

Java Project

Code, Clean, Package & Deploy

Using Jenkins

By

Keshav Kummari

Step-1 : Go to Jenkins & Create a Maven Project

☒ Discard old builds

Strategy Log Rotation

Days to keep builds

if not empty, build records are only kept up to this number of days

Max # of builds to keep 5

if not empty, only up to this number of build records are kept

Days to keep artifacts

if not empty, artifacts from builds older than this number of days will be deleted, but the logs, history, reports, etc for the build will be kept

Max # of builds to keep with artifacts 3

if not empty, only up to this number of builds have their artifacts retained

Step-2: Let's select Parameterized

☒ This project is parameterized

Git Parameter

Name **branch**

Description

Parameter Type Branch or Tag

Branch

Branch Filter

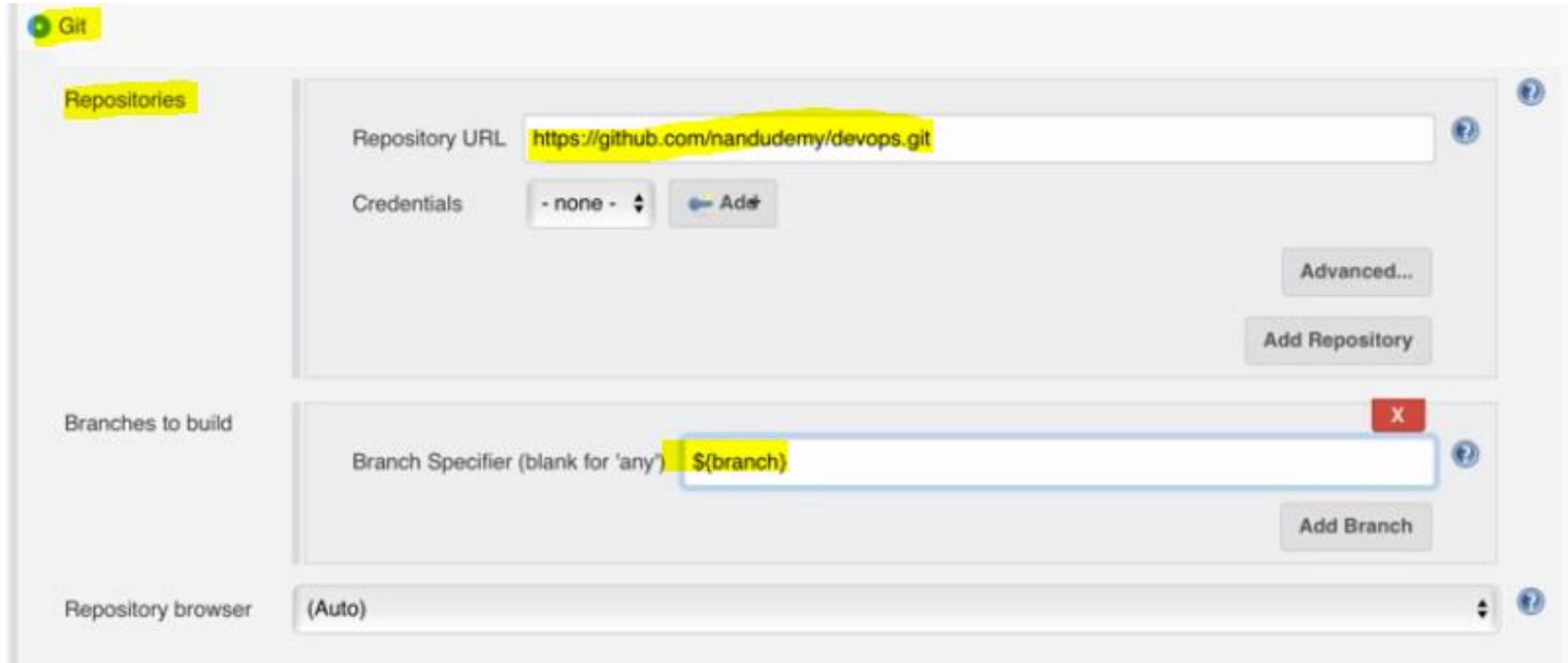
Tag Filter

Sort Mode NONE

Default Value **develop**

Selected Value NONE

Step-3: Select “Git” & Do parameterized branch



The screenshot shows the Jenkins configuration interface for a Git repository. The left sidebar has a 'Git' icon and a 'Repositories' label. The main area is divided into three sections:

- Repositories:** Contains a 'Repository URL' field with the value `https://github.com/nandudemy/devops.git`, a 'Credentials' dropdown set to '- none -' with an 'Add' button, an 'Advanced...' button, and an 'Add Repository' button.
- Branches to build:** Contains a 'Branch Specifier (blank for 'any')' field with the value `$(branch)`, a red 'X' icon, and an 'Add Branch' button.
- Repository browser:** A dropdown menu currently set to '(Auto)'.


