Build & Release Project ::

Using

1. Game Of Life Github link :: <https://github.com/wakaleo/game-of-life.git>

2. Jleetutorial Github Link :: https://github.com/jleetutorial/maven-project.git

------------------X----------------X--------------X-------------X----------------X--------

Prerequirements @ Jenkins { Pluggins }

1. Git Pluggin

2. Zentime Pluggin

3. Maven Pluggin

4. Copy Artifacts Pluggin

5. Deploy to Containers Pluggin

6. Build Pipeline Pluggin

------------------X----------------X--------------X-------------X----------------X--------

Different Phases in Maven Build Life Cycle ::

1. Validate

2. Compile

3. Test

4. Package

5. Verify

6. Install

7. Deploy

------------------X----------------X--------------X-------------X----------------X--------

First create 3 Instances in aws with seprate Security Groups

1. Jenkins-aws = Jenkins-SG = port 22 & port 8080

2 Tomcat-deployment-aws = TomCat-SG = port 22 & port 8080

3 Tomcat-production-server-aws = TomCat-SG = port 22 & port 8080

------------------X----------------X--------------X-------------X----------------X--------

Second Create 3 Instances

1. Jenkins aws Instance using Ubuntu 16.04 & add Jenkins-SG

2. Tomcat-deployment aws Instance using Ubuntu 16.04 & add TomCat-SG

3. Tomcat-production-server aws Instance using Ubuntu 16.04 & add TomCat-SG

------------------X----------------X--------------X-------------X----------------X--------

After Creating the Instances start all the 3 Instances

Now

Login to Jenkins web page using the public ip of the Jenkins Instance

@ web page

type

<Jenkins-aws-public-ip>:8080

In Jenkins

STEP 1 : Create a job or project in jenkins for package of game of life

Jenkins ==> select New Item ==> write the project name

==> Gameoflife-Package

==> select FREE STYLE PROJECT then click OK @ bottom of the page

now u will have a new page with the configuration of Gameoflife-Package Project or Job

In that new page do the following

==> In Source Code Management

give the git URL of Game Of Life Github

URL :: <https://github.com/wakaleo/game-of-life.git>

==> In Build

--> select ==> Invoke top-level Maven Target

Goal ==> clean package

==> In Post Build Action

--> select ==> Archive the Artifacts

type = \*\*/\*.war

Then Click on Save OR Apply & Save

Then Click on Build Now

check the output if its Success or Failed

if Success go to the next STEP 2

if Failed reslove the error & go to STEP 2.

STEP 2 :: Create a another job for Static Code Analysis

Go to Jenkins ==> select New Item ==> Enter Project Name

Gameoflife-code-analysis

Select ==> copy from option @ bottom of the page & give the First Job or Project name for STEP 1

Gameoflife-Package

Then Click Ok

U will be taken to the configuration step (a new configuration will be open with he copy of all the details from Gameoflife-Package )

Now do as the following steps

==> In Build

--> Selcet ==> Invoke Top-Level Mavan Target

Goal ==> checkstyle:checkstyle

==> In Post Build Action

--> Select ==> Publish Checkstyle Analysis Result

Note : donot type anything in the box given below

Then Click on Save OR Apply & Save

Then Click on Build Now

check the output if its Success or Failed

If Success follow the next step

If Failed Reslove the error & follow the next step

===============>

Now open the First Job Or First Project again “ Gameoflife-package ”

do as the following

==> In Post Build Action

--> Select ==> Build Other Project

Enter the 2nd Job or Project Name in the blank area

“ Gameoflife-code-analysis ”

Then Click on Save OR Apply & Save

Now Click on Build Job & Check the Output

Note : we will notice that after the success of Gameoflife-package Job , It is triggering “ Gameoflife-code-analysis ”

Check the Output : its successfull

if Failed resolve the error & go the next STEP 3

STEP 3 :: Create a new job for deploying the artifacts to TomCat Server

Go to Jenkins ==> Selcet New Item ==> enter the new project or job name

Gameoflife-deploy-to-tomcat

Select ==> FREE STYLE PROJECT ==> Then click OK @ bottom of the Page

now u will have a new page with the configuration of “ Gameoflife-deploy-to-tomcat ”

do as Following

==> In Build

--> Selcet ==> Copy Artifacts From Another Project

Project Name : Gameoflife-package

Artifacts to copy : \*\*/\*.war

==> In Post Build Action

--> Select ==> Deploy War/ear to a Container

WAR/EAR : \*\*/\*.war

Container : Tomcat 7.x

Usernamer : tomcat ( tomcat server login name )

Password : \*\*\*\*\* (Tomcat’s server’s password )

URL : <Private ip from aws ToMCAT>:8080

Now Click on Save OR Apply & Save

Before Clicking on Build Now

In the aws – go to Tomcat Instance & In the Security Group of Tomcat add the Jenkins Security Group ==> Then the TomCat server will all the jenkin to access the TomCat server

