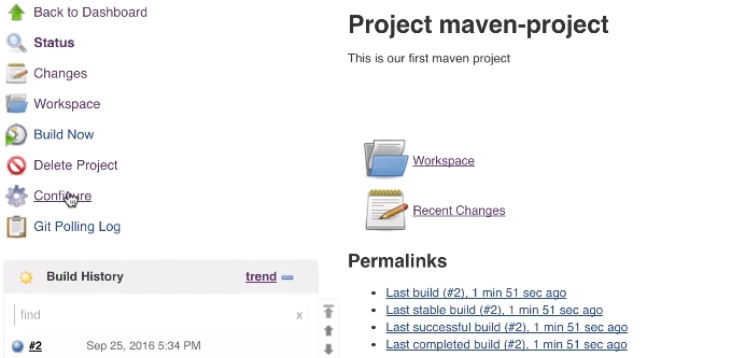
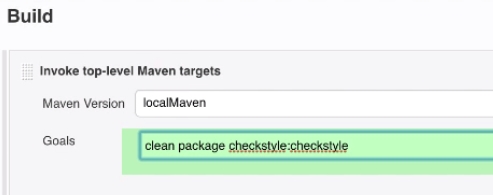
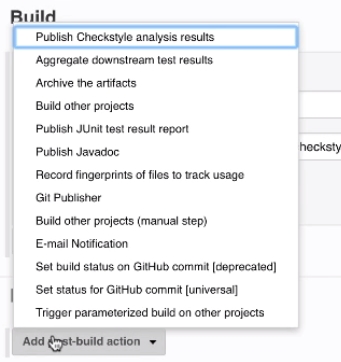


Let us install check style plugin and install it.

Now let us configure our maven project with check style.

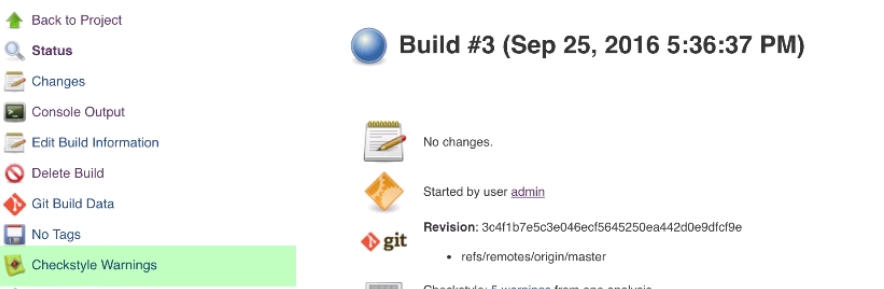






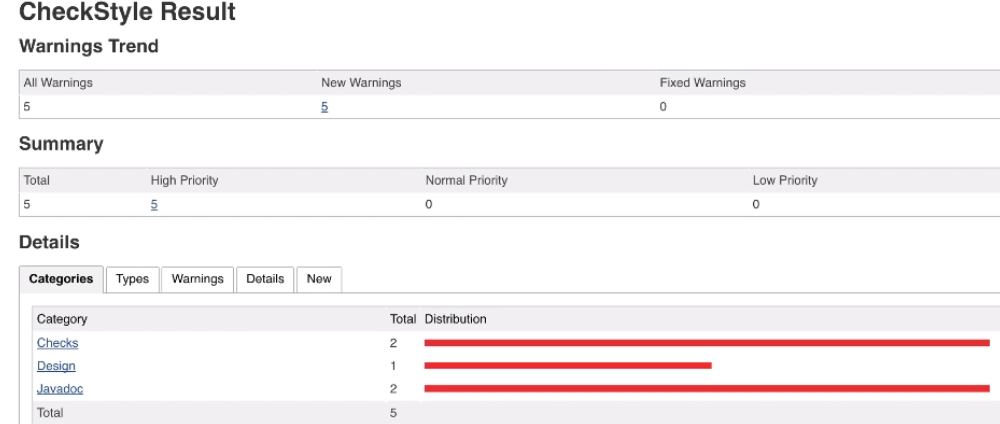
In the post build action section , select publish results. Now save it.

