CONTINUOUS INTEGRATION

( JENKINS )

* ­­­­ Introduction to Continuous Integration
* Introduction to Jenkins
* Jenkins architecture
* CI / CD workflow
* Ways to install Jenkins in Linux Environment
* Install Jenkins as a standalone in Linux environment
* Install Jenkins as a service in Linux environment
* Build periodically option
* Poll scan option
* GitHub webhooks option
* Role based user management
* Day builds
* Install tomcat 9 Jenkins server
* Deploy to container plugin
* Master and slove configuration
* Intergrating Email service
* Build pipeline
* Jenkins file
* Blue ocean

**Why we use Jenkins**

* In continuous integration after a code commit the software is build and tested immediately
* In a large project with many developers commits are made many times during a day
* With each commit code during a day
* If the test is passed build is tested for deployment
* If deployment is a success the code is pushed to production
* This commit, build, test and deploy is a continuous process and hence the name continuous integration

**Introduction to Jenkins**

1. Jenkins is the most popular open source continuous Integration tool and is entirely written in java.
2. It has tons of plugins that enhance its functionality
3. Jenkins is used to integrate all Devops stages with the help of plugins
4. The reason Jenkins became so popular is that of its monitoring of repeated tasks which arise during the development of a project
5. For example if our team is developing a project Jenkins will continuously test our project builds and show us the errors in early stages of our development
6. Default location of Jenkins is /var/lib/Jenkins
7. Default port number of Jenkins is 8080

**Know we are configuring**

**Jenkins with Java and Maven build tool**

1. Know go to AWS console and login
2. Launch Instance **[ Ubuntu ]**
3. Add security group **Inbound rule 8080**
4. Instance Type is **t2. Medium**
5. Click on create Launch Instance.
6. Connect to server.

**Requirements In server**

1. **Java**
2. **maven**
3. **Jenkins**

**Above This application Installing In Ec2 instance server**

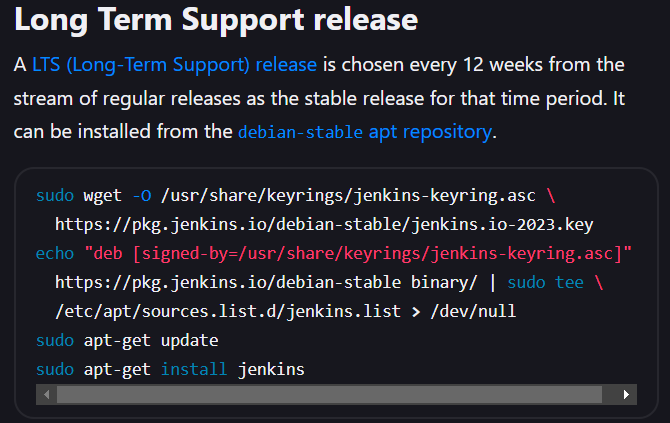
Ex:- Deploy Application

1. Train-ticket Booking ,
2. Petclinic
3. Jpetstore

**Build periodically**

**Steps 🡪 1**

* **sudo -i**
* **apt update -y**
* **apt install default-jdk -y && apt install maven -y**
* **Now go to the browser and search for Jenkins installing for LINUX**

****

* **sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \**

**https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key**

**echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \**

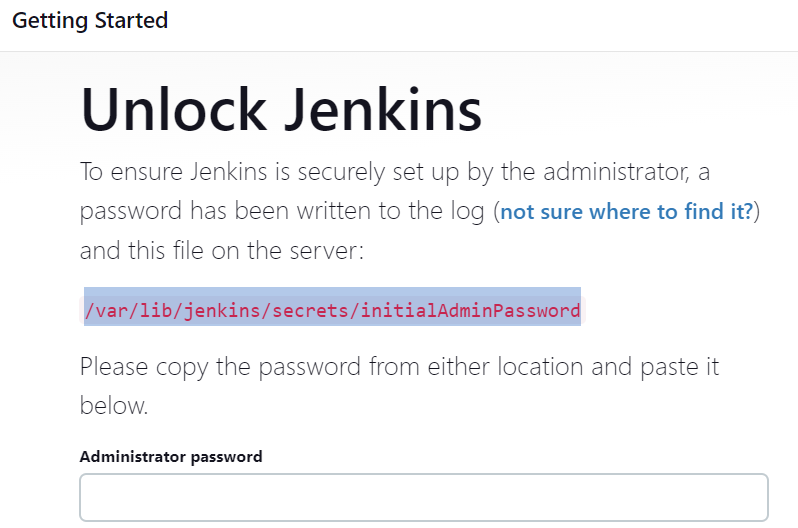
**https://pkg.jenkins.io/debian-stable binary/ | sudo tee \**

