Jenkins Declarative Script

Jenkins path location in linux

/var/lib/Jenkins/workspace

go to Jenkins

Manage Jenkins—Manage plugin— search for pipeline – install

now create new job – name the job and select pipeline

now go to pipeline tab and write the declarative script in the below formate

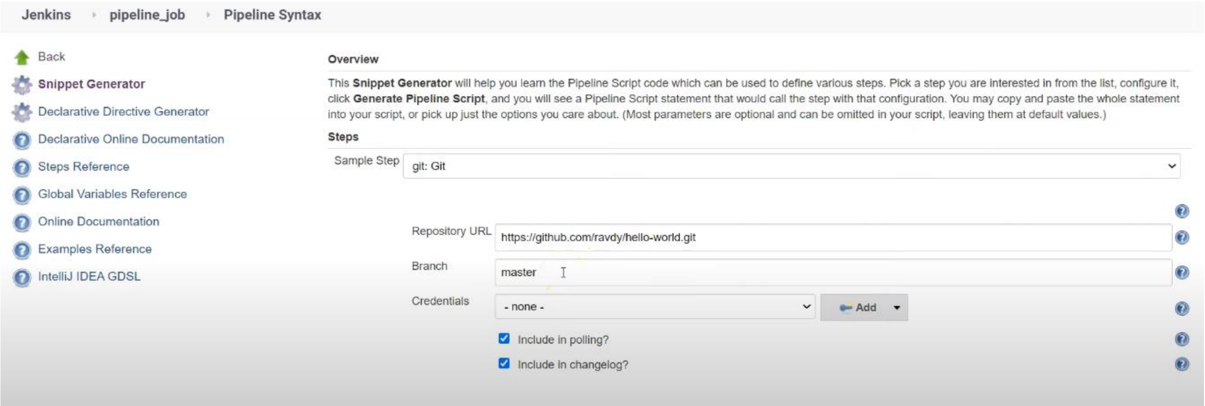
```
pipeline {
    agent any
    environment {
        PATH = "/opt/apache/bin:$PATH"
        }
        stages
            stage('git') {
                steps {
                    git url: 'https://github.com.git'
                }
            }
        stage('build') {
            steps {
                sh 'mvn clean install'
            }
        }
        stage('deploy') {
            steps {
                sshagent(['deploy_user']) {
                    scp webapp/target/webapp.war ec2-user@13.229.183.126: /opt/apache-tomcat-8.5.55/webapp
                }
            }
        }

}
```

git

for cloning git we new to use pipeline syntax tool which is in the bottom of the page

in pipeline syntax need to select git and provide repo url, branch , credentials



Jenkins › pipeline_job › Pipeline Syntax

Overview

This **Snippet Generator** will help you learn the Pipeline Script code which can be used to define various steps. Pick a step you are interested in from the list, configure it, click **Generate Pipeline Script**, and you will see a Pipeline Script statement that would call the step with that configuration. You may copy and paste the whole statement into your script, or pick up just the options you care about. (Most parameters are optional and can be omitted in your script, leaving them at default values.)

Steps

Sample Step: git: Git

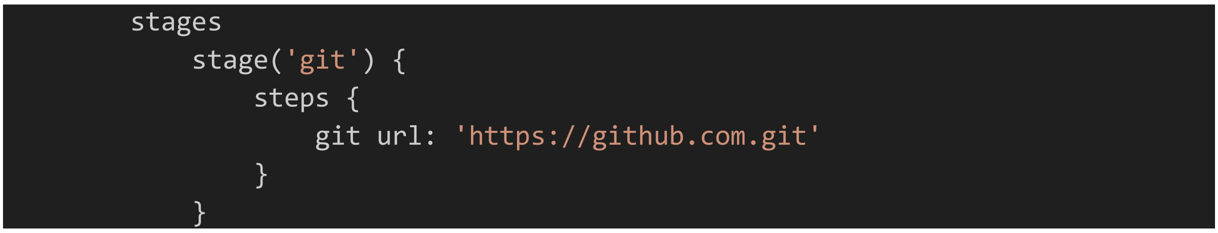
Repository URL: https://github.com/ravdy/hello-world.git

Branch: master

Credentials: - none -

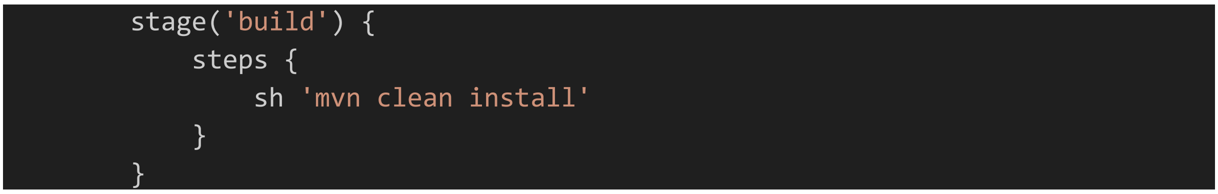
☑ Include in polling?

