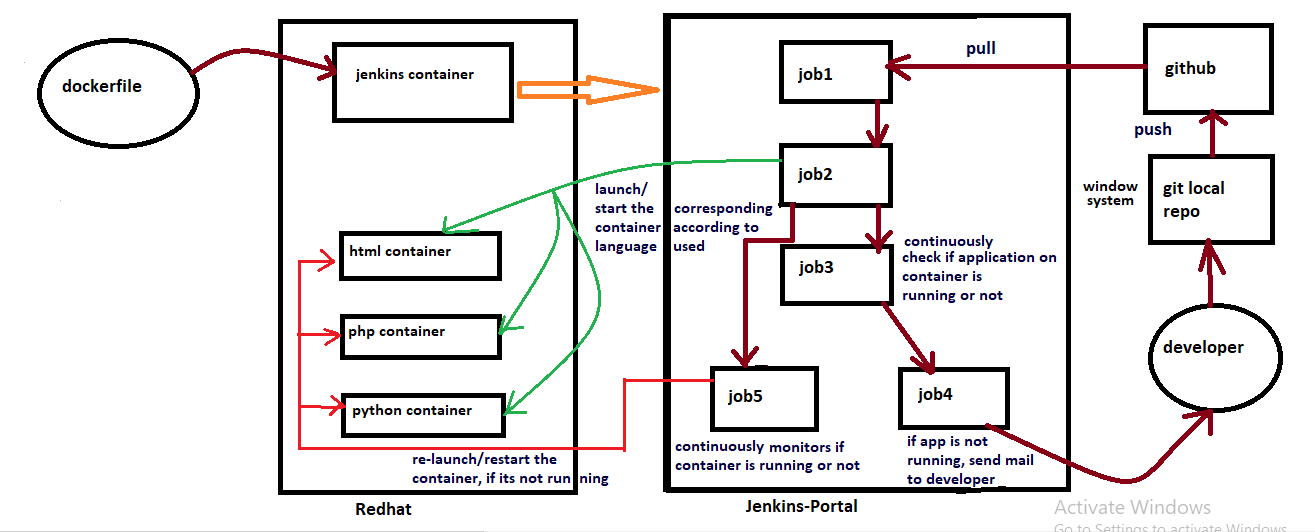
**Continuous Integration and Continuous Deployment**

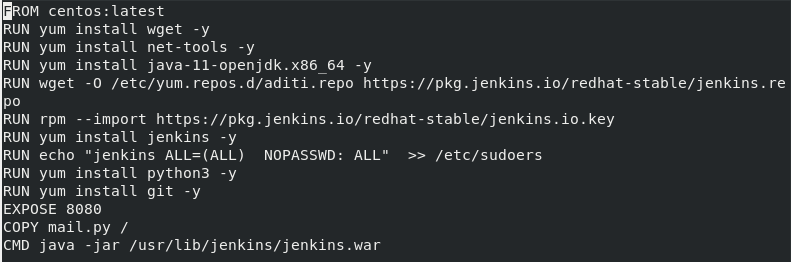
**Problem Statement:**

1. Create container image that has Jenkins installed using dockerfile
2. When we launch this image, it should automatically starts Jenkins service in the container.
3. Create a job chain of job1, job2, job3 and job4 using build pipeline plugin in Jenkins
4. Job1 : Pull the Github repo automatically when some developers push repo to Github.
5. Job2 : By looking at the code or program file, Jenkins should automatically start the respective language interpreter install image container to deploy code ( eg. If code is of PHP, then Jenkins should start the container that has PHP already installed ).
6. Job3 : Test your app if it is working or not.
7. Job4 : if app is not working , then send email to developer with error messages.
8. Create One extra job job5 for monitor : If container where app is running. fails due to any reson then this job should automatically start the container again.