Now let us go for a Build.. and see the output.



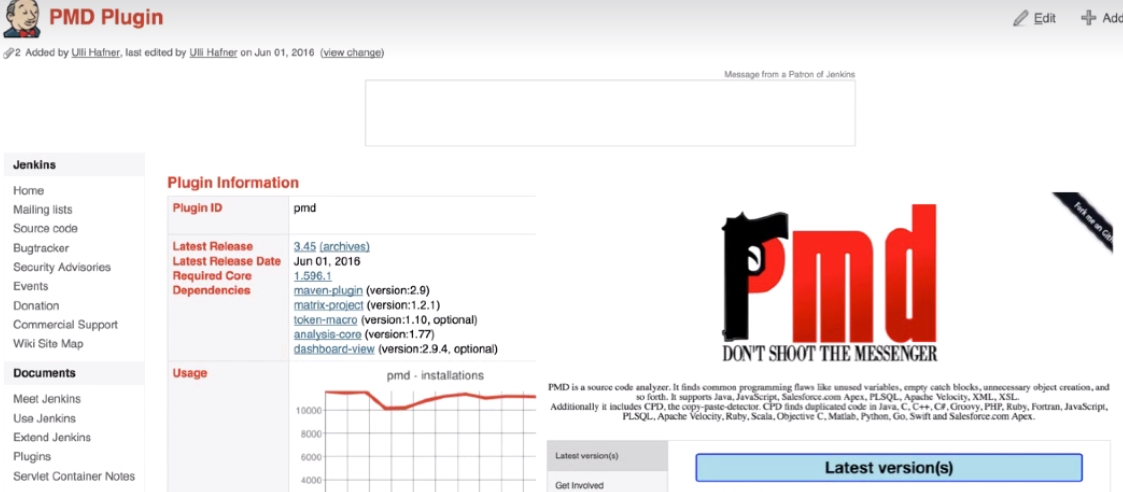
Now we see a new item called Check style warnings.

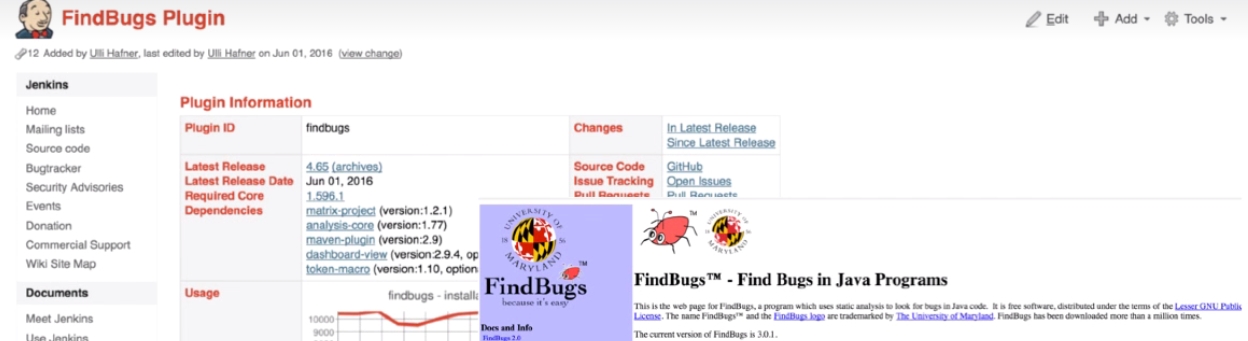
Click on it and we will see check style results. We see warning, summary and details.



Now we can go to each and every section and see the errors, and can edit the code.

Once we commit the code to git, Jenkins start building and we see better results.