☑ Include in changelog?



```
        stages
            stage('git') {
                steps {
                    git url: 'https://github.com.git'
                }
            }
```

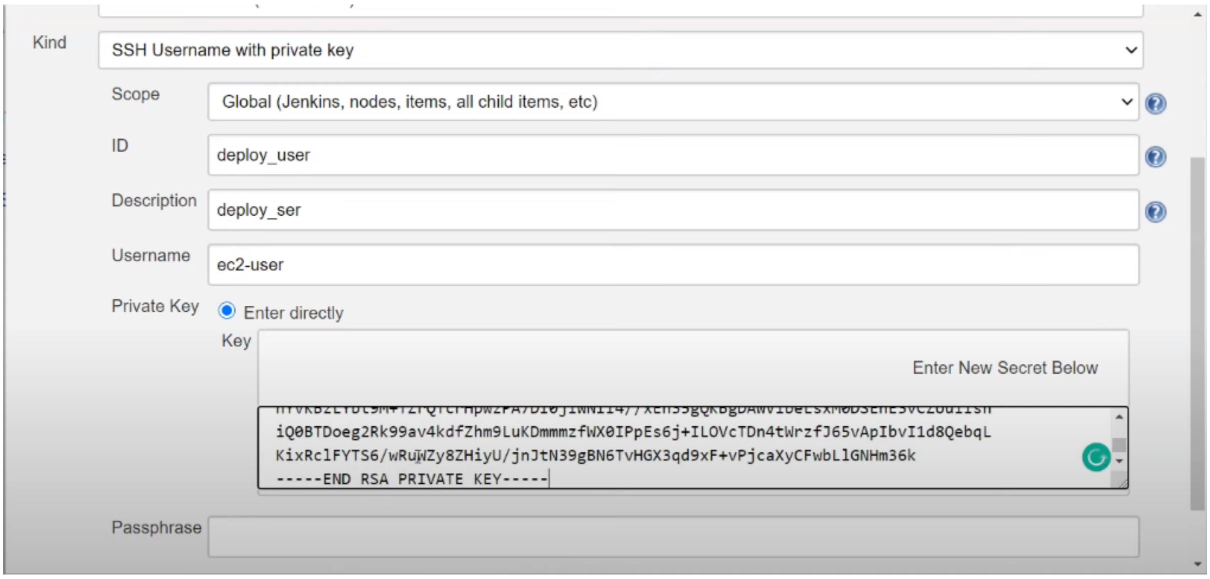
Maven



```
        stage('build') {
            steps {
                sh 'mvn clean install'
            }
        }
```

copying war file to tomcat server to slave machine with ssh agent

here we need to go pipeline syntax and select SSH Agent and add the credentials as shown below



Kind: SSH Username with private key

Scope: Global (Jenkins, nodes, items, all child items, etc)

ID: deploy_user

Description: deploy_ser

Username: ec2-user

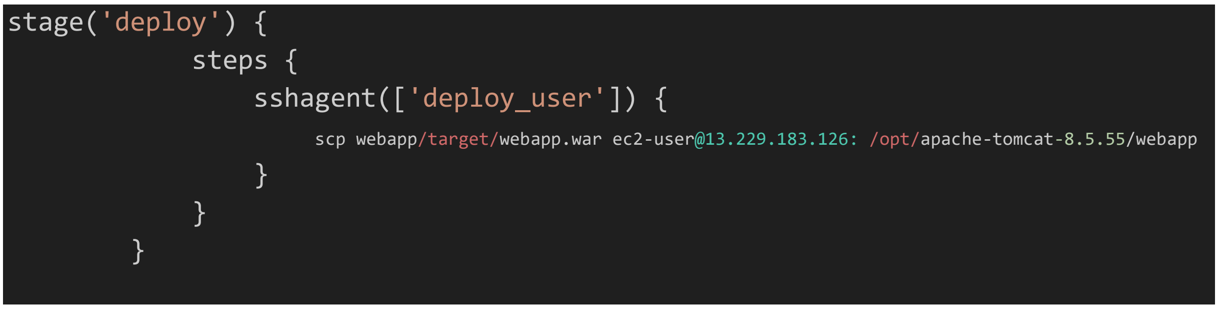
Private Key: ● Enter directly

Key: Enter New Secret Below

```
iQ0BTDoeg2Rk99av4kdfZhm9LuKDmmmzfWX0IPpEs6j+ILOVcTDn4tWrzfJ65vApIbvI1d8QebqL
KixRclFYTS6/wRuWZy8ZHiyU/jnJtN39gBN6TvHGX3qd9xF+vPjcaXyCFwbLlGNHm36k
-----END RSA PRIVATE KEY-----
```

Passphrase

and select the specific user and generate pipeline script and copy the code and add in script



```
stage('deploy') {
            steps {
                sshagent(['deploy_user']) {
                    scp webapp/target/webapp.war ec2-user@13.229.183.126: /opt/apache-tomcat-8.5.55/webapp
                }
            }
        }
```

scp <src_file> username@IP: <Dest_path>

here scp is copying file from one server to another slave machine(tomcat server)