

# Atelier Framework Spring-01

## Maven-Partie-05

### Objectifs

#### Configuration de plugins avec Maven

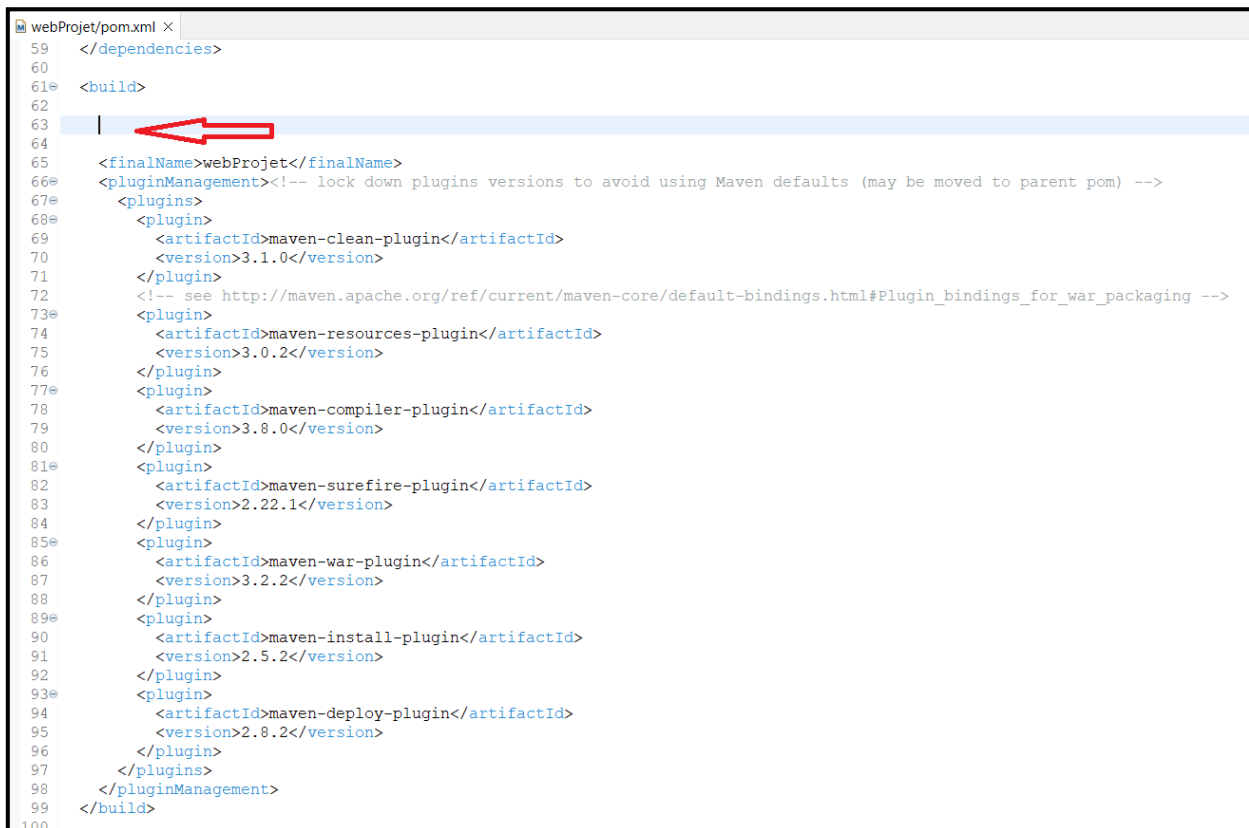
##### 1. Utiliser un plugin pour un serveur Tomcat embarqué

- déclarer le plugin dans le fichier « **pom.xml** »
- Lancer l'application via le plugin tomcat7
- Déployer le war à partir du plugin Maven vers le serveur Tomcat externe

