

SU – Master 2 – UE DEVREP

JEE – TP 1

Nous vous proposons de commencer la partie JEE de DevReps par la prise en main des différents environnements, afin de se familiariser avec toutes les configurations à faire pour développer des applications JEE. La procédure sera d'appliquer des tutoriels¹.

Faites le tutoriel proposé dans le document JEE-TP1-SimpleApp.pdf. Les exercices proposés vous feront créer une application simple utilisant un formulaire, en utilisant les technologies JSP, Servlet et JSF. Cela donnera lieu à trois mini-applications.

Pour réaliser ce tutorial, installez les outils suivants :

- Apache Tomcat : Installez Tomcat 8.x Core (à partir de 8.5, ou 9) à partir de <https://tomcat.apache.org>. Archive à télécharger et décompresser.
- JDK 8+
- Eclipse JEE (<https://www.eclipse.org/downloads/packages/release/2021-09/r/eclipse-ide-enterprise-java-and-web-developers>)
- Apache Maven (<https://maven.apache.org/>) s'installe simplement en décompressant l'archive.

Recommandations

Pour ce tutoriel, laissez de côté l'installation de Glassfish et de MySQL ; vous installerez ces outils au moment où vous en aurez besoin lors des prochains TP. Cependant, faites, a minima, l'installation de Apache Tomcat avant de commencer ce TP 1. Vous pourrez installer Maven vers la fin du tutoriel, après le troisième exercice.

Lorsqu'on vous demande de créer des packages, par exemple : pack.book.jee_eclipse.ch2.bean, vous pouvez remplacer ce nom par un nom plus simple, par exemple : fr.su.jee.bean., fr.su.jee.servlet, fr.su.jsf.beans

Dans le troisième exercice utilisant JSF, à propos des *ManagedBeans* qui sont « deprecated », consultez les liens suivants pour comprendre le problème :

- <https://www.oracle.com/technical-resources/articles/java/cdi-javaee-bien.html>
- <https://netbeans.apache.org/kb/docs/javaee/cdi-inject.html>
- <https://www.theserverside.com/tutorial/Working-with-CDI-and-JSF-20-on-Tomcat-7-Configuring-Weld>
- <https://stackoverflow.com/questions/9921324/javax-faces-bean-managedproperty-in-cdi-named-bean-returns-null/9921449#9921449>

¹ Extraits du livre : Java EE Development with Eclipse 3rd edition

- <https://stackoverflow.com/questions/10994158/difference-between-named-and-managedbean-annotations-in-jsf2-0-tomcat7?noredirect=1&lq=1>
- <https://stackoverflow.com/questions/4347374/backing-beans-managedbean-or-cdi-beans-named>

Puis, créez un quatrième projet (en y copiant les ressources utiles du troisième exercice) dans lequel vous essaieriez d'utiliser CDI et ses annotations (en l'occurrence `@java.inject.Named` et `@javax.enterprise.context.RequestScoped`) au lieu de ceux de JSF. En changeant de Serveur d'application pour le nouveau projet avec un serveur qui supporte CDI, e.g, Wildfly 24+ de RedHat JBoss (au lieu de Tomcat), vous devriez pouvoir redéployer l'application en utilisant les annotations CDI.

Quelques ressources supplémentaires sur CDI :

- <https://blogs.oracle.com/javamagazine/post/java-jakarta-cdi-interceptor-qualifier-producer>
- <https://blogs.oracle.com/javamagazine/post/java-jakarta-cdi-stereotype-decorator-spi-lite>