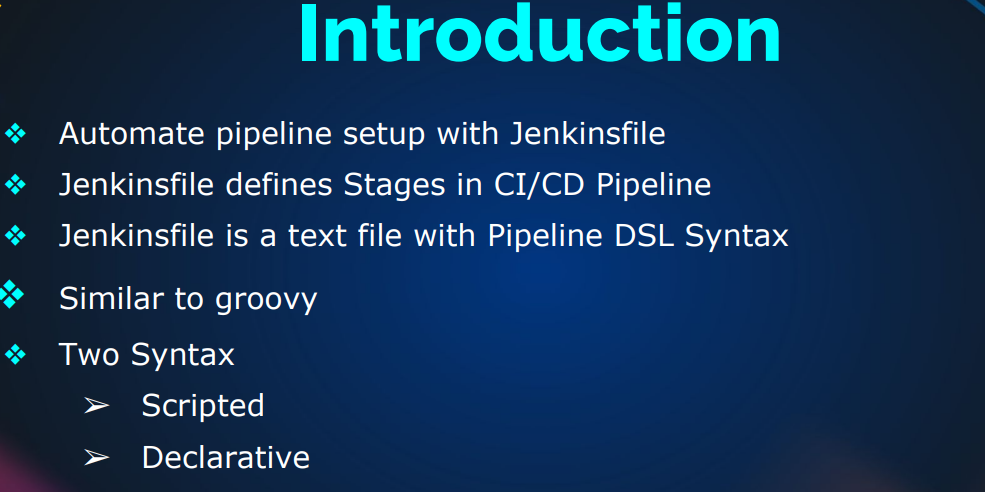
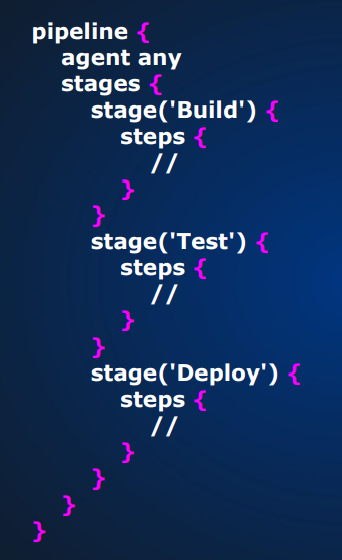
****

****

** **

**Jenkins Jobs**

Jobs basically means workload. There are two types of jobs:

* Freestyle Jobs:
* Graphical jobs
* Used for Learning, Understanding & exploring Jenkins
* Not recommended in Real time now
* Pipeline As a code
* Pipeline
* It is recommended now

**CREATING FREESTYLE JOBS IN JENKINS**

**Installing: Jenkins**

* create an EC2 instance (Select Ubuntu and name it as Jenkins server) and provision it with the Jenkins code to install (check doc-bash scripts to install Jenkins)
* Select the instance type as t2 micro free tier (This is only for small scale devs and testing)
* If your instance becomes unresponsive, then you need to increase the instance type
* Create its keypair, name it as Jenkins-key. Select RSA and .pem options. Save the Keypair
* After saving, make sure the keypair is selected.
* Under network settings, click on edit in the EC2 instance creation
* click on create security group and name it as Jenkins-SG and give same name to Description.
* Under Inbound security groups rules: (ssh … TCP …. 22 …. my IP …. select your IP)
* Click on Add another rule: (Custom TCP ….. TCP …. 8080 …. My IP …. select your IP)
* If you shutdown and poweron again, your IP may change so you need to request for a static IP from AWS or you reset following previous step (Elastic IP)
* Remember to provision it from first step above
* Click on launch instance
* Get the public IP and use it to access through your browser (e.g. 3.89.49.9:8080)
* Copy the password path you see in the browser
* Open your git bash CLI and ssh to jenkin (Example: ssh –i Download/Jenkins-key.pem [ubuntu@3.89.49.9](mailto:ubuntu@3.89.49.9))
* type yes to continue
* Check your provision script for errors (curl <http://169.254.169.254/latest/user-data>)
* systemctl status Jenkins (type q and enter key to return)
* Once Jenkins is installed the home directory will be (/var/lib/jenkins/)
* cd /var/lib/jenkins/
* ls
* once it opens, cat into the password path you see in the page to get the login password (Example: cat /var/lib/jenkins/secrets/initialAdminPassword)
* copy the unlock password (usually 32 digits long)
* Go to your browser and paste to unlock Jenkins
* click on: select plugins to install
* scroll through professionally and click on useful plugins
* click on install
* Setup your Admin User: (Username: Admin, password: your choice, full name: Admin, E-mail address: provide your official email address)
* Click on Save and continue
* take note of the Jenkins url in case your IP has changed
* click on save and finish
* click on start using Jenkins

