

Title: How to automate Maven and Java JDK8 installation with groovy for Jenkins?

Post Body:

I'm building a Jenkins Docker image and I will like to automate the installation of Maven 3 and Java 8 last JDK. But unfortunately I use these two groovy files locate into the groovy folder:

groovy/java.groovy:

```
import jenkins.model.* import hudson.model.* import hudson.tools.* def inst = Jenkins.getInstance() def desc = inst.getDescr
```

groovy/maven.groovy:

```
import jenkins.*; import jenkins.model.*; import hudson.*; import hudson.model.*; mavenName = 'maven3' mavenVersion = '3.6.0'
```

and then I run the command:

```
docker run -p 8080:8080 -v `pwd`/groovy:/var/jenkins_home/jobs --rm --name jenkinsdocker jenkinsdocker:latest
```

Unfortunately this returns an error:

```
java.io.IOException: jenkins.model.InvalidBuildsDir: ${ITEM_ROOTDIR}/builds does not exist and probably cannot be created
```

I'm just wondering if the groovy files are wrong or if there is something else I missed?

How can I automate the maven/java installation for Jenkins while running a **docker run**? Or is there another way to do it?

Accepted Answer:

I did not find jenkinsdocker image, so I guess is some docker image you generated from jenkins. In any case following [this jenkinsdocker documentation](#) you should put your groovy scripts in the `init.groovy.d` folder to be executed on init. So you need to run docker modifying jobs by `init.groovy.d` and use the `jenkins:latest` image:

```
docker run -p 8080:8080 -v `pwd`/groovy:/var/jenkins_home/init.groovy.d/ --rm --name jenkins jenkins:latest
```

You can also create your own Dockerfile (you can use version 2.60.3 for instance) as in the example above:

```
# Extended from https://github.com/jenkinsci/docker/blob/master/README.md FROM jenkins/jenkins:2.60.3 # Skip setup wizard ENV
```

Build and run the container:

```
docker build jenkinsdocker -t . docker run -p 8080:8080 --name jenkinsdocker jenkinsdocker:latest
```

Highest Rated Answer: None