**Server – code**

1. **Configure path**

# go to start > envi...

Machine generated alternative text: Edit the system environment variables
Control panel

# create a new **system variable** (not user variable)

Machine generated alternative text: Edit System Variable X
Variable name: JAVA_HOME
Variable value: C:\Program Files\Java\jdkl.8.O_202 ,
Browse Directory [ Browse File OK Cancel

Machine generated alternative text: System variables
Variable Value
JAVA_HOME C:\Program Files\Java\jdkl .8.0_202

# add the variable to the **system path**

Machine generated alternative text: Edit environment variable
C:\Program Files (x86)\Common Files\Oracle\Java\javapath
C:\ProgramData\DockerDesktop\version-bin
C:\Program Files\Docker\Docker\Resources\bin
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEM ROOT%\System32\WindowsPowerShell\vl .O\
C:\Program Files (x86)\Sennheiser\HeadSetup\
C:\Program Files\nodejs\
C:\PostgreSQL\pglO\bin - --- ______
%JAVA...HOPV1
in
C:\Program Files\MongoDB\Server\4.O\bin
%CATALI NAHOME%\bin

Check that you have the following:

JAVA\_HOME - path to JDK

M2\_HOME - path to Maven

CATALINA\_HOME - path to Tomcat

Path - %JAVA\_HOME%\bin; %M2\_HOME%\bin; %CATALINA\_HOME%\bin;

1. **Mobaxterm**

|  |  |
| --- | --- |
| Download | <https://mobaxterm.mobatek.net/download.html> |
|  |  |
| config | **# configure mobaxterm to use windows path**    Machine generated alternative text: |
| sessions | **# create a new session**    right click on user sessions > new session    Machine generated alternative text: New session New folder Clear saved sessions    chose shell    terminal shell: cmd    startup directory: you can select a folder where you want your session to start at  \* advanced shell settings(optional) : you can write commands that will be executed at startup  Machine generated alternative text: Cmd__..r    click bookmark settings  session name:  session icon  Machine generated alternative text: javac    click ok to save   now double click the saved session to open a terminal at the selected directory  if you configured mobaxterm correctly you will be able to use java commands    Machine generated alternative text: User sessions • javac |

1. **TcpView**

|  |  |
| --- | --- |
| download | [**https://tcpview.en.softonic.com/**](https://tcpview.en.softonic.com/) |
| demo | **# open tcp view and see the ports on which the processes run**    **Machine generated alternative text: TCPView - Sysinternals: www.sysinternals.com Options Process View Help Process Pl D Protocol Local Address Local Port R emote Address R emote Port State ï svchosiexe 1116 UDP WX-5CG838&<50 64745 N N ï ><Win_MobK 18536 TOE WX5CG838&<5O 64682 localhost 6000 ESTABLISHED ï >(VinMob&C.. 18536 TOE \*%5CG838EX50 64681 localhost 6000 ESTABLISHED ><Win_MobaX,. 18536 TOE WX5C6838EX50 64680 localhost 6000 ESTABLISHED ideaß4exe 12760 UDP \%5CG838&<5O 64486 X ij Skype.exe 10872 UDP WX-5CG839&’<50 63824 * Skype.exe 10872 UDPV6 [0:0Œ0:0:0:&0j 63824 N idea64. exc 12760 T CP \ÑX-5CG838&<50 63342 WX5CG 838&<50 O LISTENING idea64.exe 12760 TOE w%5CG838EX50 63342 localhost 56666 ESTABLISHED CcmE xec. cxc 15436 UDP WX-5C6838&<50 62882 * idea64.exe \wX-5CG838&’<SO 60913 idca64 cxc WX-5C6839&<50 60718 * ———----- --—- * avacxc UUfr-’ ... .  .. ¡Jjj ¡ava.exe UDPV6 [0:OE0:0:O:O:0:0J 59241 ¶Sj Iync,cxe 948 UDP WX•5CG838&<50 58208 * I rn n n n n n n ii rir’nå .**    **# end a processes**    **if you want to start an application server on a particular port and you get the 'Address already in use' error, you can identify which process is using that port and end the process.** |