**/etc/apt/sources.list.d/jenkins.list > /dev/null**

* **sudo apt-get update**
* **sudo apt-get install Jenkins**

**Step 🡪 2**

* **Go to browser and enter your publicID:8080 Ex:- 3.90.145.162:8080**

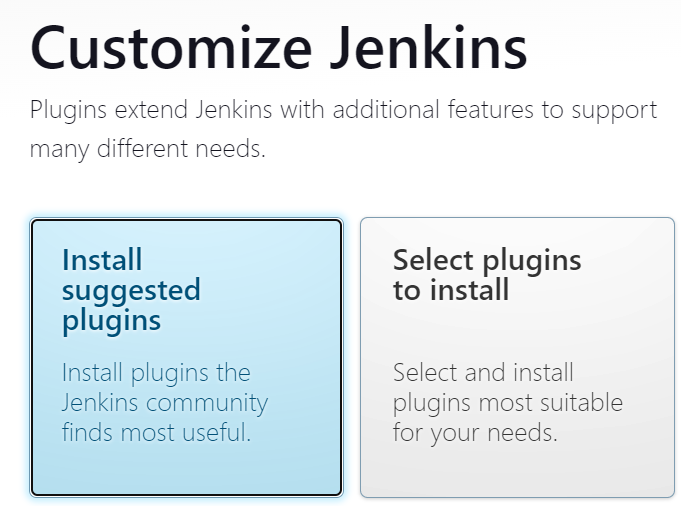
****

* **Go to GitBash & view the Admin password by (cat )**

**cat /var/lib/jenkins/secrets/initialAdminPassword**

**( Ex- password: 65c1bd42c79c407d8fb2a2\*\*\*\*\*\*\*\*\* )**

* **Go to again Browser and paste your password & click on continue**

****

* **Click on Install suggested plugins**
* **Fill the Create First Admin User Page**