Step-4: Add 2 options “Clean before checkout” & “Check out to specific branch”


The screenshot shows a configuration window with the following elements:


- Branch Specifier (blank for 'any'):** A text input field containing `${branch}` and a blue help icon.
- Add Branch:** A button located to the right of the Branch Specifier field.
- Repository browser:** A dropdown menu currently set to `(Auto)` with a blue help icon.
- Additional Behaviours:** A section containing a list of behaviors:
 - Clean before checkout:** A behavior name highlighted in yellow, with a red 'X' icon and a blue help icon to its right.
 - Check out to specific local branch:** A behavior name highlighted in yellow, with a red 'X' icon and a blue help icon to its right.
- Branch name:** A text input field below the second behavior, with a yellow highlight on the label and a blue border.
- Add:** A button with a dropdown arrow at the bottom left of the Additional Behaviours section.


Step-5: Add Build Triggers


Build Triggers


☒ Build whenever a SNAPSHOT dependency is built 


☐ Schedule build when some upstream has no successful builds 

☐ Trigger builds remotely (e.g., from scripts) 

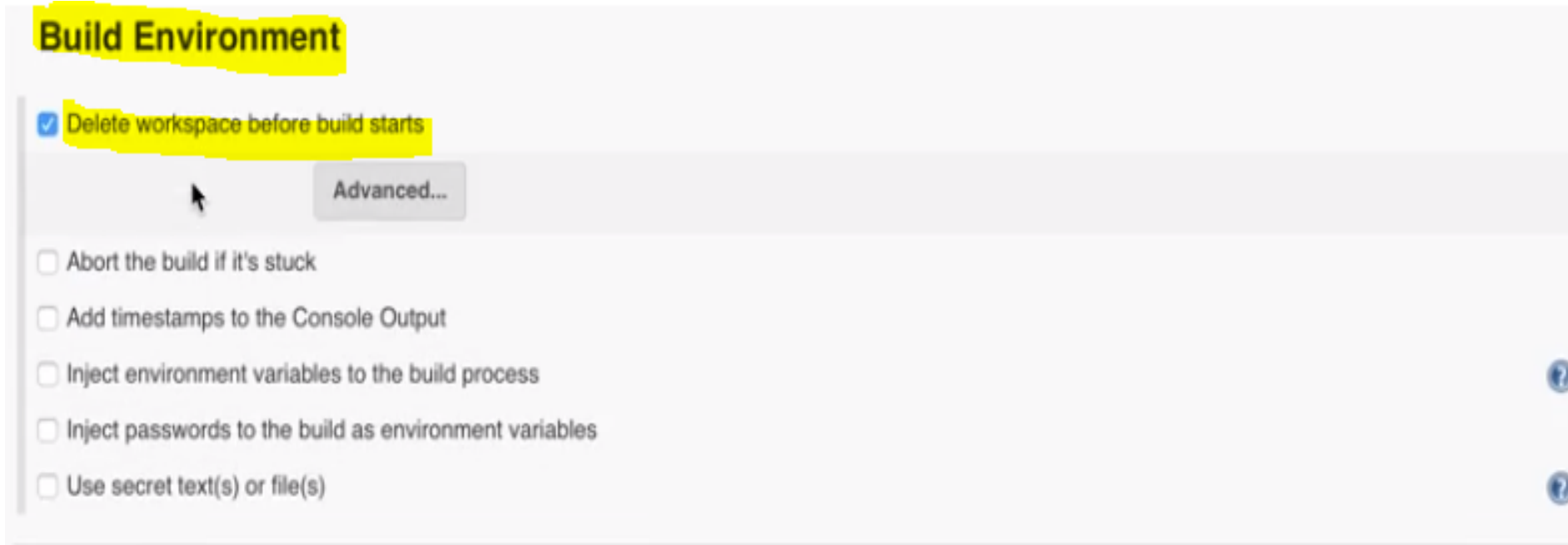
☐ Build after other projects are built 

☐ Build periodically 

☐ GitHub hook trigger for GITScm polling 

☐ Poll SCM 

Step-6: Click on “Delete workspace before build starts” check box part of Build Environment



Step-7: Complete Pre Steps

Pre Steps

Add pre-build step ▾

Build

Root POM

?

Goals and options

?

Advanced...

Step-8: Add “Choice Parameter” for a tag

Parameter Type Branch or Tag

Advanced...

Choice Parameter X

Name tag

Choices no
yes

Description Option to build a snapshot or release.

[Plain text] [Preview](#)

Step-9: Deploy build to Snapshot based on tag is no

Post Steps

☐ Run only if build succeeds ☐ Run only if build succeeds or is unstable ☒ Run regardless of build result

Should the post-build steps run only for successful builds, etc.

Conditional steps (multiple) ✕

Run? **Strings match** ?

String 1 **\$(tag)**

String 2 **no**

Case insensitive ☐ ?

Step-10: Add “Execute Shell” & “Invoke top-level Maven targets”

Steps to run if condition is met

Execute shell

Command `echo "Building snapshot from branch ${branch}"`

[See the list of available environment variables](#)

Advanced...

