

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
Groovy manages files/folders	files-folder.groovy
Configure max executors in Jenkins	master-executors.groovy
Add a list of jobs by regexp to a view	addjobstoview-byregexp.groovy
Config kubernetes jenkins	jenkins-kubernetes-cloud.groovy
List all jenkins jobs	<code>println Jenkins.instance.projects.collect { it.name }</code>
Reference	GitHub: cloudbees/jenkins-scripts
Reference	GitHub: jenkinsci/pipeline-examples

1.2 Jenkins Security Via Groovy

Name	Comment
logged-in users can do anything	logged-in-users.groovy
Enable ldap in Jenkins	enable-ldap.groovy
Create a jenkins secret text	create-secret-text.groovy
Configure authorization in Jenkins	matrix-authorization-strategy.groovy
Jenkins skip wizzard when initialization	<code>-Djenkins.install.runSetupWizard=false</code>

1.3 Load Jenkins settings via folder copy

Name	Comment
Add default jobs	<code>Copy jobs/ /usr/share/jenkins/ref/jobs/</code>
Copy custom built plugins	<code>COPY plugins/*.hpi /usr/share/jenkins/ref/plugins/</code>
Use jenkins cli	<code>COPY config/jenkins.properties /usr/share/jenkins/ref/</code>
Add jenkins groovy scripts	<code>COPY config/*.groovy /usr/share/jenkins/ref/init.groovy.d/</code>
Configure Jenkins with some defaults	<code>COPY config/*.xml /usr/share/jenkins/ref/</code>
Install jenkins plugins	<code>/usr/local/bin/install-plugins.sh < /usr/share/jenkins/ref/plugins.txt</code>

1.4 Jenkins networking Via Groovy

Name	Comment
Get hostname	<code>println InetAddress.getLocalHost.canonicalHostName</code>
Get IP address	<code>println InetAddress.getLocalHost.hostAddress</code>

1.5 Jenkins Pipeline

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

1.6 Jenkins Maintenance

Name	Comment
Deploy Jenkins via docker	https://hub.docker.com/r/jenkins/jenkins/
Clean Up Old Builds	Link: CloudBees Best Strategy for Disk Space Management

1.7 Groovy Basic

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

1.8 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>

1.9 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>

1.10 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>

1.11 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>

1.12 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: groovy list

1.13 Jenkins Plugins

Plugin	Summary
Kubernetes Plugin	Jenkins plugin to run dynamic agents in a Kubernetes/Docker environment
Credentials Plugin	Load the ssh key
SiteMonitor Plugin	Monitor URLs
Timestamper Plugin	Add timestamp to job output
Dashboard View Plugin	Create dashboard
Build-timeout Plugin	Abort if job takes too long
Naginator Plugin	Retry failed a job
ThinBackup Plugin	Backup jenkins
JobConfigHistory Plugin	Backup job configuration
Build User Vars Plugin	Describe the user who started the build

1.14 More Resources

<http://groovy-lang.org/documentation.html#gettingstarted>
<https://github.com/jenkinsci/pipeline-model-definition-plugin/wiki/Syntax-Reference>
<https://github.com/fabric8io/jenkins-docker>
<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>