**Installing tools in Jenkins:**

**STEP 1:**

* once you have access, click on manage Jenkins,
* click on global tool config
* click on add jdk(Name=OracleJDK8)
* Under install Oracle java SE Development kit, select (Java SE Development Kit 8u221) from the drop down
* check I agree to the java SE Development Kit License Agreement

OR:

* if it comes out with errors, then (uncheck the install automatically option) and follow steps below
* go to git bash and ssh into the EC2 instance to continue (ssh –i Downloads/Jenkins.pem ubuntu@(IP)
* Type yes and enter
* clear
* sudo apt update
* sudo apt install openjdk-8-jdk –y
* java -version (to check current running Java)
* sudo –i
* ls /usr/lib/jvm (to check the java for tools if you see java 1.8 then you are good)
* /usr/lib/jvm/java-1.8.0-openjdk-amd64 (to specify the working path)
* go back to your Jenkins server in your browser and paste the working path (/usr/lib/jvm/java-1.8.0-openjdk-amd64) under JAVA HOME box.
* scroll down to Maven section and click on add maven in the same page on your Jenkins server in your browser (Name=MAVEN3, version=3.8.6)

**STEP 2:**

When you are done with step 1 above, following these steps:

**Installing more tools in Jenkins:**

* once you have access, click on manage Jenkins,
* click on global tool configuration
* All tools you see is becos you have activated the plugin feature ready for installation
* Go to Maven section, click on Add Maven
* If lucky you will see the install automatically option, give a name
* select the right version and click on save.
* To set the right version of Maven to use for the Job, either from the system or from the one you specified
* click on the job and click on configuration settings
* Under General tab, scroll down to Maven section and select the Maven Version (Default means from the system or you select the one you created)
* Click on save below
* You can follow the steps above for other tools apart from Maven

**PLUGINS FOR CI**

Nexus

Sonarqube

Git

Pipeline Maven Integration Plugin

BuildTimestamp

Slack Notification

**Steps:**

* Login to Jenkins through the browser, click on manage Jenkins, then click on manage plugins
* click on available tab and search for and check the box each of:

Nexus Artifact uploader

SonarQube Scanner

Build Timestamp

Pipeline Maven Integration

Pipeline Utility steps

Slack Notification

Git (sometimes it is installed by default) …………….CLICK ON INSTALL WITHOUT RESTART

**To view your Artifacts**

* Go to workspace and click on Target directory
* Note: each time you click on build, it replaces the Artifacts. This causes breaks in our code execution
* You need to do versioning
* If versioning is setup, you will see the folder versioning

**Versioning, Plugins & More**

**Jenkins Variables(e.g. Build-ID)**

* Click on New Item, to create a new job from the existing one
* Give it a name (e.g. Versioning-Builds)
* Select Freestyle and scroll down to copy from (choose the former job that has the artifacts)
* click on ok
* In the next page, scroll down to post-build actions and take out the archive setting if it is there
* click on save and run the job by clicking on build now
* click on configure on the left pane and scroll down to the Build section
* click on Add build step and select Execute shell scripts
* type the following commands:

mkdir –p versions

cp target/vprofile-v2.war versions/vprofile-V$BUILD\_ID.war

* Please note vprofile-v2 represents the name of the artifact for these project
* click on save and run the build severally to test
* go to workspace, you will see a folder named versioning
* click to open to view the various versions

**For Individual paraments instead of Jenkins Variable (Please note: It is not the best practice, becos it will break job)**

* After completing the steps above
* click on Add parameter and select String parameter
* Give name as VERSION (Any time you run these job, it will ask you for version number and store it against the VERSION given)
* scroll down to Execute shell section and type:

mkdir –p versions

cp target/vprofile-v2.war versions/vprofile-V$VERSION.war

* Click on save and click on Build with Parameters

**FOR TIME STAMP INSTEAD OF JENKINS VARIABLE**

* When you login to Jenkins through your browser, click on Manage Jenkins
* click on Manage Plugins and click on Available tab
* search for Build Timestamp, click on install without restart
* click on the versioning build job and click on configure
* Deselect these project is parameterized and select change date pattern for the Build\_timestamp
* Click on the ? at the end of the text box to view the format (example: yy-MM-dd\_HHmm)
* scroll down to Execute shell section and type:

mkdir –p versions

cp target/vprofile-v2.war versions/vprofile-V$BUILD\_ID -$BUILD\_TIMESTAMP.war

* Click on save and click on build now
* check workspace, click on versions folder to confirm

Check Doc:

How to create jobs

Pipeline jobs in Jenkins