##### 2. Utiliser un plugin pour le serveur Jetty

#### A. Configurer le plugin tomcat7 avec Maven (Eclipse)

1. Afin de déployer le projet dans l'EDI « **eclipse** » et visualiser l'exécution en utilisant un serveur Tomcat embarqué (interne), éditer le fichier « **pom.xml** » du projet « **webProjet** » pour ajouter le plugin du « **tomcat7** » dans la configuration Maven du projet à l'intérieur de la balise « **build** » :



```
59 </dependencies>
60
61 <build>
62
63 |
64
65 <finalName>webProjet</finalName>
66 <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->
67 <plugins>
68 <plugin>
69 <artifactId>maven-clean-plugin</artifactId>
70 <version>3.1.0</version>
71 </plugin>
72 <!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin_bindings_for_war_packaging -->
73 <plugin>
74 <artifactId>maven-resources-plugin</artifactId>
75 <version>3.0.2</version>
76 </plugin>
77 <plugin>
78 <artifactId>maven-compiler-plugin</artifactId>
79 <version>3.8.0</version>
80 </plugin>
81 <plugin>
82 <artifactId>maven-surefire-plugin</artifactId>
83 <version>2.22.1</version>
84 </plugin>
85 <plugin>
86 <artifactId>maven-war-plugin</artifactId>
87 <version>3.2.2</version>
88 </plugin>
89 <plugin>
90 <artifactId>maven-install-plugin</artifactId>
91 <version>2.5.2</version>
92 </plugin>
93 <plugin>
94 <artifactId>maven-deploy-plugin</artifactId>
95 <version>2.8.2</version>
96 </plugin>
97 </plugins>
98 </pluginManagement>
99 </build>
100
```

