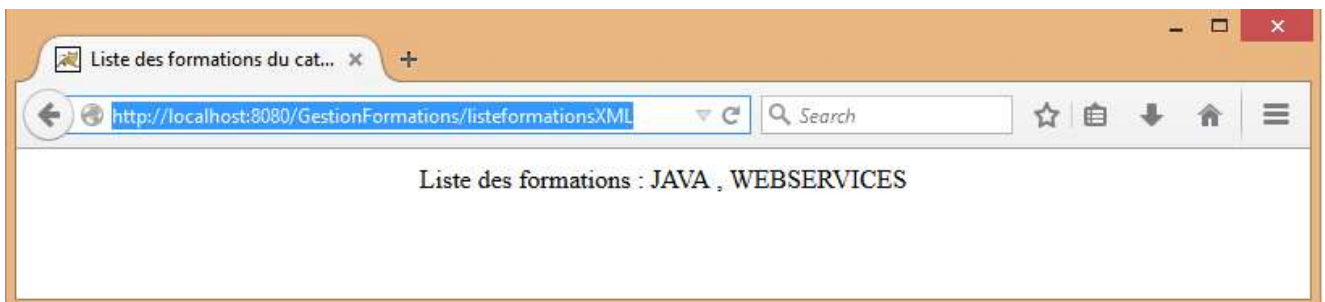
@WebServlet(« /listeformations »), nous pourrions aussi la faire en passant par notre fichier web.xml .





Nous avons changé l'*url-pattern* dans le fichier pour tester notre nouvelle url configurée sur le fichier web.xml .

L'URL, **http://localhost:8080/GestionFormations/listeformationsXML**, saisie dans le navigateur donne le résultat suivant



Remarque

1) Les annotations ont été introduites depuis JavaEE 5, cependant il reste toujours possible (mais facultatif) de configurer à travers le fichier web.xml

2) Vous ne pourrez pas avoir le même *url-pattern* avec l'annotation

```
@WebServlet("/listeformations")
```

que sur le fichier web.xml

```
<url-pattern>/listeformations</url-pattern>
```

Vous aurez une erreur au démarrage de Tomcat





Conclusion

Nous Contacter 

Vous avez codé et déployé votre première servlet sous Eclipse . Vous avez vu comment l'annotation `@WebServlet` permet de se dispenser de la configuration de la servlet dans le `web.xml` .

Enfin, vous avez implémenté la méthode `doGet()` .

Dans le prochain tutoriel vous allez apprendre à [Créer votre première JSP](#)

Démo

[GestionsFormations-servlet](#)

Votre avis nous intéresse

Vous avez remarqué une erreur ? Vous avez une recommandation pour améliorer ce tuto ?

Vous souhaitez nous encourager à partager notre savoir-faire technique et pédagogique ?

Donner votre avis sur ce Tuto objis.com

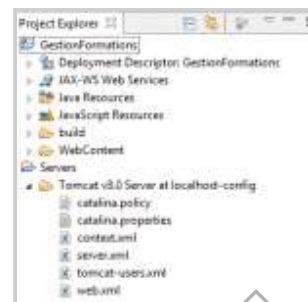
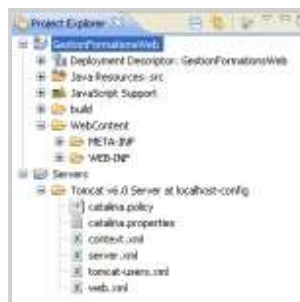
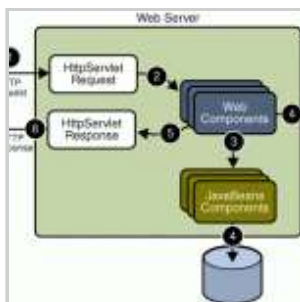
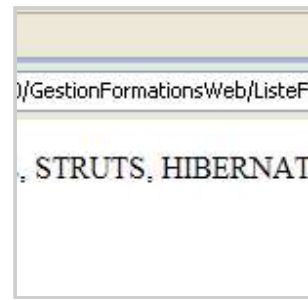
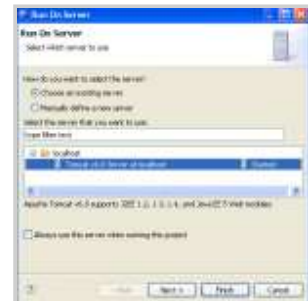
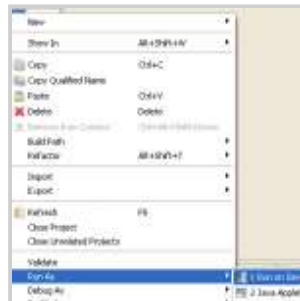
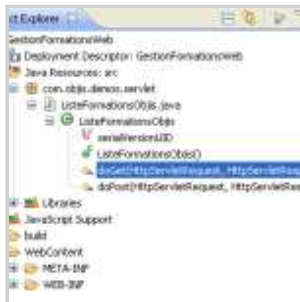


Objis

Nous allons vous faire aimer Java

+ Accueil Clients 200 Tutos Catalogue #10000Codeurs

Nous Contacter





+ Accueil

Clients

200 Tutos

Catalogue

#10000Codeurs

Nous Contacter

Q

OBJJS



Adresse
241 rue
DUGUESCLIN,
69003
Lyon



contact@objjs.com

LIENS LES PLUS VISITÉS

> Accueil
> Clients

FORMATIONS JAVA 1

Formation
JAVA
Formation

FORMATIONS JAVA 2

Formation
Javascript &
JQuery
Formation

AUTRES FORMATIONS

Formation
INTEGRATION
CONTINUE
avec



+	Accueil	Clients	> #10000Codeurs 200 Tutos	'Performance Java'	Fuse ESB	Sonar
				Catalogue	Formation	Formation
Nous Contacter			Q	> Nous	java	MAVEN
				Contacteur	architecture	Formation
				JAVA	Formation	Web Services
				Formation	java	Formation
				java	JBoss	GWT
				MAVEN	Formation	Formation
				Formation	JAVA	ExtJS
				Java	Entreprise	Conseil et
				STRUTS 2	(JEE)	formation
						ActiveMQ

Objis sas au capital de 49.500 euros | OBJIS Consulting, Centre d'affaires 241 rue DUGUESCLIN - 69003 Lyon
| RCS Lyon SIRET 48207506600027 | Organisme de formation enregistré sous le numéro 82.69.08812.69
préfecture région Rhone-Alpes. Cet enregistrement ne vaut pas agrément de l'Etat. | Tél : 07 63 03 11 08 .

Java (TM) est une marque déposée de la société SUN Microsystem.

