

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
Configure default view	jenkins-views.groovy
Configure Jenkins url	jenkins-url.groovy
Create a Jenkins user	create-jenkins-user.groovy
Groovy manages files/folders	files-folder.groovy
Configure max executors in Jenkins	master-executors.groovy
Config jenkins kubernetes plugin	jenkins-kubernetes-cloud.groovy
Configure slack plugin	config-slack.groovy
Reference	GitHub: cloudbees/jenkins-scripts
Reference	GitHub: jenkinsci/pipeline-examples

1.2 Jenkins Job Via Groovy

Name	Comment
List all jenkins jobs	<code>println Jenkins.instance.projects.collect { it.name }</code>
Add a list of jobs by regexp to a view	addjobstoview-byregexp.groovy
Create jenkins views and add jobs to it	jenkins-views.groovy
Create and trigger a job	create-jenkins-job.groovy
Manage jenkins jobs	manage-jenkins-jobs.groovy
Cancel queued jenkins jobs by regexp	killqueuedjenkins.groovy

1.3 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>

1.4 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>

1.5 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

1.6 Jenkins Git Via Groovy

Name	Comment
Git checkout code	<code>git-checkout.groovy</code>
Get all git commits since last success	<code>git-commits-before-fail.groovy</code>
List git tags and branches	<code>git-list-tags-and-branches.groovy</code>

1.7 Jenkins networking Via Groovy

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

1.8 Jenkins Pipeline

Name	Comment
Specify parameter to run jobs	<code>build job:'job1', parameters:[string(name:'name1', value:va1)]</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>

1.9 Jenkins with Kubernetes/Docker

Name	Comment
Kubernetes Plugin	Jenkins plugin to run dynamic agents in a Kubernetes/Docker environment
Config jenkins kubernetes plugin	<code>jenkins-kubernetes-cloud.groovy</code>
Cleanup for Docker stale containers/images/volumes	<code>docker-cleanup.groovy</code>

1.10 Groovy Basic

Name	Comment
Get environment variables	get-env.groovy
Groovy execute command	execute-command.groovy
Print stdout	print.groovy echo 'Action is done', println "Hello World"
Use boolean parameter	if (istruer == "false") {...}
Basic integer caculation	def a = 3, b = 7; println "\$a + \$b = \${a + b}"
Run groovy online	SaaS: Groovy Web console
Run groovy script from Jenkins	Link: Jenkins Script Console
Reference	Link: Apache Groovy

1.11 Groovy String

Name	Comment
Convert list to string	1.join(";")
Create string with multi-lines	multi-line-string.groovy
Convert string to list	split-string.groovy
Convert string to json	string-to-json.groovy
Regex match	regex-match.groovy

1.12 Groovy Array

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

1.13 Groovy File

Name	Comment
Read file content as a variable	def env = System.getenv(), def content = readFile("/tmp/test.txt")
Read and write json files	json-file.groovy
Obtain a relative path	relative-path.groovy

1.14 Groovy Dictionary

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

1.15 Groovy json

Name	Comment
Convert string to json	string-to-json.groovy
Convert dictionary to json	dict-to-json.groovy
Read and write json files	json-file.groovy

1.16 Groovy Network

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

1.17 Groovy Date

Name	Comment
Date to string	<code>new Date().format("yyyy-MM-dd'T'HH:mm:ss'Z'", TimeZone.getTimeZone("UTC"))</code>
String to date	<code>Date.parse("yyyy-MM-dd'T'HH:mm:ss'Z'", "2001-01-01T00:00:00Z")</code>
String to date	<code>Date.parse("yyyy-MM-dd'T'HH:mm:ss'Z'", "2001-01-01T00:00:00+0000")</code>

1.18 Jenkins Agent

Name	Comment
Check jenkins slave jar version	<code>check-slave-jar-version.groovy</code>
Find dead executors and remove them	<code>find-dead-executors.groovy</code>

1.19 Jenkins Maintenance

Name	Comment
Delete jenkins job by regexp	<code>delete-job-by-regexp.groovy</code>
Deploy Jenkins via docker	https://hub.docker.com/r/jenkins/jenkins/
Clean up old builds	Link: CloudBees Best Strategy for Disk Space Management

1.20 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://gist.github.com/vdupain/832964527b4b8d7d4c648169dae8c656>

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

License: Code is licensed under MIT License.