1. **Tomcat**

|  |  |
| --- | --- |
| intro | TomCat, in its simplest concept, is a web server. Well, it's more than that. While it can serve static files, its primary purpose is to act as a 'servlet container' that serves Java web applications. It can process [.jsp] files, which are like PHP scripts but for Java, and it can also run [Java Servlets], which are classes that process the GET, POST, and other HTTP requests. TomCat will listen on TCP ports for HTTP requests, and route requests to your Java classes, JSP files, or static files. It is also possible to embed TomCat in to a standalone application, but that is out of the scope of this document. |
| download | <https://tomcat.apache.org/download-90.cgi> |
| specifications | Machine generated alternative text: Serviet Spec JSP Spec EL Spec WebSocket Spec JASPIC Spec Apache Tomcat Version Latest Released Version Supported Java Versions 4.0 2.3 3.0 1.1 1.1 9.0.x 9.0.24 8and later 3.1 2.3 3.0 1.1 1.1 8.5.x 8.5.45 7and later 3.1 2.3 3.0 1.1 N/A 8.0.x (superseded) 8.0.53 (superseded) 7 and later 3.0 2.2 2.2 1.1 N/A 7.0.x 7.0.96 6and later (7 and later for WebSocket) 2.5 2.1 2.1 N/A N/A 6.0.x (archived) 6.0.53 (archived) 5 and later 2.4 2.0 N/A N/A N/A 5.5.x (archived) 5.5.36 (archived) 1.4 and later 2.3 1.2 N/A N/A N/A 4.1.x (archived) 4.1.40 (archived) 1.3 and later 2.2 1.1 N/A N/A N/A 3.3.x (archived) 3.3.2 (archived) 1.1 and later |
| serve static files | TomCat can handle serving static files out of the box. By default there is a webapps/ROOT/ directory in the TomCat's webapps/ directory. There is already an index.jsp and some other resources inside. That is the landing page you hit when you visited the TomCat server in the browser in the previous section at http://<ip>:8080    You can add more static files to the webapps/ROOT/ directory, You can create a new directory inside webapps/ directory and place static files there. accessible at /<directory\_name> (e.g. <http://localhost:8080/mywebapp>).    You can put flat files like .txt, .zip or .html and they will be served directly. You can also create subdirectories.    C:\Users\cosmin.bucur\Documents\apache-tomcat-9.0.24\webapps |
| serve JSP | Serving .jsp pages is just as easy as serving static files. You just drop in JSP files in a webapp directory like ROOT and point your client at it. There are no XML configurations or WAR files needed. Just point your browser to /yourjsp.jsp and it will run. TomCat will actually compile them and process them dynamically before returning the response to the client. They are treated like PHP scripts in that sense. They can be treated like HTML files with dynamic Java code inserted at various places. There is no need to restart the server or recompile code when making updates to .jsp pages.     <!-- hello.jsp -->  <html>  <body>  <p><% out.print("Hello, world!"); %></p>  <p>8 x 8 = <%= 8 \* 8 %></p>  <p>Time: <%= new java.util.Date() %></p>  <p>Host requested: <%= request.getHeader("Host") %></p>  </body>  </html>    Name it hello.jsp and drop it in the webapps directory, and then visit <https://localhost:8080/sample_app/hello.jsp>. Or, create a directory like sample\_app/ inside webapps and put the file in there, then visit <http://localhost:8080/sample_app/hello.jsp>. |
| modify settings | TomCat configuration files live in the conf/ directory of the root TomCat directory. The main file is server.xml and that is where you would change the listen port. Inside server.xml the main listening port is defined in a section labeled Connector. The section will look something like this:    <Connector port="8080" protocol="HTTP/1.1"  connectionTimeout="20000"  redirectPort="8443" > |
| deploy web apps | Deploying web apps can be done in a number of ways which are listed below.    Manually copying the web app directory to the TomCat's webapps/ directory  Placing a .war file in the webapps/ directory  Using Maven plugin to deploy  Using the web app named manager that comes packaged with TomCat |
| use the admin web interface | There are two web apps that come bundled with TomCat to help with administration.    One is the Virtual Host Manager which lets you listen for multiple different domains. This can be accessed at /host-manager/html and requires certain roles to access (explained below).    The other web app that comes bundled with TomCat is the Web Application Manager. This one is used more often and is more relevant to beginners. This one can be accessed at /manager/html. This one lists al the web apps that are available and which ones are running. You can start, start, deploy, undeploy, and expire sessions. It is quite useful. |
| create a tomcat admin user | You can add users to TomCat in the conf/tomcat-users.xml file. There will be a section in the xml labeled tomcat-users. Inside of that there will be several commented out examples showing how to create a user. If you want to grant all roles that allow access to the admin manager app, you can add the lines of code below, changing the username and password.    C:\Users\cosmin.bucur\Documents\apache-tomcat-9.0.24\conf    Machine generated alternative text: <!-- Inside <CATALINA_HOME>/conf/tomcat-users.xml --> <!-- Inside the <tomcat-users> element --> <!-- Roles for Web Application Manager /manager/html! --> <role rolename=”manager-gui” ,> <role rolename=”manager-script” ,> <role rolename=”manager-jnx” ,> _____ <role rolename=”manager-status” ,> <!-- Role for Virtual Host Manager /host-manager/html/ --> <role rolename=”admin-gui” I> <!-- Create user and assign roles --> <user username=”dano” password=”secret” roles=”manager-script,manager-gui,manager-jmx,manager-status,admin-gui” ,> E7__________ r    <!-- Inside <CATALINA\_HOME>/conf/tomcat-users.xml -->  <!-- Inside the <tomcat-users> element -->    <!-- Roles for Web Application Manager /manager/html/ -->  <role rolename="manager-gui" />  <role rolename="manager-script" />  <role rolename="manager-jmx" />  <role rolename="manager-status" />  <!-- Role for Virtual Host Manager /host-manager/html/ -->  <role rolename="admin-gui" />    <!-- Create user and assign roles -->  <user username="admin" password="secret" roles="manager-script,manager-gui,manager-jmx,manager-status,admin-gui" />      restart server |
| access admin web apps | Machine generated alternative text: Home Documentation Configuration Examples Wiki Mailing Lists Find Help Apache Tomcatl8.O.33 Recommended Reading: Security Considerations HOW-TO Manager Application HOW-TO ClusterinplSesslon Replication HOW-TO pache Software Foundation http://www.apache.org/ !— I I Server Status Manager App Host Manager Developer Quick Start Tomcat Setup First Web Application Realms & AAA JDBC DataSources Examples Serviet Specifications Tomcat Versions Let’s take a look at the Manager App, accessible via the link or http://server_domain_or_IP:8080/manager!html. You will need to enter the account credentials that you added to the tomcat-users .xml file. Afterwards, you should see a page that looks like this:    Machine generated alternative text: M.ssg.: 10K Tomcat Web Application Manager Start _Slop] Reload LUflPloYJ Expire sessions with idle  30 Start Stop j ¡ Reload Undeploy Expiro sessions with idle  Start Slop Reload Undeploy Expire sessions with idle  30 irt Slop Reload Undeploy Expire sessions with idle  30 Start Stop Reload undoploy Expire sessions with Idle  30 |
| load a war file | clone sample web app from    <https://github.com/cosminbucur/sample-web-app>  deploy war file to tomcat server    Machine generated alternative text: WAR file to deploy Select WAR file to upload Choose File No file chosen Deploy    open app    <http://localhost:8080/sample-web-app-1.0-SNAPSHOT/> |

