$ su - root

$ yum install git -y

$ logout

$ git --version

ls

12 mkdir project1

13 ls

14 cd project1

15 ls

16 git init

17 ls

18 ls -a

19 echo "Hello"

20 echo "Hello" > sample.txt

21 ls

22 cat sample.txt

23 echo "Hello all" > abcd.doc

24 ls

25 cat abcd.doc

26 git config --global user.name pavansw

27 git config --global user.email pavansw@rps.com

28 ls

29 git add sample.txt

30 git status

31 cat sample.txt

32 echo "Hello again" >> sample.txt

33 cat sample.txt

34 echo "Hello again" >> sample.txt

35 cat sample.txt

36 git status

37 ls

38 git add sample.txt

39 git status

40 git commit -m "submitting sample.txt first time"

41 git status

42 ls

43 cat sample.txt

44 git log

45 history

============================

Git Checkout and Git Reset (--soft, --hard)

vi sample.txt

94 cat sample.txt

95 git status

96 git add sample.txt

97 git status

98 vi sample.txt

99 cat sample.txt

100 clear

101 cat sample.txt

102 git checkout sample.txt

103 cat sample.txt

104 ls

105 git add sample.txt abcd.doc

106 git status

107 git commit -m "Adding updated sample.txt and new file abcd.doc"

108 git log

110 echo "Pavan" > newdoc.txt

111 echo "Pavan" > newdoc1.txt

112 ls

113 cat sample.txt

114 cat abcd.doc

115 git reset --soft d7c9ffbeda60bec3f2743aaab842646442ea1694

116 ls

117 cat sample.txt

118 git log

119 cat sample.txt

120 git status

121 clear

122 git

123 clear

124 git log

125 ls

126 cat abcd.doc

127 cat sample.txt

128 cat newdoc.txt

129 cat newdoc1.txt

130 git add .

131 git commit -m "Adding all files"

132 git log

133 git reset --hard d7c9ffbeda60bec3f2743aaab842646442ea1694

134 git log

GitHub with Personal login and create on repo

pwd

139 cd ..

140 ls

141 git clone https://github.com/pavansw/accentureDevops1.git

142 ls

143 cd accentureDevops1

144 ls

145 ls -a

146 git config --global user.name pavansw

147 git config --global user.email pavanwankhade4u@gmail.com

148 vi first.py

149 vi README.md

150 ls

151 git add .

152 git commit -m "My initial commit for thesting github"

153 git log

154 git status

155 git push origin master

=================================================

DAY2

=====

**ON VM 1**

Hostname

$ su - root

12 hostnamectl set-hostname vm1.example.com

13 hostname

14 exec bash

15 clear

16 git clone https://github.com/pavansw/accentureDevops1.git

17 ls

18 cd accentureDevops1/

19 ls

20 git config --global user.name=pavansw

21 git config --global user.name pavansw

22 git config --global user.email pavanwankhade4u@gmail.com

23 git log

24 git log --oneline

25 clear

26 git log --oneline

27 ls

28 vi first.py

29 vi second.py

30 ls

31 git add .

32 git commit -m "user pavansw from vm1 commiting"

33 git log --oneline

34 clear

35 git log --oneline

36 git push origin master

**ON VM 2**

**$ su - root**

hostnamectl set-hostname vm2.example.com

7 hostname

8 exec bash

9 yum install git -y

10 git clone https://github.com/pavansw/accentureDevops1.git

11 ls

12 cd accentureDevops1/

13 ls

14 cat first.py

15 ls

16 git log --oneline

17 git config --global user.name pavansw1

18 git config --global user.email pavansatishwankhade@gmail.com

19 vi third.py

20 vi first.py

21 git add .

22 git commit -m "user pavansw1 commiting from vm2"

23 git log --oneline

24 git push origin master

25 clear

26 git log --onelone

27 git log --oneline

28 git pull

29 ls

30 cat first.py

31 git log --oneline

32 git reset --soft 3deae67

33 git reset --hard 3deae67

34 ls

35 git log --oneline

36 git pull

37 git log --oneline

38 vi first.py

39 ls

40 vi third.py

41\*

42 git pull

43 git commit -m "This is pavansw1 from vm2"

44 git push origin master

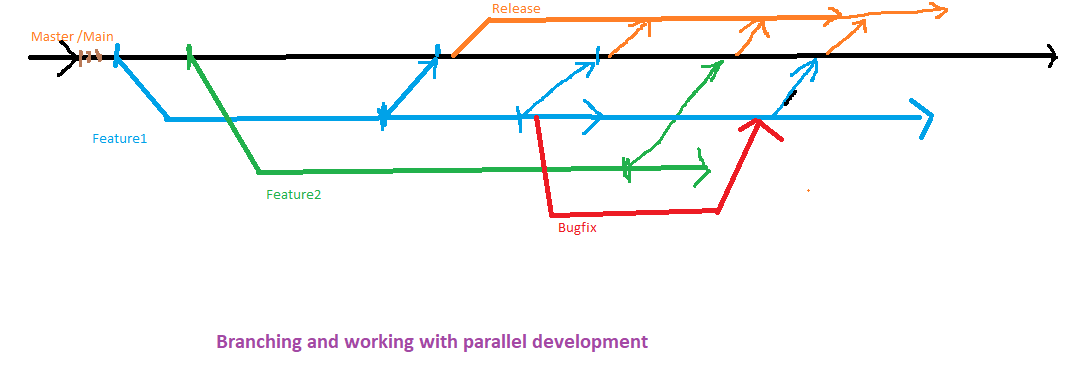
45 history

46 git add .

