jenkins class-02

this is practically we will do this here

JENKINS CLASS-1 (JENKINS SETUP) STEP-1: LAUCN AN EC2 INSTANCE WITH SEPARATE SECURITY GROUP STEP-2: GO TO JENKINS.IO AND COPY PASTE THOSE 2 LINKS STEP-3: INSTALL JAVA STEP-4: INSTALL JENKINS STEP-5: START THE JENKINS links-1: sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhatstable/jenkins.repo link-2 : sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key To install java11 : amazon-linux-extras install java-openjdk11 -y To install jenkins : yum install jenkins -y To start jenkins : systemctl start jenkins To check jenkins status : systemctl status jenkins To stop jenkins : systemctl stop jenkins

create a new job

different types jobs

freestyle,pipeline jobs,multi branch pipeline

first will go freestyle job

go to jenkins and create free style project and goto github project paste the bitbucket/github link here i select github tic-tok docker

goto source here we learn github or bitbucket here i seect github tic-tok docker
note: public repo do not require credentials so we can enter non and build the project

devs rasina code ci ki tisukaravadam devops

default path of the jenkin sis /var/lib/jenkins here it will store all deployments folder and files and the out will store in the path

everyjob will store inside workspace

/var/lib/jenkins/workspace

here we can the all files in deploymnet folder github lo unna prati code ci server ki vastundi

same thing another job create and do one more change here branch

example in localhost jenkins **Genkins-freestyle-job2**

devs rasina code ci server ki tisukuntam

as we did till now public repo need to now private repo

so here while create a job here and while selecting the job here it will select and enter credentials by default here it will not present so we have to add here

go to add jenkins select global credentials with username and password where as username we have to give github username and now where as paassword we have to give token

so fot that token we have to go github-->and seeting -->devlopers seeting -->personal acces token -->classic token
if we have token we can also regenrate the token so where has password here paste the token -->description lo github password

this is manual work not automation work

automation ante devloper when commit the code it will deploy
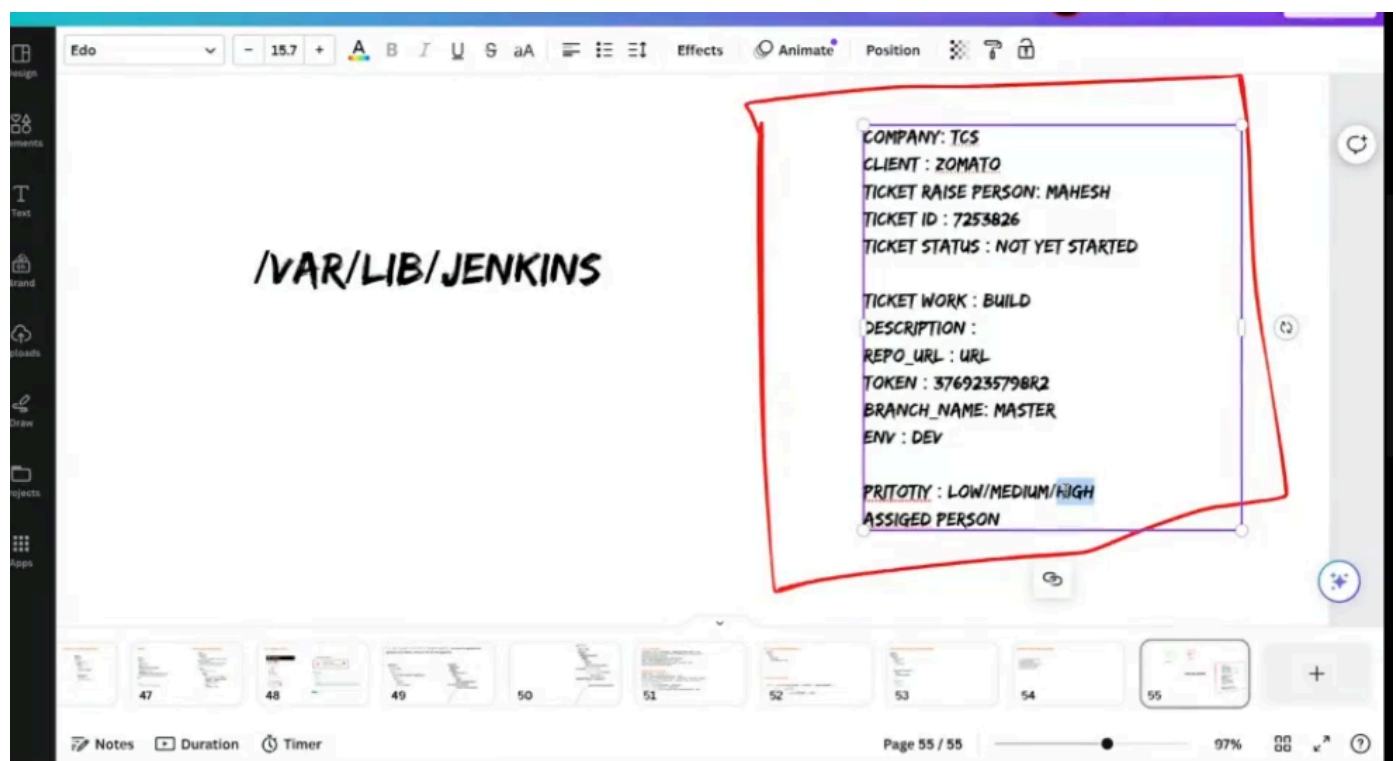
for deployment ticket raise and recive implements point