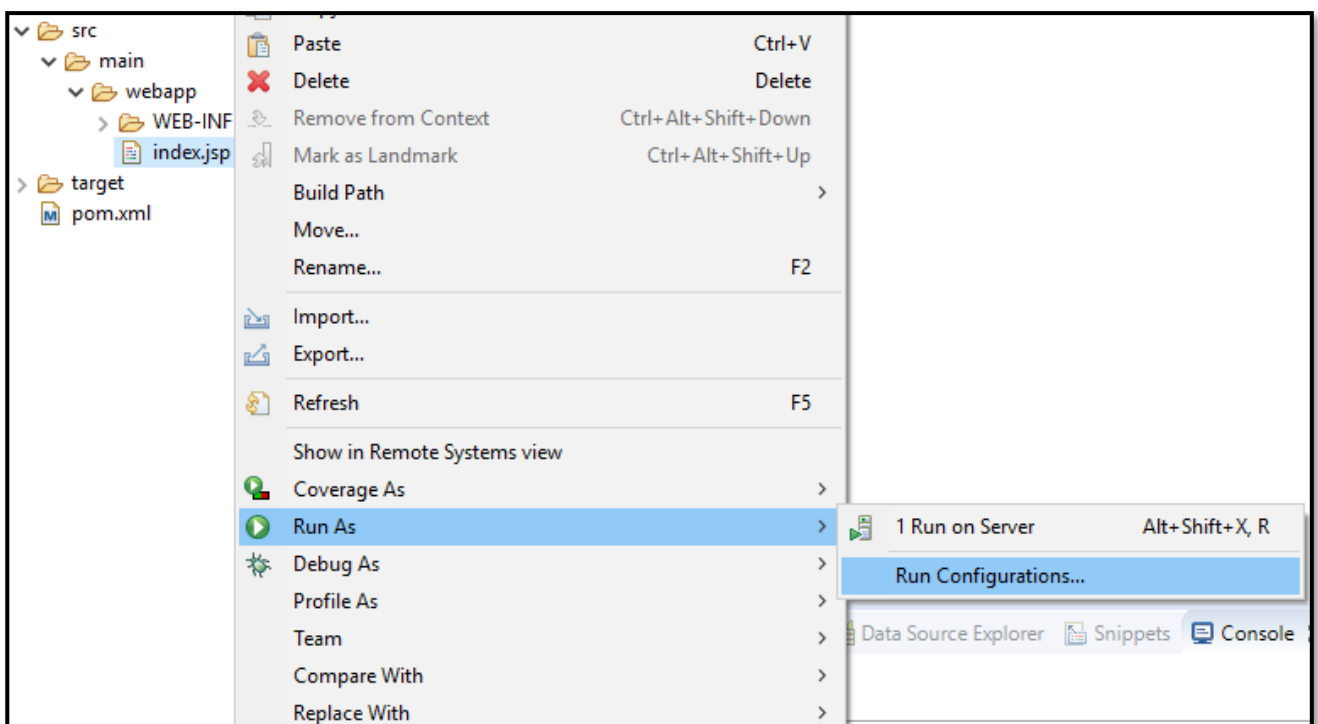
2. Ajouter les lignes en gras suivantes directement comme suit :

```
<build>

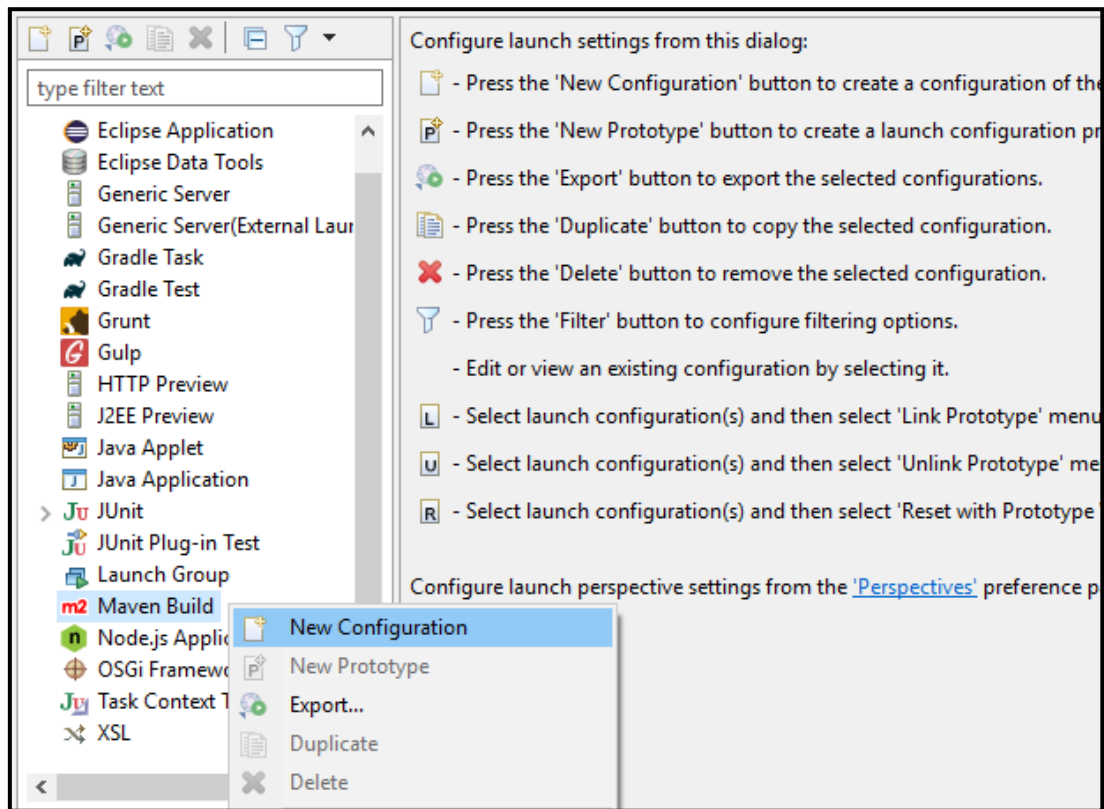
<plugins>
  <plugin>
    <groupId>org.apache.tomcat.maven</groupId>
    <artifactId>tomcat7-maven-plugin</artifactId>
    <version>2.2</version>
    <!--Config:contextPath and Port (Default:8080) -->
    <configuration>
      <path>/</path>
      <port>8899</port>
    </configuration>
  </plugin>
</plugins>

....
</build>
```

