

Jenkins exercises

**Connect to 34.250.44.150 using username:student
Password student**

1. Run Jenkins on docker with volume attached for the jenkins home directory please set a volume for yourself so you won't delete other peoples work
 - a. `docker run -v /home/student/omris:/var/jenkins_home -p 1234:8080 -d jenkins`
2. Password will be in
`/home/student/omris/secrets/initialAdminPassword`
3. Install the suggested plugins
4. Go through the steps to install a default admin user
5. Create a new build that will build the spring-boot project from <https://github.com/omrisiri/spring-boot.git>
6. Setup the build to run the maven pom from `spring-boot-samples/spring-boot-sample-atmosphere`
7. Run the clean compile test and package targets
8. Make sure the artifact exists in the target directory
9. Add an artifact archive step to the post build in jenkins
10. Install cloudbees docker build and publish plugin
11. Install the docker cloud plugin
12. Create the docker image for the build machine from:

<https://raw.githubusercontent.com/omrisiri/devops/master/examples/jenkins/builder/Dockerfile>

Tag it with your own username so you will use it in your own jenkins deployment

13. Configure the docker cloud plugin to use

tcp://34.250.44.150:2375 as the docker endpoint

14. Make the build run on the new docker cloud plugin by setting up a new cloud in the jenkins management interface and adding it to the build configuration
15. Add a build step to build the docker image from the Dockerfile in the spring-boot examples directory and tag it with your name and set the docker registry settings to push to your hub account
16. Make sure you have the docker image built and you can see it in the hub
17. Create a new build that will build the nginx docker image from yesterday and push it to the docker hub
18. Add a notification that will alert on build fail
19. Add a trigger for a build that will run the build when we commit to the git repo