

EPAM Systems, RD Dep., RD Dep.

MTN.*NIX.09

Continuous Delivery with Jenkins

MTN.*NIX.09 Continious Delivery with Jenkins

Software build principles

REVISION HISTORY					
Ver.	Description of Change	Author	Date	Approved	
				Name	Effective Date
<1.0>	Initial revision	Aleksandr Mazurenko	<25-Jun-2017>		

Task 1. Gradle build

Review

Implement Gradle build script for sample project.

Task

1. Install Gradle

Not secure | 35.225.200.81:8080/updateCenter/

kins

search

Update Center

Dashboard

Jenkins

Plugins

Installing Plugins/Upgrades

Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

Gradle Success

Loading plugin extensions Success

[Go back to the top page](#)
(you can start using the installed plugins right away)

☐ Restart Jenkins when installation is complete and no jobs are running

Jenkins > Global Tool Configuration

Menu

Gradle

Gradle installations

Add Gradle

Gradle

name

☒ Install automatically

Install from Gradle.org

Version

Delete Installer

Add Installer

Delete Gradle

Add Gradle

List of Gradle installations on this system

2. Clone GitHub project

[git@github.com:MNT-Lab/build-tools.git](https://github.com/MNT-Lab/build-tools.git)

3. Create your own branch based on master by pattern:

First char of name + surname

Example: The name of student - Serhei Ivanou. The branch will be **sivanou**.

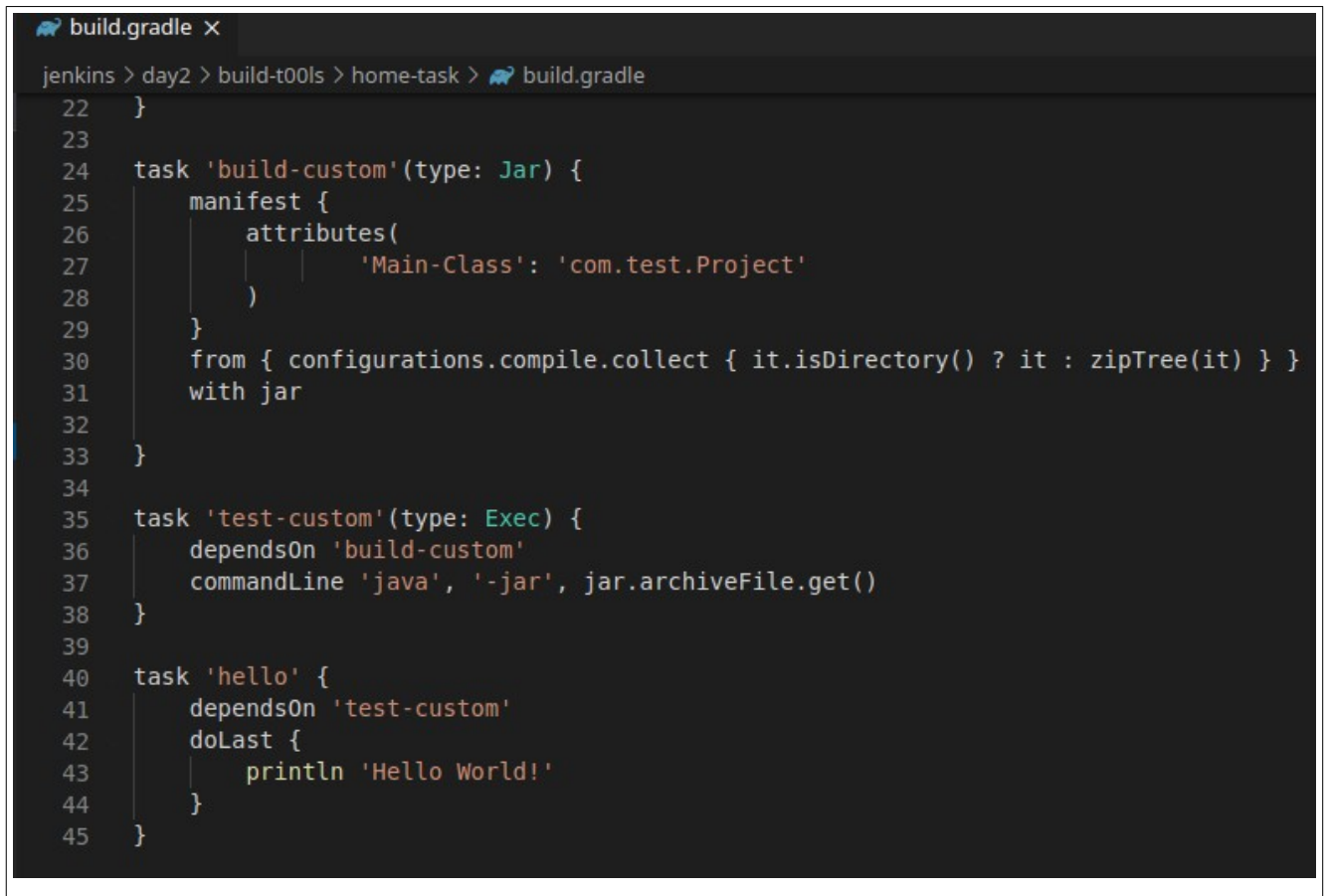
The screenshot shows the 'Source Code Management' tab for a Jenkins job named 'kkarpau_gradle'. The 'Git' option is selected under 'None' or 'Git'. In the 'Repositories' section, the 'Repository URL' is set to 'https://github.com/MNT-Lab/build-tools.git' and 'Credentials' is set to '- none -'. There are 'Advanced...' and 'Add Repository' buttons. In the 'Branches to build' section, the 'Branch Specifier (blank for \'any\')' is set to '*/kkarpau', with an 'Add Branch' button.

