

Deploying web apps to Tomcat

Devdatta Kulkarni

Approaches

- Manually deploy
- Deploy via Eclipse

Tomcat

- Web server
- Important directories
 - bin: contains scripts to start and shutdown tomcat
 - conf: contains configuration files
 - logs: contains logs (relevant file is catalina.out)
 - webapps: contains web applications deployed on tomcat
- Important files and variables
 - conf/server.xml has Connector for HTTP/1.1 which controls what port Tomcat receives HTTP/1.1 requests

Deploying a web app directly to Tomcat

- Assumptions:
 - Project is managed using maven (i.e. there is a pom.xml in the project's folder)
 - Maven is installed on the machine
- Go to project directory
- Run “mvn war:war”
 - This will generate a war file in the “target” directory
 - The default name of the war file has following pattern
 - <artifactId>-<version>, where artifactId and version are as defined in the pom.xml
 - If you want a specific name for the war file, add the “finalName” attribute in the “build” section of pom.xml
- Context Root: This is essentially the name of the war file.
- Copy the war file to tomcat's webapps directory
- Start Tomcat by running bin/startup.sh
- Once Tomcat has started, it will expand the war file into a directory. You can observe this by running “ls -l” inside the webapps directory
- If there are no errors, your web application will be available at:
 - `http://localhost:<connector port>/<context-root>/<url-pattern>`
- Check the logs
 - `more catalina.out, less catalina.out, tail -f catalina.out`

Deploying web app through Eclipse

- Setup Eclipse to use Tomcat as the web server
 - Eclipse → Window → Preferences → Servers → Runtime Environment → Add
 - Provide a name to the “New Server Runtime environment”
 - On the next screen, provide path of Tomcat Installation Directory (the directory where you downloaded and unzipped Tomcat)
- Running a Servlet project
 - Right click on the project
 - Select ‘Run As’ -> ‘Run on Server’
 - Choose the Tomcat you configured
 - Eclipse will start Tomcat and deploy your web application / Servlet to it
 - You can change the port on which Tomcat runs by clicking “Servers” tab and double clicking the Server that you are using
- Servlet will be now accessible at:
 - `http://localhost:8080/<context_root>/<url-pattern>`
- Context Root
 - Right click on the project
 - Web Project settings
 - Note that the context-root is not affected by presence of pom.xml
- url-pattern
 - specified in web.xml