

Deployment e caricamento di
un'applicazione web su Tomcat

Deployment lato programmatore web

1. Esporta la tua applicazione come file .war (web archive) creando un artefatto apposito in IntelliJ (o tramite Export.. in Eclipse)
2. Fai il deployment del file .war su Tomcat
 - Copiando il file .war in tomcat/webapps, oppure
 - Tramite manager-gui della home page di Tomcat

Deployment lato programmatore web

1. Esporta la tua applicazione come file .war (web archive) creando un artefatto apposito in IntelliJ (o tramite Export.. in Eclipse)

In IntelliJ, vai su *Build/Build Artifacts..* e seleziona *All Artifacts*

Il .war si troverà nella cartella arancione sulla sinistra chiamata *target*

Deployment lato programmatore web 1/2

2. Fai il deployment del file .war su Tomcat

- Tramite manager-gui della home page di Tomcat, oppure

a. Fissa la password per il manager-gui di Tomcat

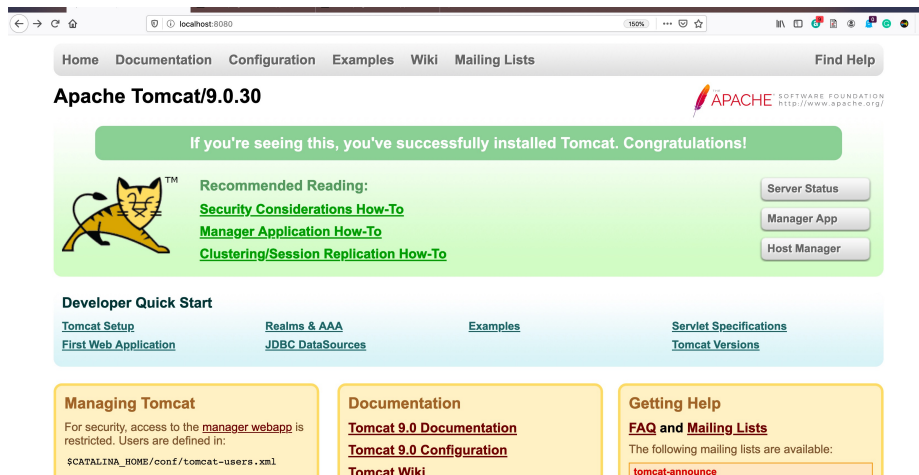
1. Apri il file `<tomcatdir>/conf/tomcat-user.xml` in un editore di testo

2. Aggiungi il seguente prima del testo `</tomcat-users>` a fine file:

```
<role rolename="manager-gui"/>
<role rolename="manager-script"/>
<role rolename="manager-status"/>
<user username="root" password="root" roles="manager-gui,manager-
script,manager-status"/>
```

b. Con Tomcat attivo, vai alla pagina <http://localhost:8080> e clicca su bottone «Manager app» e quindi si faccia upload del file .war nella sezione «WAR file to deploy»

Deployment lato programmatore web 2/2



/modelloMVC_war	None specified	true	0	<div> <div>Start</div> <div>Stop</div> <div>Reload</div> <div>Undeploy</div> </div> <div> <div>Expire sessions</div> <div>with idle ≥ 30 minutes</div> </div>
/modelloMVCdbPref_war	None specified	true	0	<div> <div>Start</div> <div>Stop</div> <div>Reload</div> <div>Undeploy</div> </div> <div> <div>Expire sessions</div> <div>with idle ≥ 30 minutes</div> </div>
/modelloMVCdbPref_war_explored	None specified	false	0	<div> <div>Start</div> <div>Stop</div> <div>Reload</div> <div>Undeploy</div> </div>

Deploy	
Deploy directory or WAR file located on server	
Context Path:	<input type="text"/>
Version (for parallel deployment):	<input type="text"/>
XML Configuration file path:	<input type="text"/>
WAR or Directory path:	<input type="text"/>
<input type="button" value="Deploy"/>	
WAR file to deploy	
Select WAR file to upload	<input type="button" value="Sfoglia..."/> Nessun file selezionato.
<input type="button" value="Deploy"/>	
Configuration	
Re-read TLS configuration files	
TLS host name (optional)	<input type="text"/>
<input type="button" value="Re-read"/>	

Deployment di applicazione da parte di Tomcat

- Registrazione di tutte le servlet e jsp dell'applicazione tramite lettura di web.xml e annotations
 - Ad es. ogni servlet viene associata al suo url-pattern, viene settato il load-on-startup, etc.
- Caricamento, istanziazione ed inizializzazione (chiamata ad init()) di ciascuna **servlet** che nella registrazione presenti un valore per **load-on-startup maggiore o uguale a 0**. L'ordine di caricamento dipende dal valore (valori maggiori vengono caricati per ultimo). Il default per load-on-startup è -1.
- Le **servlet** che sono registrate con **load-on-start non settato o settato con intero negativo**, sono caricate, istanziate ed inizializzate quando vuole il server (al deployment o alla prima chiamata)
- Le **jsp** vengono compilate, caricate, istanziate ed inizializzate (ed anche eseguite) solo alla prima chiamata. I compilati sono salvati su hd per cui, alla caduta e ripresa del server la prima chiamata evita la compilazione.