this all manual

so in the automatic we have to use webhooks consept implements so when dev complete the commit it will deploy and came to ci server

webhooks will present in github

now let us consider github goto repository settings and left side you will see wehbhook click on webhooks and add webhook

in the payload url we have to give the jenkins server url as well as github-webhook name

example http://43.204.123.43:8080/github-webhook/

content type we have to select application js type

then add webhooks and click on refresh button if we see right click symbol it is succuful

now go to jenkins job configure

in github we have to enble webhook and where as in jenkins we have to enable GitHub hook trigger for GITScm polling? in build triggers

and save and goto github repo which you have configure the webhook and take one file and edit and commit there it will automatically trigger

Console Output

✓ **Console Output** 🗐

```
Started by GitHub push by Harishjangidi
Running as SYSTEM
[EnvInject] - Loading node environment variables.
Building on the built-in node in workspace /var/lib/jenkins/workspace/Test_Job
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Test_Job/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/Harishjangidi/tic-tac-toe-docker.git # timeout=10
Fetching upstream changes from https://github.com/Harishjangidi/tic-tac-toe-docker.git
 > git --version # timeout=10
 > git --version # 'git version 2.25.1'
 > git fetch --tags --force --progress -- https://github.com/Harishjangidi/tic-tac-toe-docker.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 550cf6de024ed8e0ef596ae86a84f9c15163ce48 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 550cf6de024ed8e0ef596ae86a84f9c15163ce48 # timeout=10
Commit message: "Update tic.css"
 > git rev-list --no-walk b48f585c0fb58f3e92b5295a510c5f74f6bac19f # timeout=10
Finished: SUCCESS
```

trigger means job will get automated

and without this also we can do 2 more ways one is build periodically and poll scm
now let go build prerodically

cron:

it is like sedule build like cronjobs we can sedule whenever we want
cron is used automatic the jenkins build , cron is represented in stars and it has 5 stars

star-1---->MINUTES
star-2---->HOURS
star-3---->DayOfTheMonth/Date
star-4---->MonthOfTheYear
star-5---->DayOfTheWeek
sunday will take zero

1=monday
2=tuesday
3=wednesday
4=thrusday
5=friday
6=saturday

example: 12/oct 11:45 build and nov 23 12:23

cron syntax:
24 11 12 10 0
23 12 23 11 02

now every minutes

*****

here 5 stars means everyminutes,evryhour,everyday,evrymonth,everyweek
suppose evry 5 minutes wants for that use/5 *****__suppose every 5 hours wants for that use /5__*

__if you face difficulty you can learn from crontab.guru__

- *so in the genkins job configure one job and select build periodically where you have give cronsyntax
  right now iam give evryone minute *****

Build Triggers

- Trigger builds remotely (e.g., from scripts) ?
- Bitbucket webhook trigger ?
- Build after other projects are built ?
- ☑ Build periodically ?

  Schedule ?

  ```
  * * * * *
  ```

  ⚠ Do you really mean "every minute" when you say "* * * * *"? Perhaps you meant "H * * * *" to poll once per hour
  Would last have run at Saturday, October 11, 2025 at 4:24:34 PM Coordinated Universal Time; would next run at Saturday, October 11, 2025 at 4:24:34 PM Coordinated Universal Time.

- Build when a change is pushed to BitBucket

  *****

so it is completed

now move to poll scm where as schedule so here also give same syntax i.e ***** so if we give evryminute it will not do

there is one difference build peridically and poll scm both are using sedule
in build peridically it will not take any changes are done if we sedule it will deploy it will not care changes and commites
in poll scm if the changes are occur if we sedule that time it will automatic deploy and it will take care commits and messages

TOTAL 5 consepts

integrate github public and private job in jenkins
automatic build trgeers 3 ways
1.
2.Build Peridically
3.Poll scm