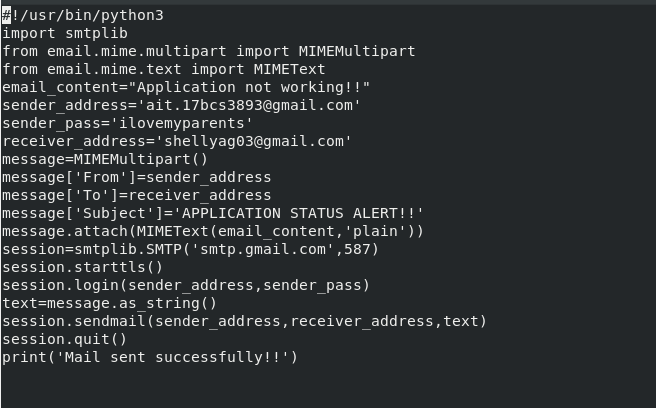
**Flowchart :**



**Dockerfile for creating jenkins image:**



**mail.py for sending mail to developer:**



**Building the Jenkins image from Dockerfile:**

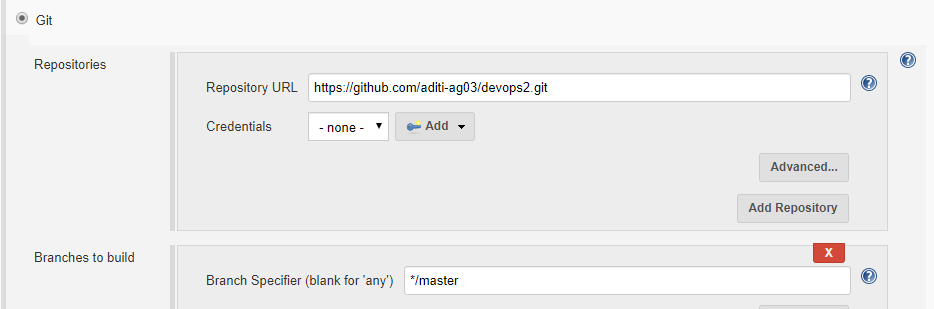
**#docker build -t jenkins\_image Dockerfile**

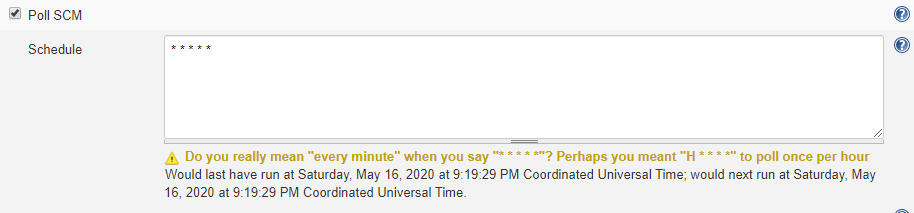
**Container launching through Jenkins image:**

**#docker run -dit –name jenkins\_c1 -P -v /:/host jenkins\_image**

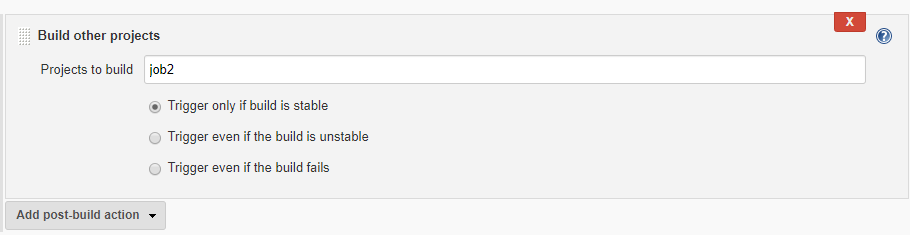
**Job 1:**

It pulls the github repository through poll scm and copy all the files with different extensions to their respective folders