PMD Jenkins plugin:

https://wiki.jenkins-ci.org/display/JENKINS/PMD+Plugin

Findbugs Jenkins Plugin:

https://wiki.jenkins-ci.org/display/JENKINS/FindBugs+Plugin

#### Apache Ant:

http://ant.apache.org/

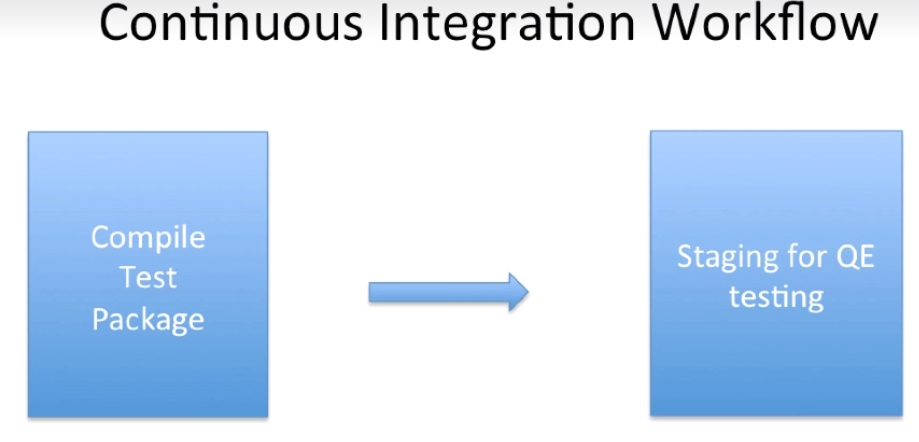
#### Ant Targets:

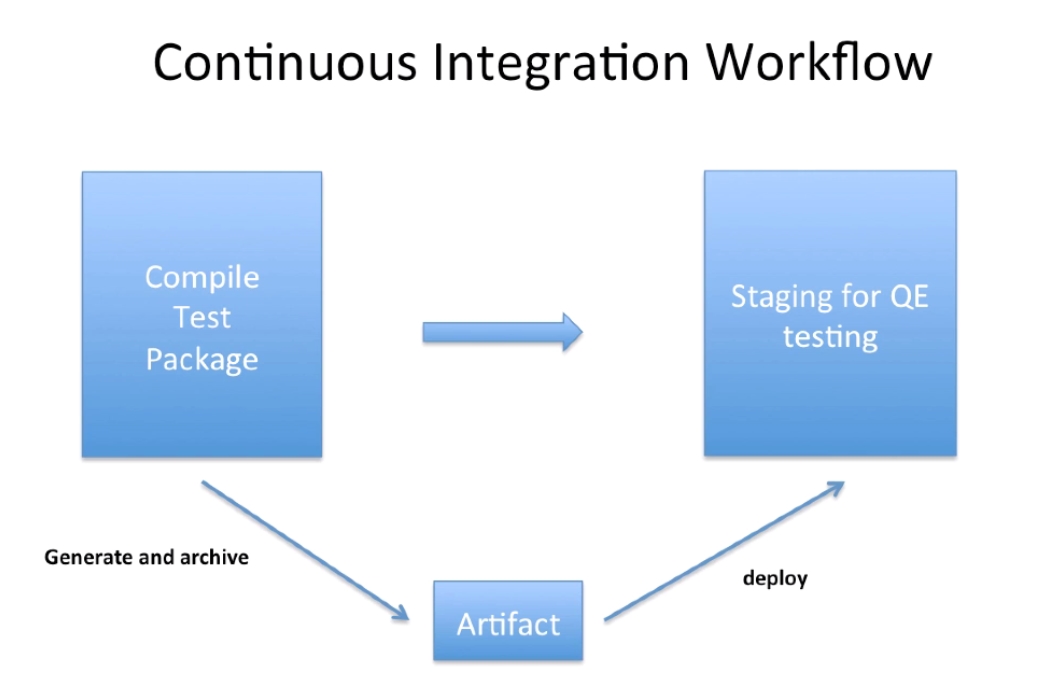
http://ant.apache.org/manual/targets.html

