

CheatSheet: Jenkins & Groovy

LANGUAGES

- PDF Link: [cheatsheet-jenkins-groovy-A4.pdf](#), Category: languages
- Blog URL: <https://cheatsheet.dennyzhang.com/cheatsheet-jenkins-groovy-A4>
- Related posts: Jenkins CheatSheet, #denny-cheatsheets

File me Issues or star this repo.

1.1 Config Jenkins Via Groovy

| Name | Comment |
|--|--|
| Set timezone for jenkins | timezone.groovy |
| Create jenkins views and add jobs to it | jenkins-views.groovy |
| Configure default view | jenkins-views.groovy |
| Create a Jenkins user | create-jenkins-user.groovy |
| logged-in users can do anything | logged-in-users.groovy |
| Add default jobs | Copy jobs config.xml to /usr/share/jenkins/ref/jobs/ |
| Jenkins skip wizzard when initialization | -Djenkins.install.runSetupWizard=false |
| Install jenkins plugins | /usr/local/bin/install-plugins.sh < /usr/share/jenkins/ref/plugins.txt |
| Configure jenkins slave Node | |

1.2 Jenkins Pipeline

| Name | Comment |
|---|--|
| Specify parameter to run jobs | build job:'job1', parameters:[string(name:'name1', value:va1)] |
| Run job in different agents | node(\$agent_label) {...} |
| Ask for user input | stage('stage2'){ input "OK to go?" } |
| Actively fail current pipeline job | error("Build failed because of this and that..") |
| Keep going when previous stage has failed | keep-going-with-errors.groovy |
| Send slack notification in pipeline | slack-notification.groovy |
| Pass parameter across jenkins jobs | jenkinsfile-pass-parameter.groovy |
| Set timeout & retry | jenkinsfile-timeout-retry.groovy |
| Use finally to do cleanup | jenkinsfile-finally.groovy |
| Run jenkins jobs in a sequential way | jenkinsfile-sequentially.groovy |
| Run jenkins jobs in parallel | jenkinsfile-parallelly.groovy |

1.3 Jenkins Basic

| Name | Comment |
|---------------------------|---|
| Deploy Jenkins via docker | https://hub.docker.com/r/jenkins/jenkins/ |

1.4 Groovy Basic

| Name | Comment |
|---------------------------------|--|
| Read file content as a variable | def env = System.getenv(), def content = readFile("/tmp/test.txt") |
| Print stdout | echo 'Action is done', println "Hello World" |
| Use boolean parameter | if (istru == "false") {...} |
| Basic integer caculation | def a = 3, b = 7; println "\$a + \$b = \${a + b}" |
| Get environment variables | get-env.groovy |
| Run groovy online | SaaS: Groovy Web console |
| Run groovy script from Jenkins | Link: Jenkins Script Console |
| Reference | Link: Apache Groovy |

1.5 Groovy String

| Name | Comment |
|--------------------------------|---------------------------------------|
| Convert list to string | <code>1.join(";")</code> |
| Create string with multi-lines | <code>multi-line-string.groovy</code> |
| Convert string to list | <code>split-string.groovy</code> |
| Convert string to json | <code>string-to-json.groovy</code> |

1.6 Groovy Array

| Name | Comment |
|-----------------------------|---|
| Iterate a list | <code>for(item in [1,2,3,4]){ println item }</code> |
| Iterate a list | <code>(1..3).each { println "Number \${it}" }</code> |
| Add item to list | <code>def alist = [10, 9, 8]; alist << 7</code> |
| List size | <code>def alist = [10, 9, 8]; alist.size()</code> |
| Split string with delimiter | <code>'1128-2'.tokenize('-')</code> |

1.7 Groovy Dictionary

| Name | Comment |
|---------------------|---|
| Create a map | <code>def m = ['fruit':'Apple', 'veggie':'Carrot']</code> |
| Add an item to map | <code>m.put('denny','hello')</code> |
| Check if key exists | <code>m.containsKey('key1')</code> |
| Loop a map | <code>loop-map.groovy</code> |

1.8 Groovy Network

| Name | Comment |
|---------------------------------|---|
| Get hostname by ip | <code>get-ip-by-hostname.groovy</code> |
| validate user input: ip address | <code>assert ip_address.matches("\\d{1,3}\\\\.\\d{1,3}\\\\.\\d{1,3}\\\\.\\d{1,3}")</code> |

1.9 Jenkins Groovy

| Name | Command |
|--------------------------------|---|
| Print message to stdout | See <code>print.groovy</code> |
| Get environment variables | See <code>get-env.groovy</code> |
| Convert string to list | See <code>split-string.groovy</code> |
| Create string with multi-lines | See <code>multi-line-string.groovy</code> |
| Loop a map | See <code>loop-map.groovy</code> |
| Get hostname by ip | See <code>get-ip-by-hostname.groovy</code> |
| Convert list to string | <code>1.join(";")</code> , link: <code>groovy list</code> |

1.10 Jenkins Plugins

| Name | Command |
|-----------------------------|-------------------------|
| Backup jenkins | ThinBackup Plugin |
| Load the ssh key | Credentials Plugin |
| Add timestamp to job output | Timestamping Plugin |
| Create dashboard | Dashboard View Plugin |
| Abort if job takes too long | Build-timeout Plugin |
| Retry failed a job | Naginator Plugin |
| Backup job configuration | JobConfigHistory Plugin |

1.11 More Resources

<http://groovy-lang.org/documentation.html#gettingstarted>

<https://jenkins.io/doc/>

License: Code is licensed under MIT License.

2 DONE Better Jenkins UI: ocean-blue

<https://jenkins.io/doc/book/blueocean/getting-started/#as-part-of-jenkins-in-docker>