Invoke top-level Maven targets

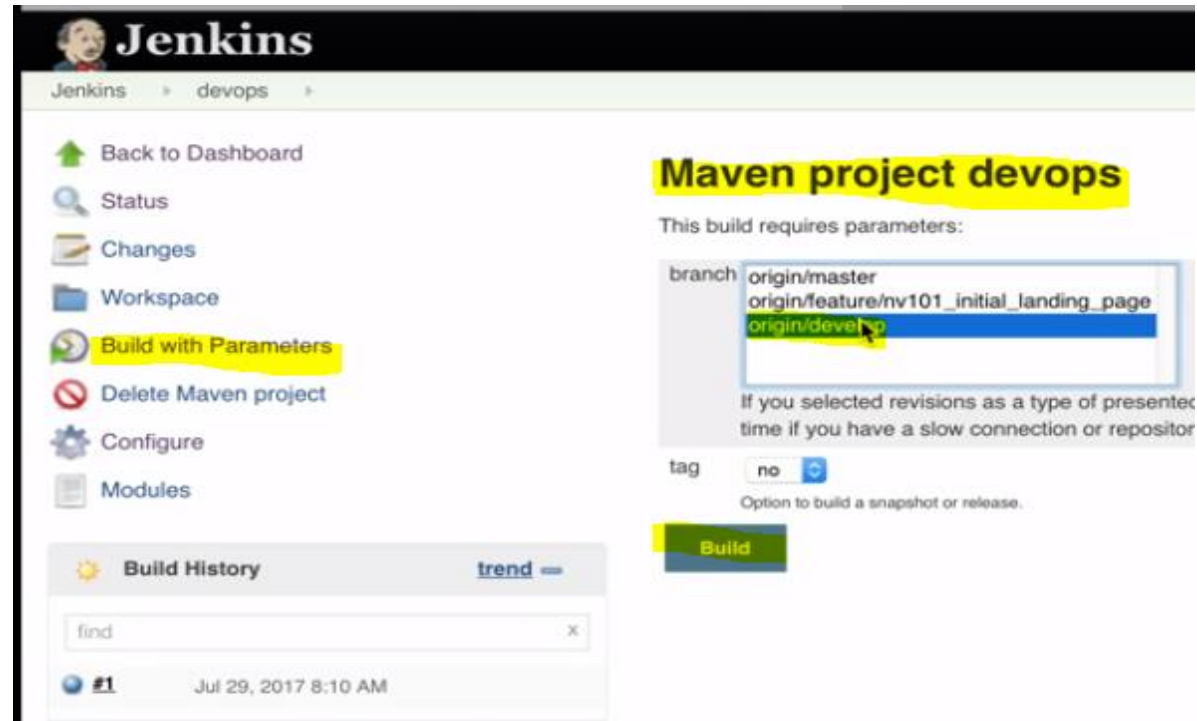
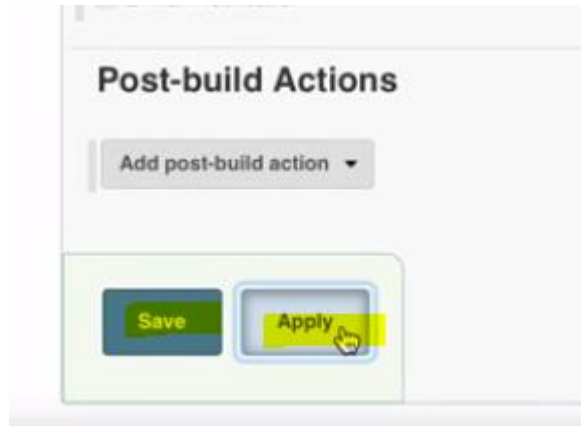
Maven Version `maven`

Goals `deploy`

Advanced...

Add step to condition ▼

Step-11: Click on Apply & Build based on develop branch



Step-12: It's Building and Deploying the War to Jfrog Artifactory under Snapshot path

```
Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml (767 B at 51 kB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml
Progress (1): 767 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml (767 B at 37 kB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml
Progress (1): 313 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml (313 B at 13 kB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml
Progress (1): 313 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml (313 B at 16 kB/s)
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 13.870 s
[INFO] Finished at: 2017-07-29T08:18:36-05:00
[INFO] Final Memory: 33M/379M
[INFO] -----
Finished: SUCCESS
```