#### Gradle:

https://docs.gradle.org/current/dsl/index.html

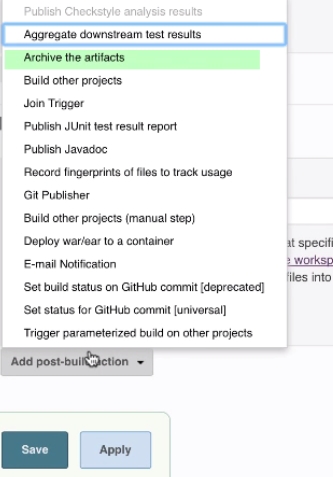
**Deploying The Application:**



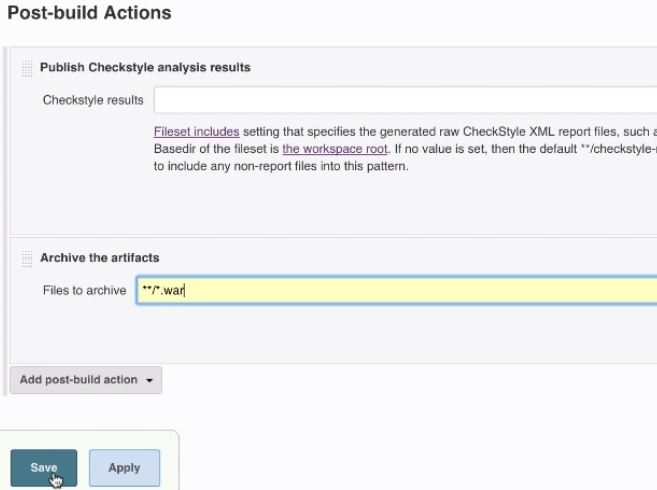


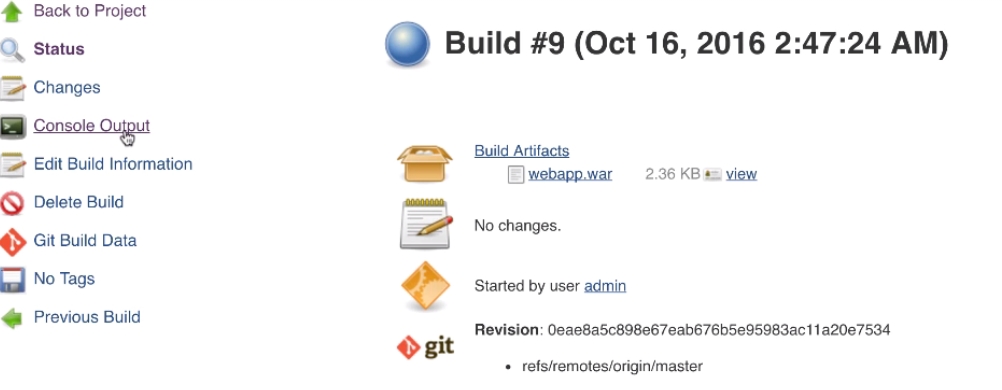
We do compile the package and test and creates an artifact (.war). This can be done by adding a post build step.

The generated war will be deployed to the environments.



Go to the project , go to archive artifacts and then give as below.