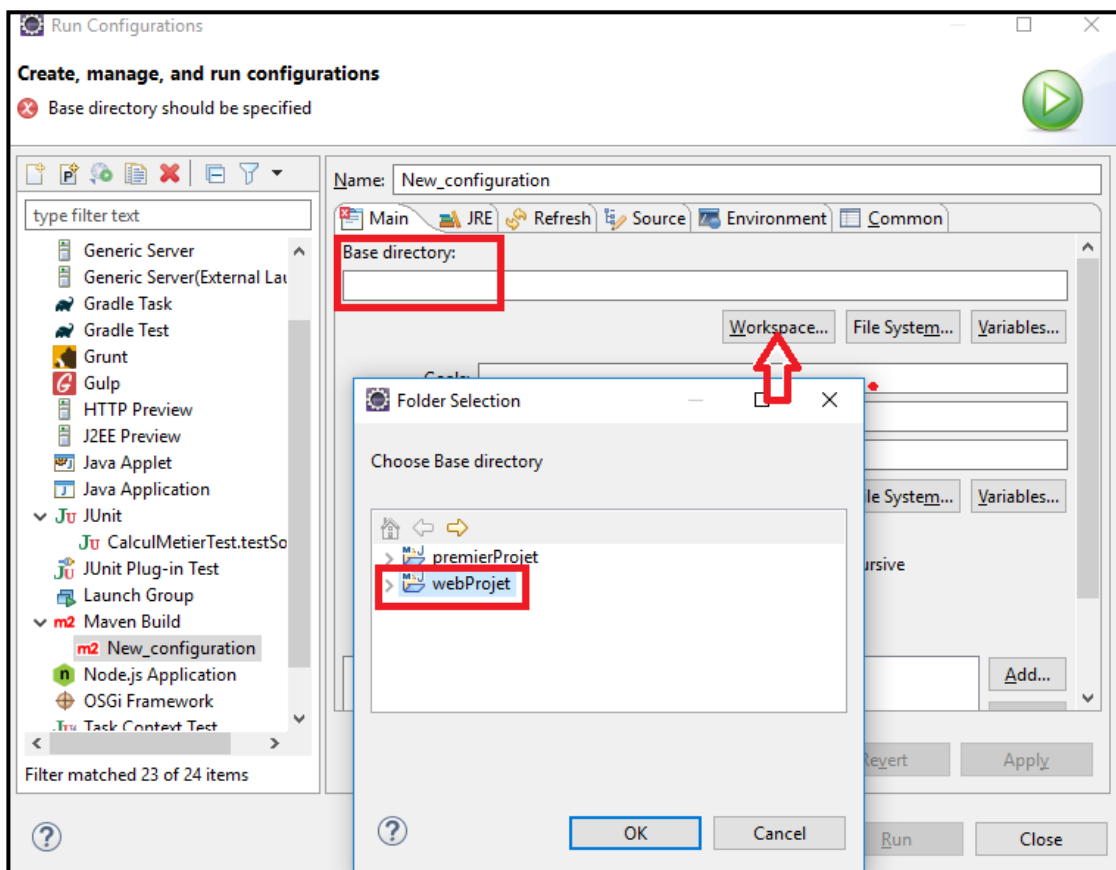
3. Sélectionner le fichier « **index.jsp** » et choisir ,en cliquant par le bouton droit de la souris, la commande « **Run As / Run Configurations...** » :



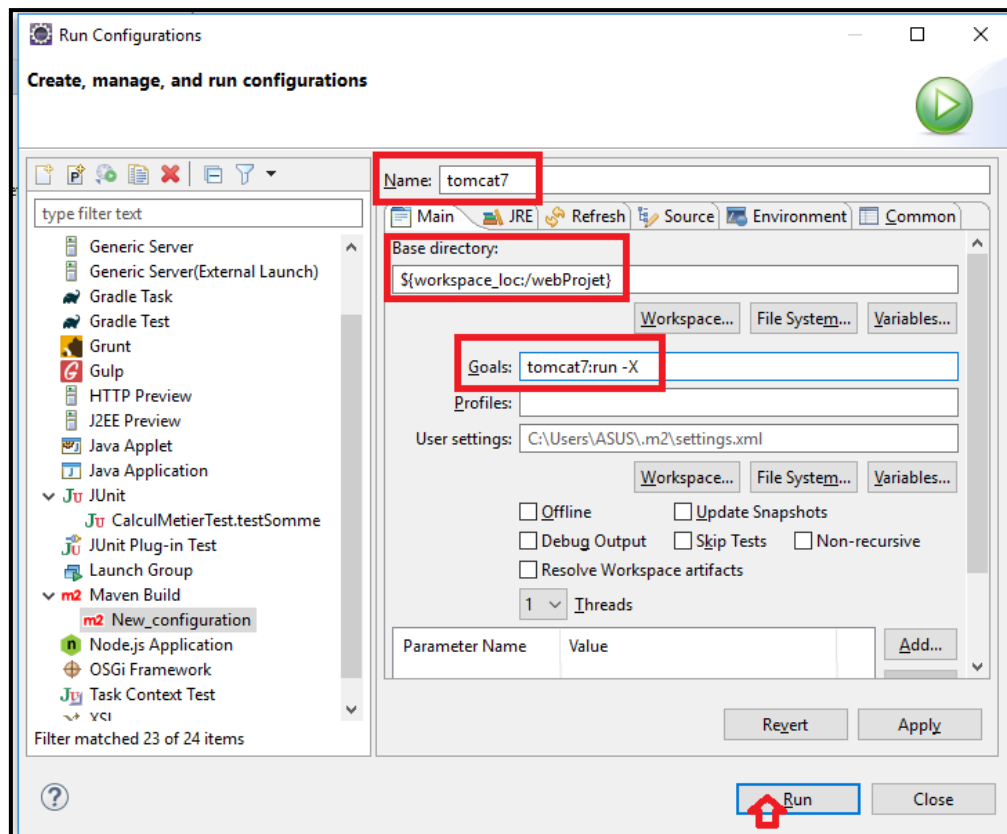
4. Sélectionner « **Maven Build** » Créer une nouvelle configuration d'exécution Maven (avec le bouton droit de la souris):