1. **Glassfish**

|  |  |
| --- | --- |
| docs | <https://javaee.github.io/glassfish/documentation> |
| download | <https://javaee.github.io/glassfish/download> |
| start server | go to    **C:\Users\cosmin.bucur\Documents\glassfish5\bin**    in cmd:    **asadmin start-domain domain1**      open server console    <http://localhost:4848/>      stop server    **asadmin stop-domain domain1** |
| deploy an app | Machine generated alternative text: Common Tasks O Domain server (Admin Server) Clusters Standalone Instances Nodes •1 . fl Applications a Lifecycle Modules • Monitoring Data y Resources Concurrent Resources ( Connectors JDBC JMS Resources JNDI JavaMail Sessions Resource Adapter Configs y ii Configurations b  default-config b 1h server-config Home About.!’ User: admin Domain: domaini Server: localhost GlassFish’ Server Open Source Edition r ——---—--—- r -. r Applications Applications can be enterprise or web applications, or various I enabled on Deployed Applications (O) I Deploy... J Undeploy Enable Disable I Filter: Select Name Deployment Order No items found. |
| launch app | Machine generated alternative text: sample-web-app-i .0-SNAPSHOT [web] Launch sample-web-app-i .0-SNAPSHOT default Servlet sample-web-app-i .0-SNAPSHOT demoServiet Servlet sample-web-app-i .0-SNAPSHOT isp Servlet |
| open app | Machine generated alternative text: Web Application Links lithe server or listener is not running, the link may not work. In this event, check the status of the server instance. After launching the web application, use the browsers Back button to return to this screen. Application Name: sample-web-app-I .0-SNAPSHOT Links: [server] http://WX-5CG8396X50: 8080/sam pie-web-app-i 3662596385449768684.0-SNAPSHOT [server] https://WX-5CG8396X50: 8181/sam pie-web-app-i 3662596385449768684.0-SNAPS HOT |

1. **Wildfly**

|  |  |
| --- | --- |
| docs | <https://docs.wildfly.org/17/> |
| download | <https://wildfly.org/downloads/> |
| start server | <https://www.youtube.com/watch?v=sjHmwRA2Bkk>    go to    **C:\Users\cosmin.bucur\Documents\wildfly-17.0.1.Final\bin**    in cmd:    **standalone.bat**      open server console    <http://127.0.0.1:9990>      stop server    **jboss-cli.bat --connect command=:shutdown**    restart server    **jboss-cli.bat --connect command=:reload** |

1. **Commands**

tomcat

GUI: localhost:8080

start: startup.bat

stop: shutdown.bat

glassfish

GUI: <http://localhost:4848>

start:                asadmin start-domain domain1

stop:                asadmin stop-domain domain1

wildfly

GUI: <http://localhost:9990>

start:         standalone.bat

stop:         jboss-cli.bat --connect command=:shutdown

1. **Recap concepts**

what is a server?

what is the role of a server?

serve static pages

container for servlets / apps

how do servers communicate?

admin console

user

user roles

path

environment variables

(system) path variables

bin

executables

conf

tomcat\_users.xml

server.xml

deploy

a web app

1. **web app structure**

root

src

main

java

com.bucur.project

webapp

META-INF

MANIFEST.MF

WEB-INF

web.xml

index.html

welcome.jsp

pom.xml

README.md

.gitignore