Step-13: Verify the Build Part of Jfrog Artifactory

JFrog Artifactory

Artifact Repository Browser

Tree Simple

libs-release

libs-snapshot

libs-release-local

libs-snapshot-local

com/example/devops

1.0.0-SNAPSHOT

devops-1.0.0-20170729.034735-1.pom

devops-1.0.0-20170729.034735-1.war

devops-1.0.0-20170729.044953-2.pom

devops-1.0.0-20170729.044953-2.war

devops-1.0.0-20170729.131835-3.pom

devops-1.0.0-20170729.131835-3.war

maven-metadata.xml

maven-metadata.xml

libs-release

General

Info

Name: libs-release

Package Type: Maven

Repository Path: libs-release/

Repository Layout: maven-2-default

Included Repositories

libs-release-local jcenter

Index of **libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT**

Name	Last modified	Size
./		
./		
devops-1.0.0-20170729.034735-1.pom	28-Jul-2017 22:47	2.70 KB
devops-1.0.0-20170729.034735-1.pom.md5	28-Jul-2017 22:47	32 bytes
devops-1.0.0-20170729.034735-1.pom.sha1	28-Jul-2017 22:47	40 bytes
devops-1.0.0-20170729.034735-1.war	28-Jul-2017 22:47	20.28 MB
devops-1.0.0-20170729.034735-1.war.md5	28-Jul-2017 22:47	32 bytes
devops-1.0.0-20170729.034735-1.war.sha1	28-Jul-2017 22:47	40 bytes
devops-1.0.0-20170729.044953-2.pom	28-Jul-2017 23:49	2.70 KB
devops-1.0.0-20170729.044953-2.pom.md5	28-Jul-2017 23:49	32 bytes
devops-1.0.0-20170729.044953-2.pom.sha1	28-Jul-2017 23:49	40 bytes
devops-1.0.0-20170729.044953-2.war	28-Jul-2017 23:49	20.28 MB
devops-1.0.0-20170729.044953-2.war.md5	28-Jul-2017 23:49	32 bytes
devops-1.0.0-20170729.044953-2.war.sha1	28-Jul-2017 23:49	40 bytes
devops-1.0.0-20170729.131835-3.pom	29-Jul-2017 08:18	2.70 KB
devops-1.0.0-20170729.131835-3.pom.md5	29-Jul-2017 08:18	32 bytes
devops-1.0.0-20170729.131835-3.pom.sha1	29-Jul-2017 08:18	40 bytes
devops-1.0.0-20170729.131835-3.war	29-Jul-2017 08:18	20.28 MB
devops-1.0.0-20170729.131835-3.war.md5	29-Jul-2017 08:18	32 bytes
devops-1.0.0-20170729.131835-3.war.sha1	29-Jul-2017 08:18	40 bytes
maven-metadata.xml	29-Jul-2017 08:18	767 bytes
maven-metadata.xml.md5	29-Jul-2017 08:18	32 bytes
maven-metadata.xml.sha1	29-Jul-2017 08:18	40 bytes

Artifactory/5.4.5 Server at localhost Port 8081

Deploy Java Build on Tomcat Server Using Jenkins

- Go to Post Build Actions >> Select >> Deploy war/ear to a container
- Tomcat Manager UserName : admin & Password : redhat@123

The screenshot shows the Jenkins 'Post-build Actions' configuration page. The 'Deploy war/ear to a container' action is selected. The 'WAR/EAR files' field is set to '**/*.war'. The 'Context path' is set to 'devops'. Under the 'Containers' section, a container named 'Tomcat 7.x' is configured with 'Manager user name' as 'manager', 'Manager password' as '*****', and 'Tomcat URL' as 'http://localhost:8080'. There is an 'Add Container' button and a 'Deploy on failure' checkbox at the bottom.

Post-build Actions

Deploy war/ear to a container

WAR/EAR files: ****/*.war**

War/ear files to deploy. Relative to the workspace root. You can also specify Ant-style GLOBs, like ***/*.war** (from [Deploy to container Plugin](#))

Context path: **devops**

Containers

Tomcat 7.x

Manager user name: **manager**

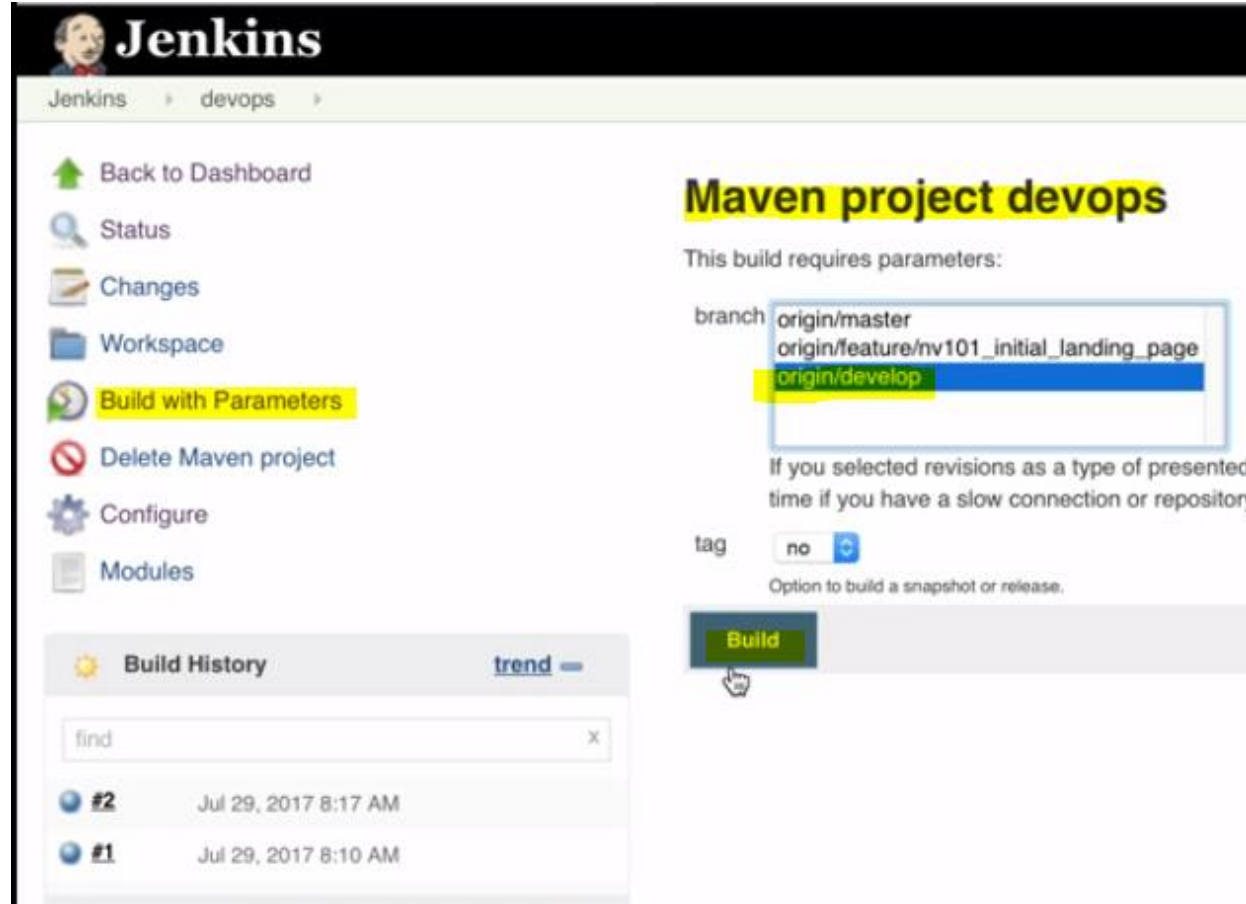
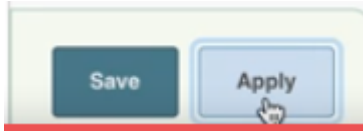
Manager password: *********