5. Cliquer sur le bouton « **Workspace** » pour sélectionner le projet « **webProjet** » comme dossier de base (Base directory) :



6. Spécifier un nom pour cette nouvelle configuration (tomcat7) et spécifier la valeur « **tomcat7:run -X** » pour « **Goals** » :

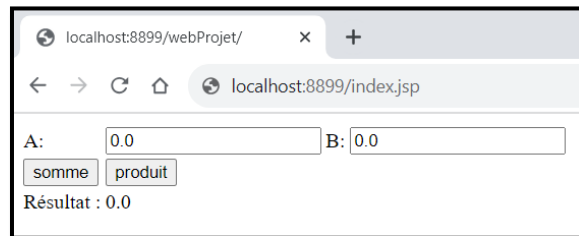


7. Lancer l'exécution en appuyant sur « **Run** ».
8. Après avoir téléchargé les modules nécessaires au plugin, la console affiche le succès de démarrage du plugin « **tomcat7** » :
- a. Sur le port : **8899** (selon la configuration du « pom.xml »)
  - b. avec le path : **/**
  - c. et l'url : **http://localhost:8899/**

```
[DEBUG] (f) warSourceDirectory = D:\Dev\Spring\workspace\webProjet\src\main\webapp
[DEBUG] (f) project = MavenProject: dev.ms.maven:webProjet:0.0.1-SNAPSHOT @ D:\Dev\Spring\workspace\webProjet\pom.xml
[DEBUG] (f) settings = org.apache.maven.execution.SettingsAdapter@5ba1b62e
[DEBUG] -- end configuration --
[INFO] Démarrage du war sur http://localhost:8899/
[INFO] Création de la configuration du serveur Tomcat sur D:\Dev\Spring\workspace\webProjet\target\tomcat
[INFO] create webapp with contextPath:
[DEBUG] context reloadable: false
[DEBUG] adding classPathElementFile file:/D:/Dev/Spring/workspace/webProjet/target/classes/
[DEBUG] add dependency to webapploader javax.servlet:jstl:1.2:compile
[DEBUG] add dependency to webapploader dev.ms.maven:premierProjet:1.0-SNAPSHOT:compile
août 20, 2025 1:48:47 PM org.apache.coyote.AbstractProtocol init
INFO: Initializing ProtocolHandler ["http-bio-8899"]
août 20, 2025 1:48:47 PM org.apache.catalina.core.StandardService startInternal
INFO: Starting service Tomcat
août 20, 2025 1:48:47 PM org.apache.catalina.core.StandardEngine startInternal
INFO: Starting Servlet Engine: Apache Tomcat/7.0.47
août 20, 2025 1:48:48 PM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-8899"]
```

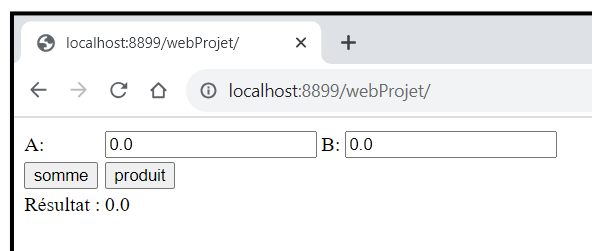
9. Accéder, via le navigateur web, à la page « **index.jsp** » en spécifiant l'url suivante :