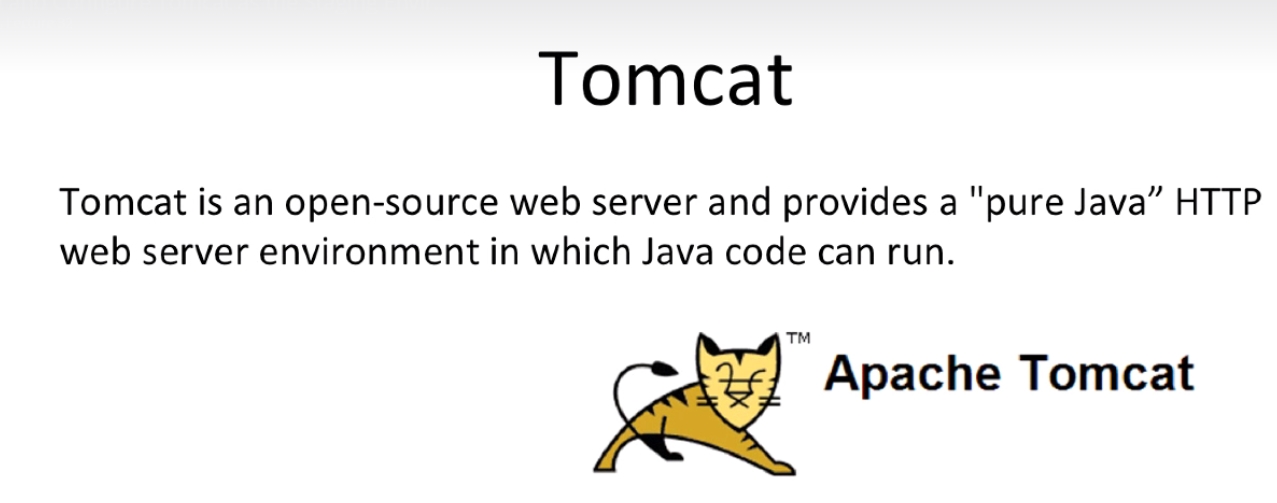
Lets go and build now.

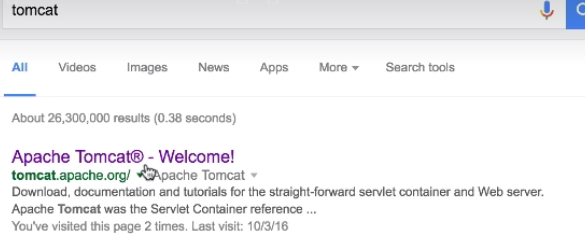
We can see the artifact by clicking on build.

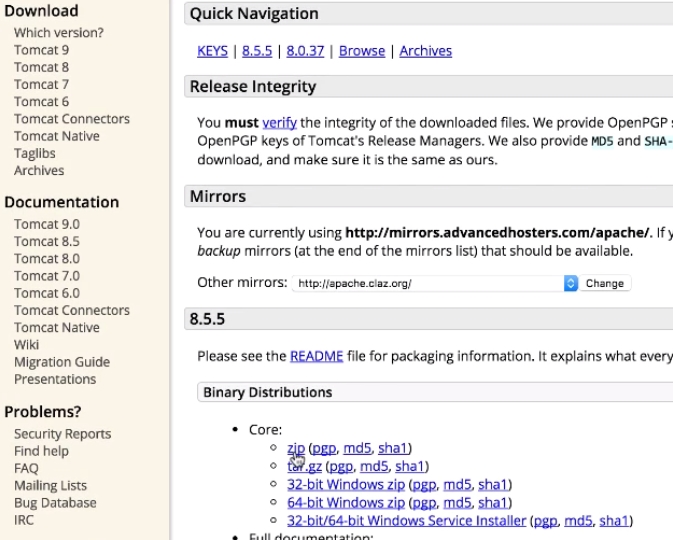


We can see the console out to check the webapp.war and it gets saved in workspace.

**Tomcat Installation and Configuration**







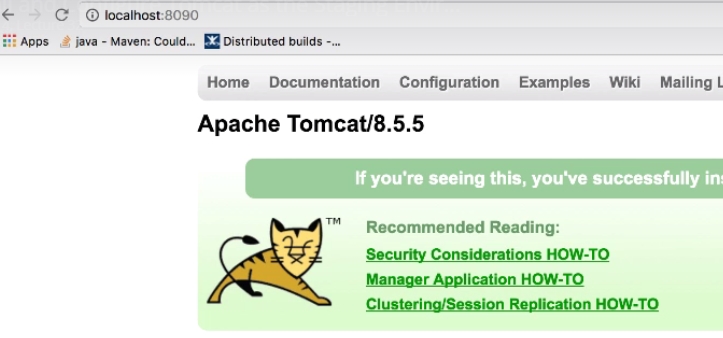
The current stable version is 8.. download the zip. Extract the zip and copy any where. Tomcat is ready. Change the tomcat port because Jenkins also runs on the same port.