Tomcat URL: **http://localhost:8080**

Add Container ▾

Deploy on failure ☐

Step-2: Click on Apply & Build based on develop branch



Jenkins

Jenkins » devops »

- Back to Dashboard
- Status
- Changes
- Workspace
- Build with Parameters**
- Delete Maven project
- Configure
- Modules

Build History trend

find x

#2	Jul 29, 2017 8:17 AM
#1	Jul 29, 2017 8:10 AM

Maven project devops

This build requires parameters:

branch

- origin/master
- origin/feature/nv101_initial_landing_page
- origin/develop**

If you selected revisions as a type of presented time if you have a slow connection or repository

tag no

Option to build a snapshot or release.

Build

Step-3: It's Building and Deploying the War to Jfrog Artifactory under Snapshot path

```
Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml (767 B at 51
kB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml
Progress (1): 767 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml (767 B at 37
kB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml
Progress (1): 313 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml (313 B at 13 kB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml
Progress (1): 313 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml (313 B at 16 kB/s)
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 13.870 s
[INFO] Finished at: 2017-07-29T08:18:36-05:00
[INFO] Final Memory: 33M/379M
[INFO] -----
Finished: SUCCESS
```

Step-4: Verify the Build Part of Jfrog Artifactory

JFrog Artifactory

Tree Simple

libs-release

libs-snapshot

libs-release-local

libs-snapshot-local

com/example/devops

1.0.0-SNAPSHOT

devops-1.0.0-20170729.034735-1.pom

devops-1.0.0-20170729.034735-1.war

devops-1.0.0-20170729.044953-2.pom

devops-1.0.0-20170729.044953-2.war

devops-1.0.0-20170729.131835-3.pom

devops-1.0.0-20170729.131835-3.war

maven-metadata.xml

maven-metadata.xml

libs-release

General

Info

Name: libs-release

Package Type: Maven

Repository Path: libs-release/

Repository Layout: maven-2-default

Included Repositories

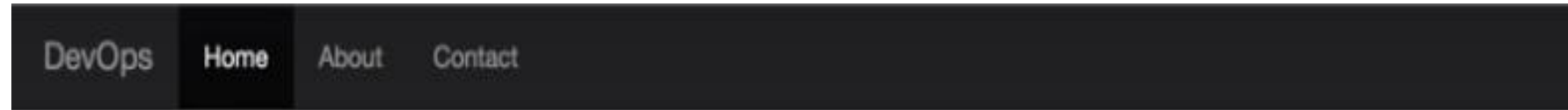
libs-release-local jcenter

Index of **libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT**

Name	Last modified	Size
./		
./		
devops-1.0.0-20170729.034735-1.pom	28-Jul-2017 22:47	2.70 KB
devops-1.0.0-20170729.034735-1.pom.md5	28-Jul-2017 22:47	32 bytes
devops-1.0.0-20170729.034735-1.pom.sha1	28-Jul-2017 22:47	40 bytes
devops-1.0.0-20170729.034735-1.war	28-Jul-2017 22:47	20.28 MB
devops-1.0.0-20170729.034735-1.war.md5	28-Jul-2017 22:47	32 bytes
devops-1.0.0-20170729.034735-1.war.sha1	28-Jul-2017 22:47	40 bytes
devops-1.0.0-20170729.044953-2.pom	28-Jul-2017 23:49	2.70 KB
devops-1.0.0-20170729.044953-2.pom.md5	28-Jul-2017 23:49	32 bytes
devops-1.0.0-20170729.044953-2.pom.sha1	28-Jul-2017 23:49	40 bytes
devops-1.0.0-20170729.044953-2.war	28-Jul-2017 23:49	20.28 MB
devops-1.0.0-20170729.044953-2.war.md5	28-Jul-2017 23:49	32 bytes
devops-1.0.0-20170729.044953-2.war.sha1	28-Jul-2017 23:49	40 bytes
devops-1.0.0-20170729.131835-3.pom	29-Jul-2017 08:18	2.70 KB
devops-1.0.0-20170729.131835-3.pom.md5	29-Jul-2017 08:18	32 bytes
devops-1.0.0-20170729.131835-3.pom.sha1	29-Jul-2017 08:18	40 bytes
devops-1.0.0-20170729.131835-3.war	29-Jul-2017 08:18	20.28 MB
devops-1.0.0-20170729.131835-3.war.md5	29-Jul-2017 08:18	32 bytes
devops-1.0.0-20170729.131835-3.war.sha1	29-Jul-2017 08:18	40 bytes
maven-metadata.xml	29-Jul-2017 08:18	767 bytes
maven-metadata.xml.md5	29-Jul-2017 08:18	32 bytes
maven-metadata.xml.sha1	29-Jul-2017 08:18	40 bytes