<http://localhost:8899/index.jsp>

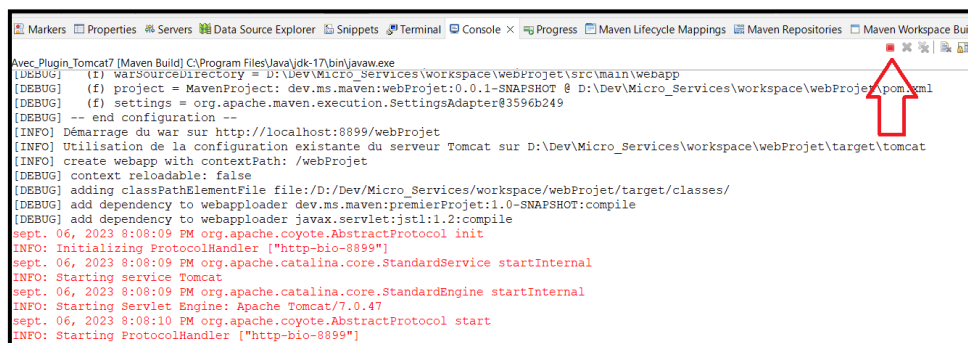


10. Réaliser la modification nécessaire de la configuration du plugin « **tomcat7** » dans le fichier « **pom.xml** » afin de démarrer l'application WEB sur :

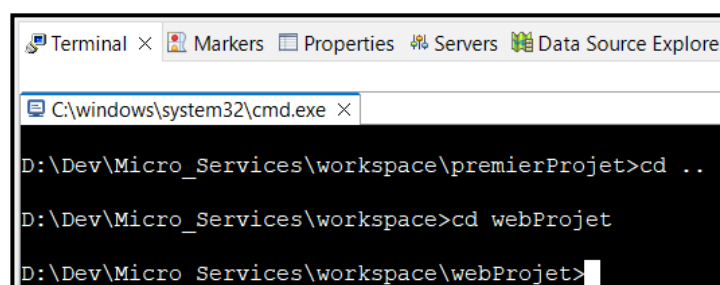
<http://localhost:8899/webProjet>



11. Pour démarrer le plugin de tomcat7 à travers le terminal :
- a. Arrêter le serveur tomcat7:



- b. Puis se positionner dans le terminal d'eclipse dans le dossier « **webProjet** » :

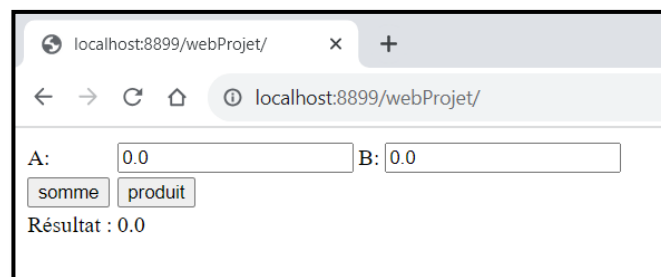


c. Taper la commande suivante :

**mvn tomcat7:run**

```
C:\WINDOWS\system32\cmd.exe - mvn tomcat7:run X
[INFO] --- tomcat7:2.2:run (default-cli) @ webProjet ---
[INFO] D:\marrage du war sur http://localhost:8899/webProjet
[INFO] Utilisation de la configuration existante du serveur Tomcat sur D:\Dev\Spring\workspace\webProjet\target\tomcat
[INFO] create webapp with contextPath: /webProjet
ao|_t 20, 2025 1:53:45 PM org.apache.coyote.AbstractProtocol init
INFO: Initializing ProtocolHandler ["http-bio-8899"]
ao|_t 20, 2025 1:53:45 PM org.apache.catalina.core.StandardService startInternal
INFO: Starting service Tomcat
ao|_t 20, 2025 1:53:45 PM org.apache.catalina.core.StandardEngine startInternal
INFO: Starting Servlet Engine: Apache Tomcat/7.0.47
ao|_t 20, 2025 1:53:46 PM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-8899"]
```

d. Tester l'exécution de l'application WEB



12. Maintenant, déployer l'application WEB directement sur le serveur Tomcat standalone à travers le plugin tomcat7 de Maven. Pour se faire :

a. Démarrer le serveur externe (**Tomcat9**) à partir de la commande «**startup** » dans le dossier ( s'il n'est pas encore démarré) :