47 git commit -m "This is pavansw1 from vm2"

48 git commit -m "This is pavansw1 from vm2 again"

Git Branching and parallel Development



40 git branch

41 git branch feature1

42 git branch

43 ls

44 git checkout feature1

45 git branch

46 ls

47 vi newcode.py

48 vi newfeatures.py

49 ls

50 git checkout master

51 git branch

52 ls

53 git checkout feature1

54 git add .

55 git commit -m "new features added to feature1"

56 git push

57 git pull

58 git push

59 git push origin feature1

ls

62 git checkout master

63 git pull

64 git log --oneline

65 ls

66 git checkout feature1

67 ls

68 vi newcode.doc

69 git add newcode.doc

70 git commit -m "Documents uploaded"

71 git push origin feature1

72 clear

73 git branch

74 ls

75 git log --oneline

76 git checkout master

77 git branch

78 ls

79 git merge feature1

80 ls

81 git log --oneline

82 git branch

83 git push origin master

===================================================

Jenkins Installation on VM1

hostname

89 cat /etc/os-release

90 clear

91 yum install java -y

92 java -version

93 wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo

96 rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key

97 yum install jenkins -y

98 systemctl start jenkins

99 systemctl status jenkins

100 vi /etc/sysconfig/jenkins

DONT CHANGE ANYTHING , only note down port no

101 firewall-cmd --permanent --add-port=8080/tcp

102 firewall-cmd --reload

103 hostname

104 hostname -I

Note down IP and from Win/host machine browser

Open <http://192.168.131.128:8080>

Initial Admin pass is here in this file

105 cat /var/lib/jenkins/secrets/initialAdminPassword

Then select Install Suggested Plugins

===========================================================

Maven-Java Projects

===================

java -version

122 yum install maven -y

123 mvn --version

124 git clone https://github.com/pavansw/simpleMavenJunit.git

125 ls

126 cd simpleMavenJunit/

127 ls

128 ls src/main/java/hello/Greeter.java

129 pwd

130 ls

131 mvn clean

132 clear

133 ls

134 mvn compile

135 ls

136 ls target/

137 ls target/classes

140 mvn test

141 ls

142 ls target/surefire-reports/

146 mvn package

147 ls

148 ls target/

149 cd target/

150 ls

151 java -jar gs-maven-0.1.0.jar

152 ls

153 cd ..

154 ls

155 mvn clean

========================

Project myJavaProject

Compile, Test, Package my first Java project

Source Code: <https://github.com/pavansw/simpleMavenJunit.git>

Master branch

Build Trigger:

Poll SCM : Schedule: \*/3 \* \* \* \*

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
|  | | |  |

============================================================

Jenkins Master-Slave node setup on 2 CentOS 7 vms

On VM1

id jenkins

164 passwd jenkins

165 usermod -s /bin/bash jenkins

ON VM2

cd ..

79 whoami

81 useradd jenkins

82 passwd jenkins

83 mkdir /important

84 chown jenkins:jenkins /important

85 ls -l /

87 hostname -I

88 java -version

On VM1

166 su - jenkins

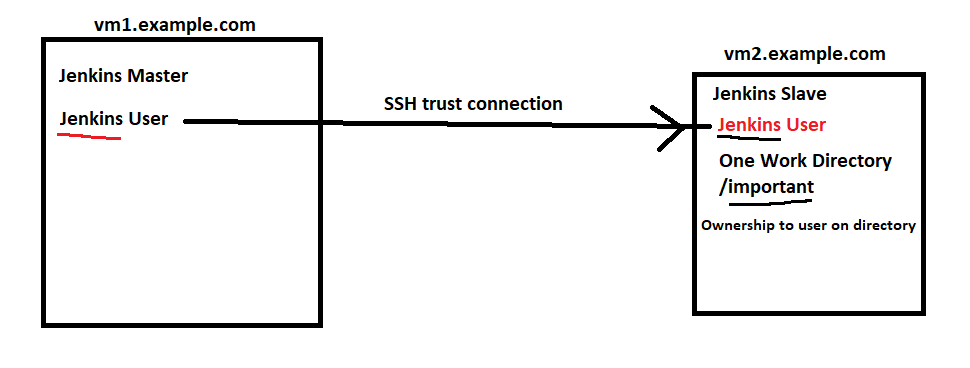
whoami

2 ssh-keygen

3 ssh-copy-id jenkins@192.168.131.129

$ exit

Now need to do setup from Jenkins Dashboard



============================================================================

Pipeline Example

pipeline {

agent any

stages{

stage('Code') {

steps {

echo 'This is code section'

}

}

stage('Build'){

steps{

echo 'This is Build Section'

}

}

stage('Test'){

steps{

echo 'This is Test Section'

}

}

stage('Deploy'){

steps{

echo 'This is Deploy Section'

}

}

}

}

=====================================================

Create Pipeline from SCM

Ex.