Artifactory/5.4.5 Server at localhost Port 8081

Step-5: Build has been deployed to Tomcat through Jenkins

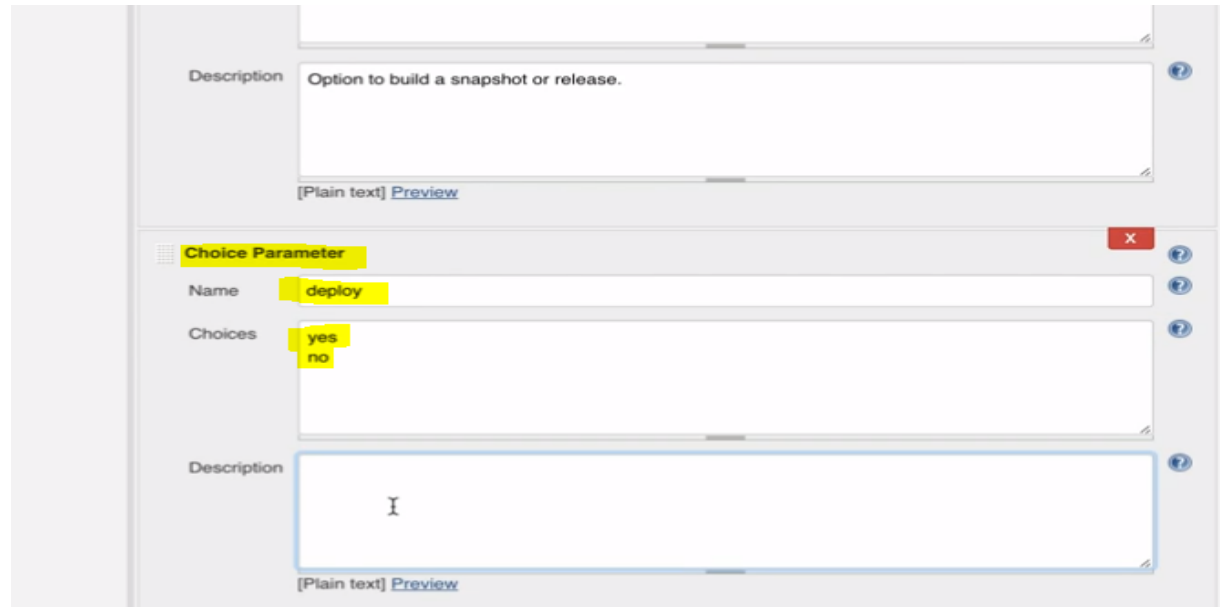


Welcome to the DevOps Course

Course Name:DevOps

Deploy Java Build on Tomcat Server Using Jenkins based on Tagging

- Step-1 : Go to >> Parameterized >> Add “Choice parameter”



The screenshot shows the Jenkins configuration interface for adding a choice parameter. The top section has a 'Description' field with the text 'Option to build a snapshot or release.' and a '[Plain text] Preview' link. Below this is a 'Choice Parameter' section with a red 'X' icon and a help icon. It contains a 'Name' field with the value 'deploy', a 'Choices' field with the values 'yes' and 'no', and a 'Description' field with a cursor. The bottom of the 'Choice Parameter' section also has a '[Plain text] Preview' link.

Step-2: Go to >> Build Environment >> Inject passwords to the build as environment variables

- Tomcat Manager User Name : admin & Password : redhat@123

The screenshot shows the Jenkins configuration page for 'Inject passwords to the build as environment variables'. The checkbox is checked. Under 'Global passwords', there is an unchecked checkbox. Under 'Job passwords', there are two password entries. The first entry has 'Name' set to 'tcuser' and 'Password' set to '*****'. The second entry has 'Name' set to 'tcpass' and 'Password' set to '*****'. There is an 'Add' button at the bottom left and a 'Passwords list' label at the bottom. A red 'X' button is visible next to each password entry.

☒ Inject passwords to the build as environment variables

Global passwords ☐

Job passwords

Name tcuser Password *****

Name tcpass Password *****

Add

Passwords list

Step-3: Go to >> Pre Steps & add as below

Pre Steps

Add pre-build step ▾

Build

Root POM	<input type="text" value="pom.xml"/>	
Goals and options	<input type="text" value="clean"/>	

Advanced...