**D:\Dev\Spring\Serveurs\apache-tomcat-9.0.108\bin**

b. Aller à la page d'administration des applications WEB pour retirer l'application WEB « **webProjet** » de la liste des applications déployées :

→ ↺ ⓘ localhost:8080/manager/html

<

c. Arrêter le serveur du plugin tomcat7 (dans le volet Termina) :  
**Tapar Ctrl+c**

d. Se positionner, au niveau Terminal dans le dossier **webProjet** puis taper la commande :

```
mvn tomcat7:deploy -Dtomcat.password=admin -Dtomcat.username=admin
```

```
[INFO] --- tomcat7:2.2:deploy (default-cli) @ webProjet ---
[INFO] Deployment du war sur http://localhost:8080/webProjet
Uploading: http://localhost:8080/manager/text/deploy?path=%2FwebProjet
Uploaded: http://localhost:8080/manager/text/deploy?path=%2FwebProjet (365 KB at 20272.4 KB/sec)
[INFO] tomcatManager status code:200, ReasonPhrase:
[INFO] OK - Application deployment pour le chemin de contexte [/webProjet]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.580 s
```

e. Remarquer le déploiement de l'application « webProjet » dans le serveur externe Tomcat :

Applications					
Chemin	Version	Nom d'affichage	Fonctionnelle	Sessions	Commandes
/	Aucun spécifié	Welcome to Tomcat	true	0	Démarrer Arrêter Recharger Retirer Expier les sessions inactives depuis ≥ 30 minutes
/docs	Aucun spécifié	Tomcat Documentation	true	0	Démarrer Arrêter Recharger Retirer Expier les sessions inactives depuis ≥ 30 minutes
/examples	Aucun spécifié	Servlet and JSP Examples	true	0	Démarrer Arrêter Recharger Retirer Expier les sessions inactives depuis ≥ 30 minutes
/host-manager	Aucun spécifié	Tomcat Host Manager Application	true	0	Démarrer Arrêter Recharger Retirer Expier les sessions inactives depuis ≥ 30 minutes
/manager	Aucun spécifié	Tomcat Manager Application	true	1	Démarrer Arrêter Recharger Retirer Expier les sessions inactives depuis ≥ 30 minutes
webProjet	Aucun spécifié	Archetype Created Web Application	true	0	Démarrer Arrêter Recharger Retirer Expier les sessions inactives depuis ≥ 30 minutes

*Il est à noter que le war est déployé directement du dossier « target » du projet et nom à partir du dépôt local de Maven.*

*NB : La commande `mvn tomcat7:deploy` ne fonctionne que si le serveur externe (Tomcat9) est écoute sur le port par défaut : 8080. Sinon, il est nécessaire de modifier la configuration du plugin tomcat7 pour qu'il puisse accéder au serveur externe avec le port personnalisé.*

f. L'opération inverse est aussi possible : retirer l'application web déployée. Tester ceci via la commande :

```
mvn tomcat7:undeploy -Dtomcat.password=admin -Dtomcat.username=admin
```

## B. Configurer le plugin Jetty avec Maven (Eclipse)

13. Sur le même projet, refaire la même procédure pour ajouter un autre plugin similaire pour un autre serveur web « **Jetty** ». Voici le code configuration du plugin : (Ajouter ce code dans le même niveau que celui du plugin du tomcat7, à l'intérieur de la balsaie « plugins » )

```
<!-- Jetty Plugin. Default port is 8080 -->  
<plugin>  
  <groupId>org.eclipse.jetty</groupId>  
  <artifactId>jetty-maven-plugin</artifactId>  
  <version>11.0.16</version>  
  <configuration>  
    <httpConnector>  
      <!--host>localhost</host-->  
      <port>9999</port>  
    </httpConnector>  
  </configuration>  
</plugin>
```

14. Lancer l'exécution du projet avec le plugin « **Jetty** » sur l'url :

**`http://localhost:9999/index.jsp`**

## Exercice

Réaliser un déploiement du projet «webProjet» vers un serveur externe GlassFish.