<https://github.com/pavansw/accentureDevops1.git>

File name : Jenkinsfile

===============================================================

Deployment Process

-------------------------------

On VM2 Install and Configure tomcat

systemctl status tomcat

93 yum install tomcat

94\* yum install tomcat-webapps tomcat-admin-webapps tomcat-docs-webapp tomcat-javadoc

95 clear

96 systemctl stop httpd

97 systemctl start tomcat

98 systemctl enable tomcat

99 systemctl status tomcat

100 firewall-cmd --permanent --add-port=8080/tcp

101 firewall-cmd --reload

===================================================================

Docker

whoami

188 docker --version

189 yum install yum-utils -y

192 yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo

193 yum repolist

194 yum install docker-ce docker-ce-cli containered.io

195 systemctl start docker

196 systemctl status docker

197 systemctl enable docker

198 docker --version

199 docker version

==============================

Docker Containers

docker images

218 docker search ubuntu

219 docker pull ubuntu

220 docker images

221 docker ps

222 docker ps -a

223 docker run -i -t ubuntu

cat /etc/os-release

touch test.txt

ls

exit

224 docker ps

225 docker ps -a

226 docker run -i -t ubuntu

227 docker ps

228 docker ps -a

============================

docker pull opensuse/tumbleweed

246 docker ps

247 docker ps -a

248 docker images

249 docker run -it opensuse/tumbleweed

250 history

251 docker ps -a

252 docker run -it -d opensuse/tumbleweed

253 docker ps

254 docker exec -it 17b31bf24902 bash

255 docker ps

256 docker ps -a

257 docker exec -it 17b31bf24902 ls /

258 docker exec -it 17b31bf24902 bash

259 docker ps

260 docker stop 17b31bf24902

261 docker ps

262 docker ps -a

263 docker start 17b31bf24902

264 docker ps

265 docker start ecc27b97266b

266 docker ps

267 docker exec -it ecc27b97266b bash

268 docker exec -it 17b31bf24902 bash

269 docker ps

270 docker stop 17b31bf24902

271 docker stop ecc27b97266b

272 docker ps

273 docker ps -a

=======================================

298 clear

299 docker ps

300 docker ps q

301 docker ps -q

302 docker stop $(docker ps -q)

303 docker ps

304 docker ps -a

305 docker rm 45f3c4d30850

306 docker ps -a

307 clear

308 docker ps -aq

309 docker rm $(docker ps -aq)

310 docker ps

311 docker ps -a

312 ls

313 clear

314 docker images

315 docker rmi 7666c92f41b0

316 clear

317 docker run -itd --name computer1 ubuntu

318 docker ps

319 docker run -itd --name computer2 ubuntu

320 docker ps

321 docker inspect computer1

322 docker inspect computer2

331 docker stop computer2

332 docker ps

333 docker ps -a

334 docker inspect computer2

335 docker start computer2

336 docker inspect computer2

====================================

Port mappings and Network services

docker run -d --name mywebsite1 -p 18080:80 httpd

344 docker ps

348 ip a

349 docker ps

350 docker inspect mywebsite1

351 curl http://172.17.0.3

352 history

353 docker run -d --name mywebsite2 -p 28080:80 httpd

358 docker run -d --name mywebsite4 -p 4545:80 httpd

359 history

Access Using HOST Vm1 IP and Port no

360 docker ps

361 docker exec -it mywebsite4 bash

echo "Hello this is Pavan" > htdocs/index.html

=================================================================

Using Databases in Conteiners

docker stop $(docker ps -q)

365 docker ps

366 docker run -d -e MYSQL\_ROOT\_PASSWORD=**rps** --name mydb1 mysql:5.6

367 docker ps

368 docker exec -it mydb1 bash

mysql -u root -p

show databases;

exit;

exit

369 docker ps

370 docker run -d -e MYSQL\_ROOT\_PASSWORD=rps -e MYSQL\_USER=pavan -e MYSQL\_PASSWORD=rps -e MYSQL\_DATABASE=accenture --name mydb2 -p 3306:3306 mysql:5.6

371 docker ps

372 docker exec -it mydb2 bash

mysql -u pavan -p

show databases;

exit;

exit

373 history

374 docker inspect mydb1

375 clear