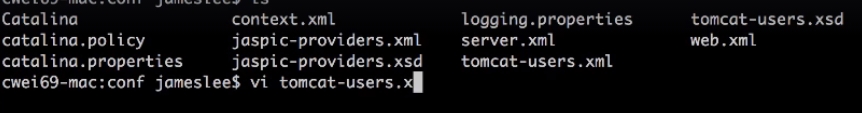


Open Conf/server.xml………. search for connector to change the port.

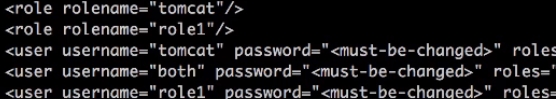


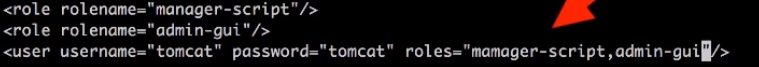
The above are the scripts we use to start the tomcat. Make sure we have complete permissions. Launch the below URL once it starts.





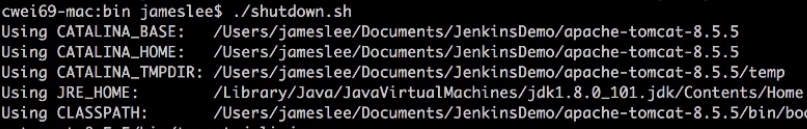
We need to configure user/password for tomcat in the above file.



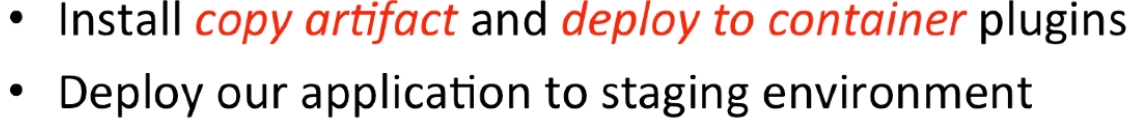


We can create any roles for the same user/pwd

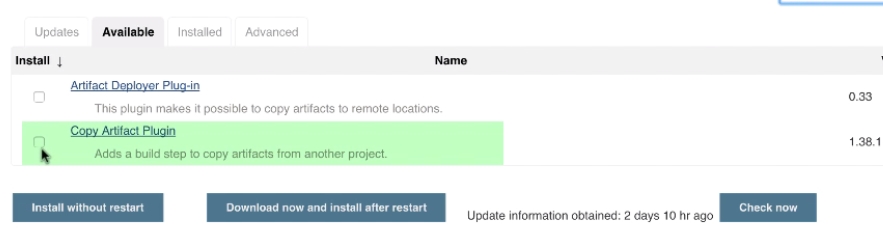
Lets use the tomcat/tomcat user and pwd and assign roles to it. Now restart once.

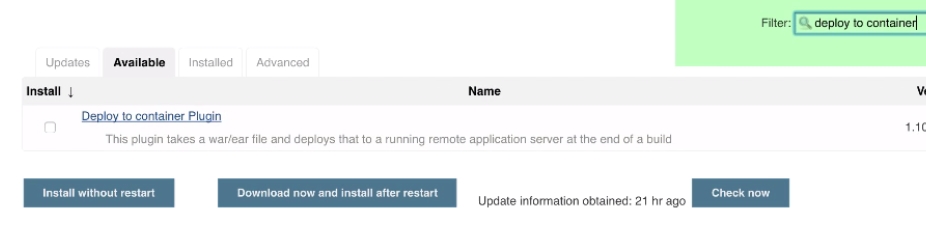


In order to deploy the application to tomcat.. we need to configure two plugins.