Step-4: It's already add in the above so, we can skip

Post Steps

☐ Run only if build succeeds ☐ Run only if build succeeds or is unstable ☒ Run regardless of build result

Should the post-build steps run only for successful builds, etc.

Conditional steps (multiple) X

Run? **Strings match** ?

String 1 **\$(tag)**

String 2 **no**

Case insensitive ☐ ?

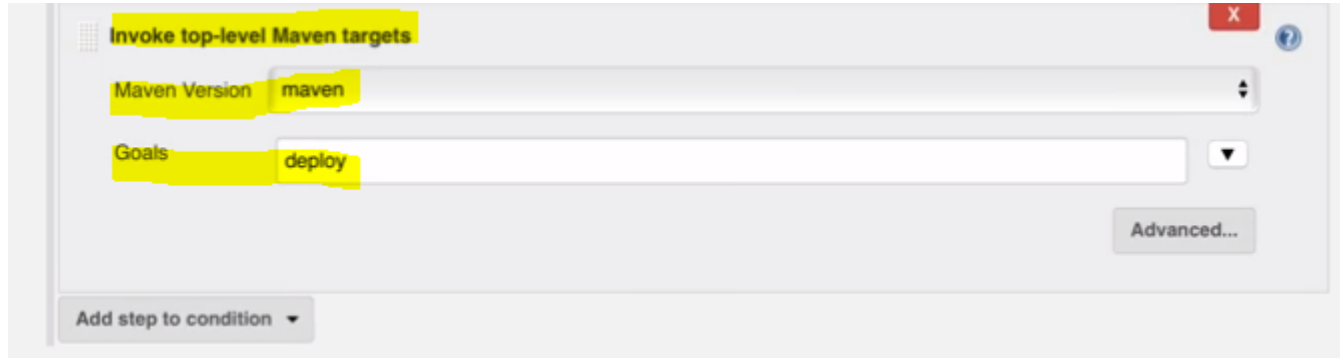
Advanced...

Steps to run if condition is met

Execute shell X ?

Command **echo "Building snapshot from branch \${branch}"**

Step-5: Add new parameter >> Conditional steps(multiple) and as below



Invoke top-level Maven targets

Maven Version:

Goals:

Advanced...

Add step to condition ▾



Conditional steps (multiple)

Run?

String 1:

String 2:

Case insensitive ☐

Advanced...

Steps to run if condition is met

Add step to condition ▾

Step-5: Add parameter >> Invoke top-level Maven targets & add as below

The screenshot shows the Jenkins configuration interface for a build step. The main section is titled "Steps to run if condition is met". It contains two steps:

- Execute shell**: The command field contains the text `echo "Building tag from branch ${branch}"`. Below the command field is a link that says "See the list of available environment variables".
- Invoke top-level Maven targets**: The "Maven Version" dropdown is set to "maven". The "Goals" field contains the text `release:clean release:prepare release:perform`.

Each step has a red "X" icon and a blue question mark icon in the top right corner. There is also an "Advanced..." button for each step.

Step-6: Copy the below data based on environment and update in Jenkins

TC Deploy:

DEV:

```
curl -u ${tcuser}:${tcpass} -T target/**/*.war "http://localhost:8080/manager/text/deploy?path=/devops&update=true"
```

QA:

```
curl -u ${tcuser}:${tcpass} -T target/**/*.war "http://localhost:9080/manager/text/deploy?path=/devops&update=true"
```

PROD:

```
echo "Deploying war from http://localhost:8081/artifactory/libs-release-local/com/example/devops/\${tag:1}/devops-\${tag:1}.war"
```

```
curl -O http://localhost:8081/artifactory/libs-release-local/com/example/devops/\${tag:1}/devops-\${tag:1}.war
```

```
curl -u ${tcuser}:${tcpass} -T *.war "http://localhost:7080/manager/text/deploy?path=/devops&update=true"
```

Windows Users (please note):

```
curl -u ${tcuser}:${tcpass} -F filedata=target/**/*.war "http://localhost:8080/manager/text/deploy?path=/devops&update=true"
```

Step-7: Add new parameter >> Conditional steps(multiple) and as below

- `curl -u ${tcuser}:${tcpass} -T target/**/*.war "http://localhost:8080/manager/text/deploy?path=/devops&update=true"`

Conditional steps (multiple) X

Run? Strings match ?

String 1 `${deploy}`

String 2 `yes`

Case insensitive ☐

Steps to run if condition is met

Execute shell X ?

Command `curl -u ${tcuser}:${tcpass} -T target/**/*.war "http://localhost:8080/manager/text/d`

[See the list of available environment variables](#)

Advanced...

Add step to condition ▾

Step-8: Click on Apply & Build based on develop branch

- Note : If you come across “Post Build Actions” i.e. Deploy a war/ear to Container

Post-build Actions

Deploy war/ear to a container

WAR/EAR files ****/*.war**

Context path **devops**

Containers

Tomcat 7.x

Manager user name **jenkins**

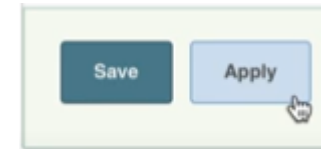
Manager password *********

Tomcat URL **http://localhost:8080**

Add Container ▼

Deploy on failure ☐

Delete



Step-9: Build, Deploy the build to dev environment & Verify.