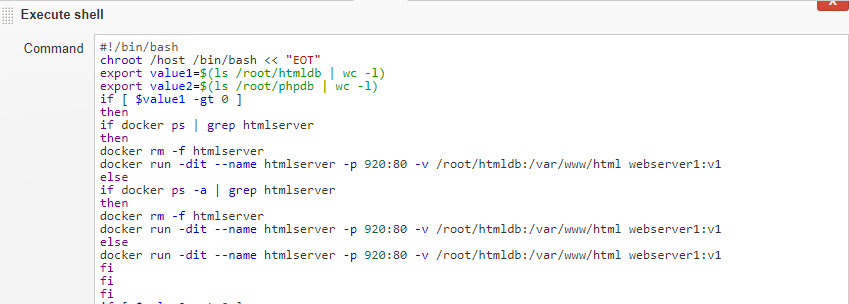


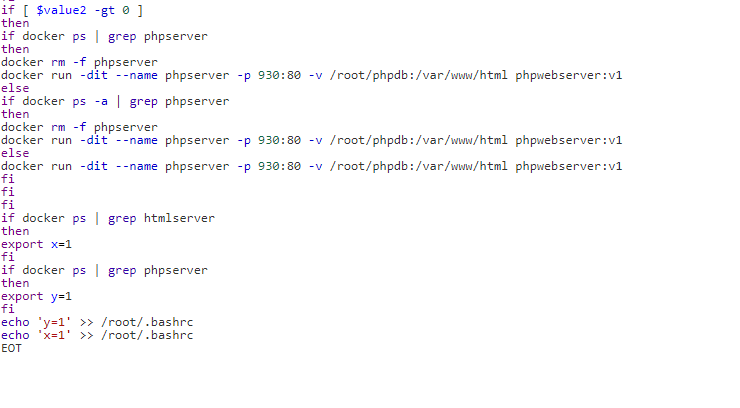


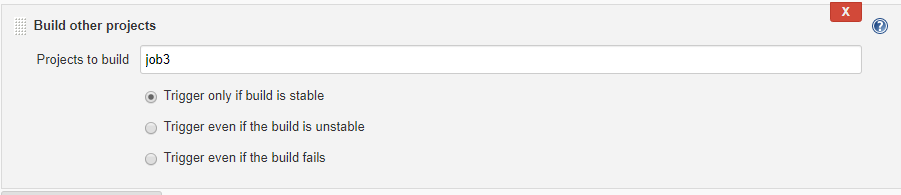


**Job 2:**

It will launch the container according to the program file received

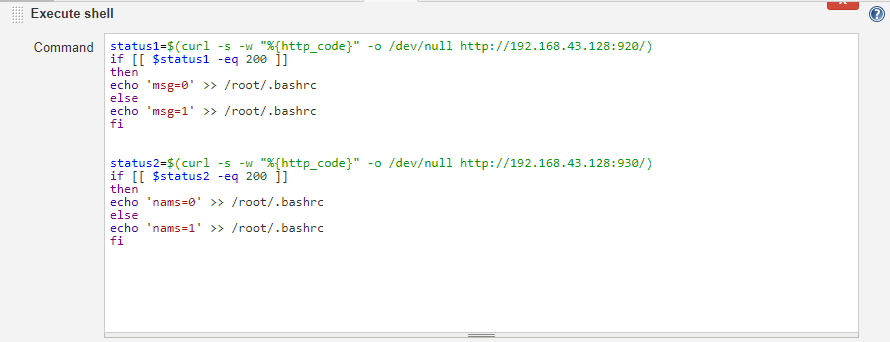


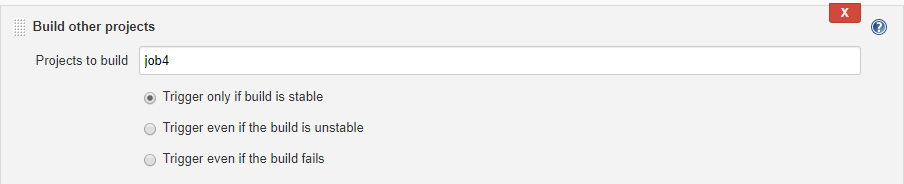




**Job 3:**

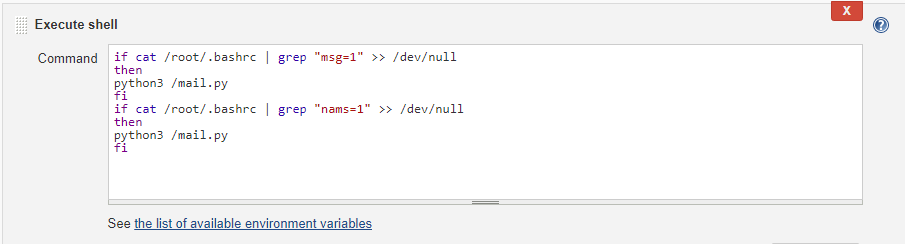
If applications on respective containers are not working it will trigger the job 4





**Job 4:**

It will send the mail to the developer if application is not working



**Extra:**

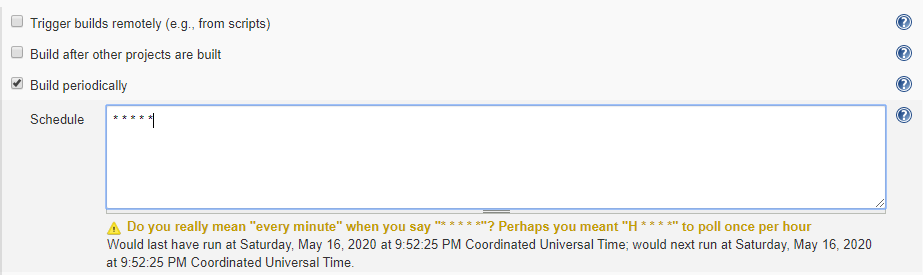
**#chroot /host /bash/bin << “EOT”**

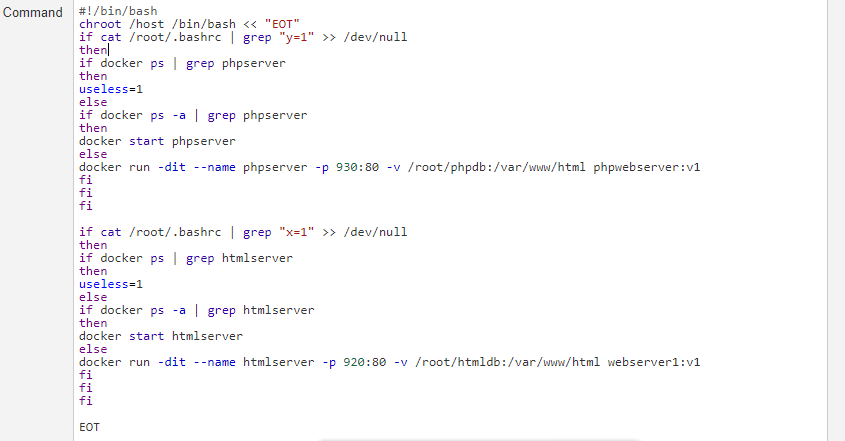
**#EOT**

This is a special trick for using the host OS from container, we are using this from Jenkins container to work inside base OS. (in my case RHEL8)

**Job 5:**

It is an independent job, which every minute monitors the containers launched by job 2 are still running or not, if not, then it run them again





**Build Pipeline:**