[ In shot we are telling the TomCat that when ever the request comes from Jenkins Blindly allow the Jenkins to access Tomcat server ]

Now Click on Build Now

Check the Output of the Job or Project

If the the Output is Successfull then follow the next step

if Failed reslove the error & go to the next step

====================>

Now open the 2nd Job or Project which we created

“ Gameoflife-code-analysis ”

do as the following

==> In Post Build Action

--> Selcet Projects to build

give the name of the 3rd project “ Gameoflife-deploy-to-tomcat ”

OR

==> In Build

--> Selcet Trigger/calls builds on other projects

Build Triggers

Projects to Build : “ Gameoflife-deploy-to-tomcat ”

Now Click on Save OR Apply & Save

Now go the Jenkins

Select first Job :: Gameoflife-package & click on Build Now

check the output ::

After “ Gameoflife-package ” Job or Projectis complete it will trigger “ Gameoflife-code-analysis ” Job or Project

After “ Gameoflife-code-analysis ” Job or Project is completed it will trigger “ Gameoflife-deloy-to-tomcat ” Job or Project

After completeting the Build Now

go to Tomcat web page

u will find the gameoflife is deployed to the sever (in the web)

After completeing the STEP 3 , Now start the STEP 4

STEP 4 :: Create a another job for the procdution deployment

Go to Jenkins ==> Select New Item ==> enter the Project Name

“ Gameoflife-production-tomcat ”

Then Select FREE STYLE PROJECT & click OK @ bottom of the page

Now a new page will be open with a where we have to set the following

==> In Build

--> Select ==> Copy Artifacts From Another Project

Project Name : Gameoflife-package

Artifacts to copy : \*\*/\*.war

==> In Post Build Action

--> Select ==> Deploy War/ear to a Container

WAR/EAR : \*\*/\*.war

Container : Tomcat 7.x

Usernamer : tomcat ( tomcat server login name )

Password : \*\*\*\*\* (Tomcat’s server’s password )

URL : <Private ip from aws ToMCAT 2 Instance>:8080

Now Click on Save OR Apply & Save

================>

Go back to “ Gameoflife-deploy-to-tomcat ” Project or Job

Now do as the following

==> In Post Build Action

--> Select ==> Build other project (manual step)

Down Stream Job :: “ Gameoflife-production-tomcat ”

Then Click on Save OR Apply & Save

Now go to Jenkins

Select ==> Gameoflife-package & Click on Build Now

Check the Output ::

Result ::

Gameoflife-package will b started ... After completing it will Trigger Gameoflife-code-analysis .. After Gameoflife-code-analysis is completed it will Trigger Gameoflife-deploy-to-tomcat

After Gameoflife-deploy-to-tomcat is completed it will Trigger Gameoflife-production-tomcat

Now login to Production TomCat Server @ web & check the gameoflife

Now in an Organisation we have ‘n’ number of jobs running

To find out which job is linked as Down Stream Job & Up Steam Job

To Find out the status of the Job [ success or failed | to reslove the errors ] we need to group the jobs in one place so that the person checking the status of the job can easly identity the flow , error & status of the job

for the above requirement we now use BUILD PIPELINE FLOW

Go to Jenkins

@ centre of the page u will find ALL followed by a ‘ + ’ symbol

click on the ‘ + ’ symbol we will file 3 Options

1st Option = Build Pipeline Flow

Select the Build Pipeline Flow

Enter the Name of Ur Choise :: “ GameOfLife-Pipeline ”

Then Click on Ok

a New page will b open

Now follow as the below steps

==> Select initial Job

enter the 1st Project or Job name

here the Project name is “ Gameoflife-package ”

Click on Apply & Save

Now @ Jenkins @ centre of the page u will fine ALL

In the same line after ALL

u will find the pipeline name which we have set “ Gameoflife-Pipeline ”

Click on that option

u will find all the linked jobs to the Gameoflife-package

Just lick on Build Now option

and see the OutPut.

Note :: After the completion of the STEP 4 we went back to previous Job & we click on

Build other Project (Manual Step)

==> in the pipeline , last job will wait for the user to click on start job

after the user clicks on the start job it turns into Green (job completed)

if we donot click on build now , it will remain in waiting state

This is Called Manual Overhead , where , we have to click on it Manually so that the Artifacts goes to the Prodution Server

This is called Continious Delevory , where all the above steps ==> Package – Code Analysis & deploy to Tomcat server is done Automatically

But when the deployment comes to Production server , is waiting for the permission from the Devloper

=============>

Now we will Do Continious Deployment

Go the Jenkins

Select ==> Gameoflife-deploy-to-tomcat

==> In Post Build Actions

--> Delete the option ==> Build other project (manual step)

--> And Select ==> Build other project

enter the name : “ Gameoflife-production-tomcat ”

now Click on Save OR Apply & Save

Go to Jenkins

@ centre of the Page , after ALL click on the pipeline name which we have set

& Click on Build Now

Check the Output

Check the TomCat Production Server @ web

Thank You

Srinivas.T