Dev Environment : <http://ipaddress:8080/devops>

Jenkins > devops

Back to Dashboard
Status
Changes
Workspace
Build with Parameters
Delete Maven project
Configure
Modules

Maven project devops

This build requires parameters:

branch
origin/master
origin/feature/nv101_initial_landing_page
origin/develop

If you selected revisions as a type of presented time if you have a slow connection or repository

tag no
Option to build a snapshot or release.

deploy yes

Build

Build History

trend

#	Time
#4	Jul 29, 2017 8:27 AM
#3	Jul 29, 2017 8:25 AM
#2	Jul 29, 2017 8:17 AM
#1	Jul 29, 2017 8:10 AM

```
Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml (767 B at 51 KB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml
Progress (1): 767 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/1.0.0-SNAPSHOT/maven-metadata.xml (767 B at 37 KB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml
Progress (1): 313 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml (313 B at 13 KB/s)
Uploading: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml
Progress (1): 313 B

Uploaded: http://localhost:8081/artifactory/libs-snapshot-local/com/example/devops/maven-metadata.xml (313 B at 16 KB/s)
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 13.870 s
[INFO] Finished at: 2017-07-29T08:18:36-05:00
[INFO] Final Memory: 33M/379M
[INFO] -----
Finished: SUCCESS
```

DevOps Home About Contact

Welcome to the DevOps Course

Course Name:DevOps

Start a Release

1. Start a Release Flow and Update Version on develop branch
2. Tag the code for Release
3. Release code to QA from tag
4. Release code to Production from Artifactory
5. Finish current Release

Step-1: Create a branch & pull locally

The screenshot shows the GitHub interface for the repository 'keshavkummari / devops'. At the top, there are navigation tabs: Code, Issues (0), Pull requests (1), Projects (0), Wiki, Insights, and Settings. Below the repository name, there are statistics: 34 commits, 4 branches, 0 releases, and 1 contributor. A section titled 'Your recently pushed branches:' lists 'feature_branch' (10 minutes ago) and 'develop' (9 minutes ago), each with a 'Compare & pull request' button. Below this, there are buttons for 'Branch: release', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. The main content area shows a file tree for the 'release' branch, listing files like 'src', 'target', 'JenkinsFile', 'README.md', 'mvnw', 'mvnw.cmd', 'pom.xml', and 'pom.xml_bkp' with their respective commit messages and timestamps. At the bottom, there is a 'README.md' file viewer.

```
jenkins@ip-172-31-82-186:~/devops$ git fetch
From https://github.com/keshavkummari/devops
* [new branch]      release    -> origin/release
jenkins@ip-172-31-82-186:~/devops$ git pull
Already up-to-date.
jenkins@ip-172-31-82-186:~/devops$ git branch
* master
jenkins@ip-172-31-82-186:~/devops$ |
```

Step-2: Deploy the release from “develop” branch as below

```
jenkins@ip-172-31-82-186:~/devops$ git branch
* master
jenkins@ip-172-31-82-186:~/devops$ git checkout develop
Branch develop set up to track remote branch develop from origin.
Switched to a new branch 'develop'
jenkins@ip-172-31-82-186:~/devops$ vim pom.xml
jenkins@ip-172-31-82-186:~/devops$ grep -H -n "1.1.0" pom.xml
pom.xml:7:      <version>1.1.0-SNAPSHOT</version>
jenkins@ip-172-31-82-186:~/devops$
```

The screenshot shows the Jenkins configuration page for a Maven project named "Maven project devops". On the left, a sidebar contains navigation links: "Back to Dashboard", "Status", "Changes", "Workspace", "Build with Parameters", "Delete Maven project", "Configure", and "Modules". The main area is titled "Maven project devops" and states "This build requires parameters:". Below this, a "branch" dropdown menu is open, showing options: "origin/master", "origin/feature/hv101_initial_landing_page", "origin/release/initial_release" (which is highlighted), and "origin/develop". Below the dropdown, there is a note: "If you selected revisions as a type of presented c time if you have a slow connection or repository i". Further down, there are two parameters: "tag" with a value of "yes" and "deploy" with a value of "yes", both in green boxes. At the bottom, there is a large blue "Build" button with a hand cursor icon. Below the configuration area, there is a "Build History" section with a search bar and a table of recent builds. The table has two entries: build #4 from Jul 29, 2017 8:27 AM and build #3 from Jul 29, 2017 8:25 AM.

Back to Dashboard

Status

Changes

Workspace

Build with Parameters

Delete Maven project

Configure

Modules

Maven project devops

This build requires parameters:

branch

- origin/master
- origin/feature/hv101_initial_landing_page
- origin/release/initial_release
- origin/develop

If you selected revisions as a type of presented c time if you have a slow connection or repository i

tag yes

Option to build a snapshot or release.

deploy yes

Build

Build History

trend ==

find X

#4	Jul 29, 2017 8:27 AM
#3	Jul 29, 2017 8:25 AM