* Creating a pipeline project in Jenkins for back end and pushing the image in Elastic Container Registry

1. Copy the below code to download the package and run the Jenkins in local instance

1. 
2. Open the Jenkins page using the public-ip
   1. Open Browser
   2. Public-ip:8080
3. Enter the details according the instructions and wait for few seconds the packages to get installed.
4. Click on New item at the left corner, give some proper name to the item and select the pipeline from option and press ok button below
5. Scroll down and you will find the pipelie area to write the block.
6. We will create the pipeline in block by block format.
7. Before moving forward the skeleton of the pipeline is –
   1. SKELETON
8. In first build we will install git in our Jenkins and pull the file from git and check if it is working or not

* 1. 

1. Now this code will show error as the Jenkins has no sudo access. We will add Jenkins to sudors so that it can run properly withput asking the password (in real time scenario we don’t give all access to the Jenkins users as it will affect or may cause major issue.
2. Open the linux in your any environment and add the Jenkins in sudors and add the Jenkins user below the root user also we will add the nopasswd so that it would not ask for password of the user when it would be used.
   1. sudo vim /etc/sudoers
   2. root ALL=(ALL) ALL
   3. jenkins ALL=(ALL) NOPASSWD:ALL
3. Now click on the build if it is working or not, if it is not then check the previous steps for any that you have left mistakenly.
4. Now we will create a new block that will be used to create a dockerfile in our Jenkins

* 1. 

1. Now we will build this docker file that we have pulled and shift the student artifact to new working directory and build an image

* 1. 

1. Now we have built the file, and need to have to push this image to a container, generally we used to push this image in our docker but we have to automate this thing via Jenkins pipeline but to do so it would require us enterprise version but we are doing this for free version so we will use “Amazon Elastic Container Registry”.
2. Go to search bar of the AWS and search “Amazon Elastic Container Registry” and select the service. Click on Get Started Button under a Create a Repository.
3. Give some proper name for your repository and let other settings to default and click on Create Repository.
4. Now click the name of the repository you created and you will get inside your repository. Check the right side and you will see the button with “View Push Command” click on it. (you can also just select the image and the button would be active outside as well it is good if you wanna push in more then one repository)
5. Now go back to your instance and select the instance you’re working on and click on Action search for Security and click on it and select “Modify IAM role”. Select the IAM role that has admin access and then click on “Update IAM role”.
6. G back to AECR service and pick the desire code (According to your system). For this we will select AWS CLI (it will help to access the pipeline and service to communicate) Image tag (to tag the image with unique name) and then we will need to push the image to our repo so.
   1. aws ecr get-login-password --region ap-south-1 | sudo docker login --username AWS --password-stdin 203772634753.dkr.ecr.ap-south-1.amazonaws.com
   2. sudo docker tag studentapp 203772634753.dkr.ecr.ap-south-1.amazonaws.com/tomcat-student:latest
   3. sudo docker push 203772634753.dkr.ecr.ap-south-1.amazonaws.com/tomcat-student:latest
7. Now open the service and select the repository that you have created and check if the image is been uploaded or not(check only after pipeline is run successfully). Click the service refresh button if the page is on.