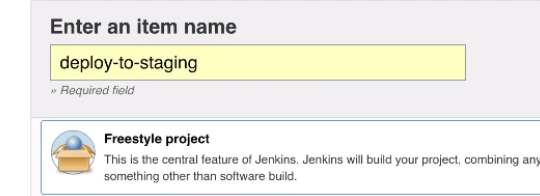


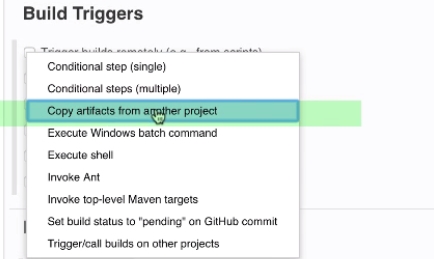
Allows Jenkins to copy artifacts from other projects, and to deploy to tomcat.

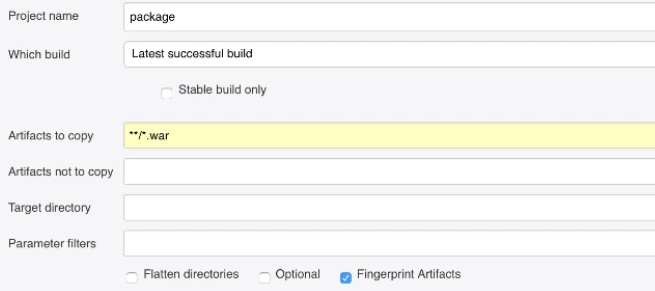




Now go back to new project….

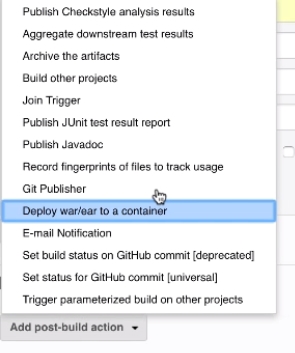


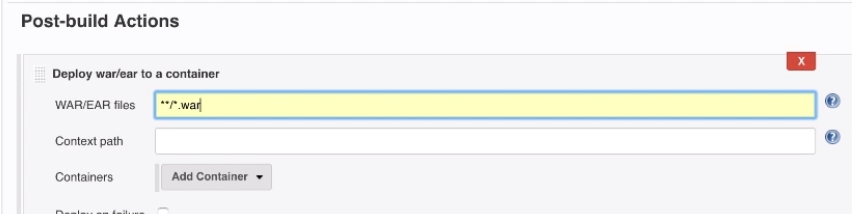




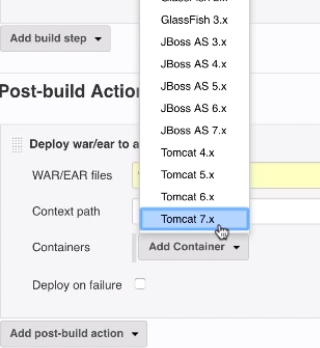
We mentioned the project where we need to copy the artifact and what is the type of the artifact.

Now go to post build action and select below.

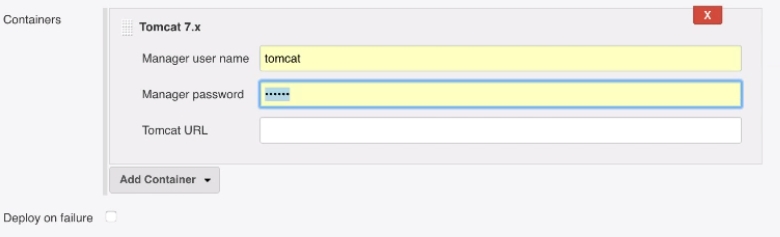




Now select the format of application and Container



Any highest version is fine.

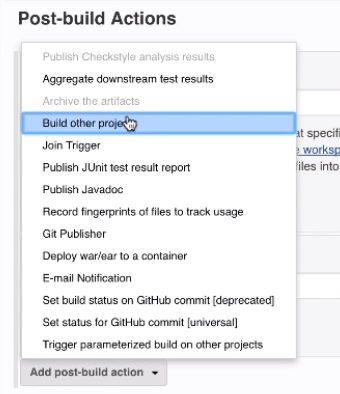


Or we can configure under credentials.

Now the job is ready, this would be automatically triggered once the upstream project is run.



Now select the base project (package job) and select as below.





Now if the package job is success, deploy will also be success.. test it!!