4. Implement **build.gradle** build to clean, compile, test, create jar file and distribution artifact. (source code in 'src' repo).

The screenshot shows the 'Build' tab for the 'kkarpau_gradle' Jenkins job. Under 'Invoke Gradle script', the 'Invoke Gradle' option is selected, 'Gradle Version' is set to 'kkarpau_gradle', and 'Tasks' is set to 'clean build'. There is an 'Advanced...' button. Below this, the 'Execute shell' section contains a command: `java -jar home-task/build/libs/kkarpau-gradle.jar > output.log` and `tar -cvf hello-${BUILD_NUMBER}.tar home-task/build/libs/kkarpau-gradle.jar`. A link to 'the list of available environment variables' is provided at the bottom.

5. Build file should contain:

- Tasks: “build-custom” , “test-custom” and “hello”.
- “hello” should depend on “test-custom”, “test-custom” should depend on “build”
- setting custom artifact name

A screenshot of a Jenkins console window. The title bar shows a Jenkins icon and the text 'build.gradle x'. The breadcrumb navigation path is 'jenkins > day2 > build-tools > home-task > build.gradle'. The main area displays the content of the build.gradle file with line numbers 22 through 45. The code defines three tasks: 'build-custom' (type: Jar), 'test-custom' (type: Exec), and 'hello' (type: default). 'build-custom' uses the 'manifest' task to set the 'Main-Class' attribute to 'com.test.Project' and uses 'from' and 'with jar' to create a jar file. 'test-custom' depends on 'build-custom' and runs 'java -jar' on the generated jar file. 'hello' depends on 'test-custom' and prints 'Hello World!' at the end of the build.

```
22 }
23
24 task 'build-custom'(type: Jar) {
25     manifest {
26         attributes(
27             'Main-Class': 'com.test.Project'
28         )
29     }
30     from { configurations.compile.collect { it.isDirectory() ? it : zipTree(it) } }
31     with jar
32 }
33
34
35 task 'test-custom'(type: Exec) {
36     dependsOn 'build-custom'
37     commandLine 'java', '-jar', jar.archiveFile.get()
38 }
39
40 task 'hello' {
41     dependsOn 'test-custom'
42     doLast {
43         println 'Hello World!'
44     }
45 }
```

6. All changes should be tracked by commits to own branch

7. Push to GitHub

8. Setup Jenkins job(“helloworld-**sivanou**-gradle-build”)::

- a) build from repository (poll SCM)
- b) use Gradle plugin
- c) run jar and output save to file ‘output.log’
- d) archive and attach jar file
ARTEFACT_NAME=hello-\$BUILD_NUMBER.tar.gz
- e) attach log file to build

karpau_gradle

General
Source Code Management
Build Triggers
Build Environment
Build
Post-build Actions

Advanced...

Add build step

Post-build Actions

Archive the artifacts

Files to archive
home-task/build/libs/*.jar, *.log, *.tar

Advanced...

Add post-build action

Save
Apply

karpau_gradle
#5

Project

Output

Information

Build '#5'

Data

Build #5 (Oct 19, 2020)

Build Artifacts

hello-4.tar	3.00 KB	view
hello-5.tar	3.00 KB	view
karpau-gradle.jar	1.13 KB	view
output.log	16 B	view

No changes.

Started by user [devops](#)

<
>
↻

Not secure
35.225.200.81:8080/job/karpau_gradle/5/artifact/output.log

Hello, MNT Lab!

© EPAM Systems, 2017

Page: 5/6

Additional Practice: install gradle wrapper and try to manipulate by tasks/commands

Report:

Email pattern: [MNT-CD-4]-FirstName-LastName

Contains: GitHub link, Plugin config, Job config, output (screenshot)

GitHub Repository:

<https://github.com/MNT-Lab/build-tools/tree/kkarpau>