**Username: \*\*\*\*\*\***

**Password: \*\*\*\*\*\***

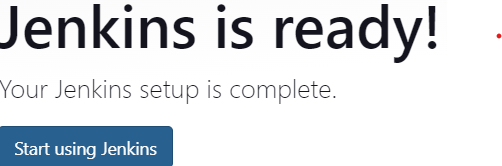
**Confirm password: \*\*\*\*\*\***

**Full name: \*\*\*\*\*\***

**E-mail address: \*\*\*\*\*\*@gmail.com**

**Know your Entering Instance Configuration page**

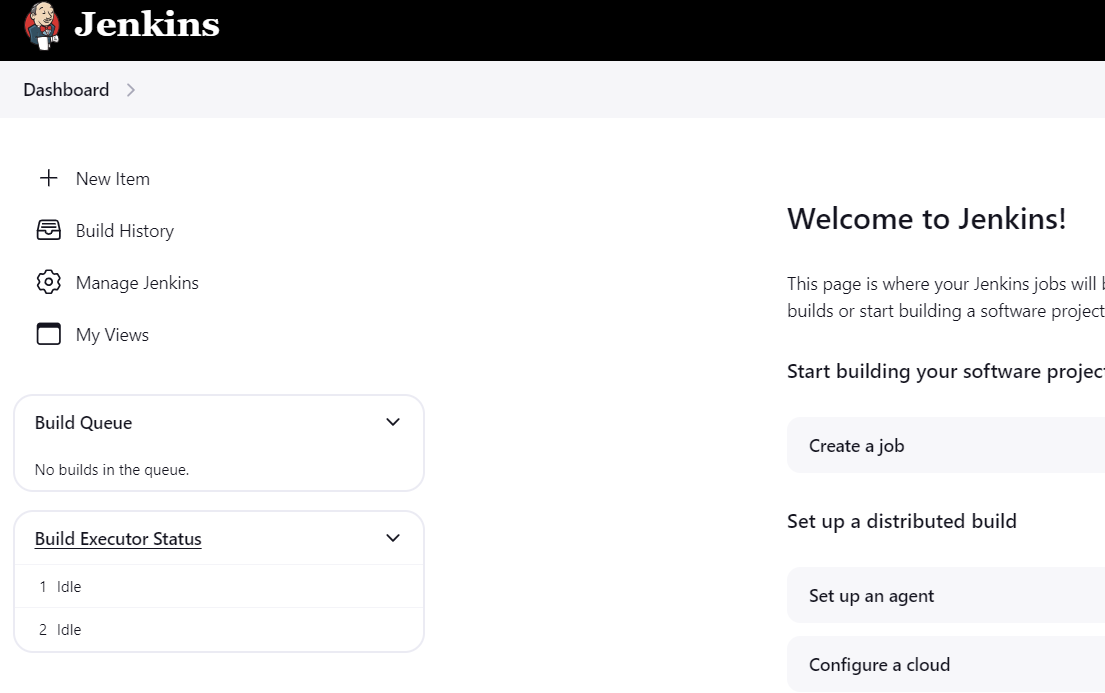
* **Click on Save and Finesh**
* **Click on start using Jenkins**

****

**OPEN YOUR MAIN JENKINS DASH BOARD**

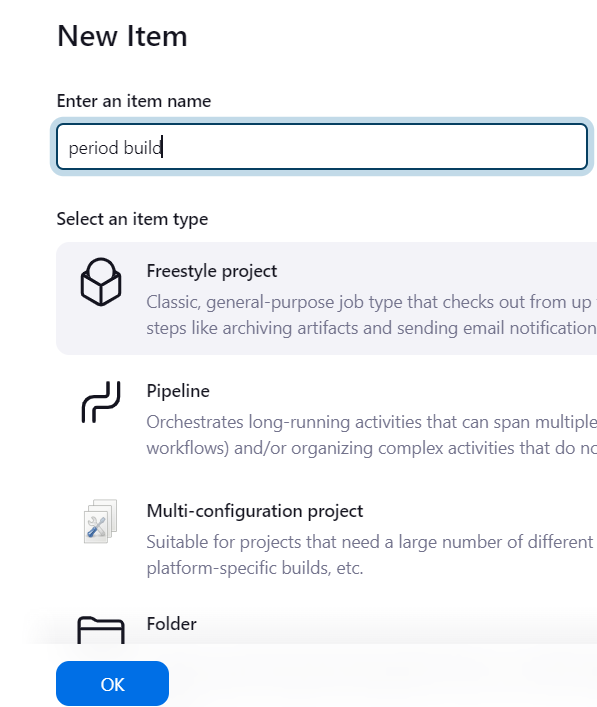
**Step 🡪 3**

* **Know your Entering Main Jenkins dash board**

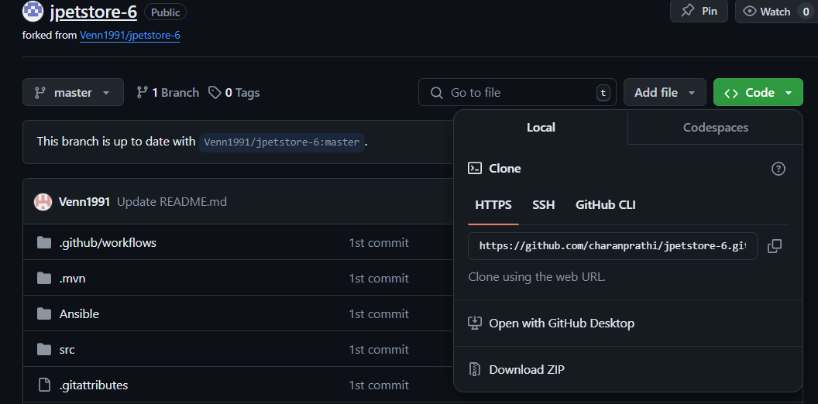
****

* **Click on New Item**

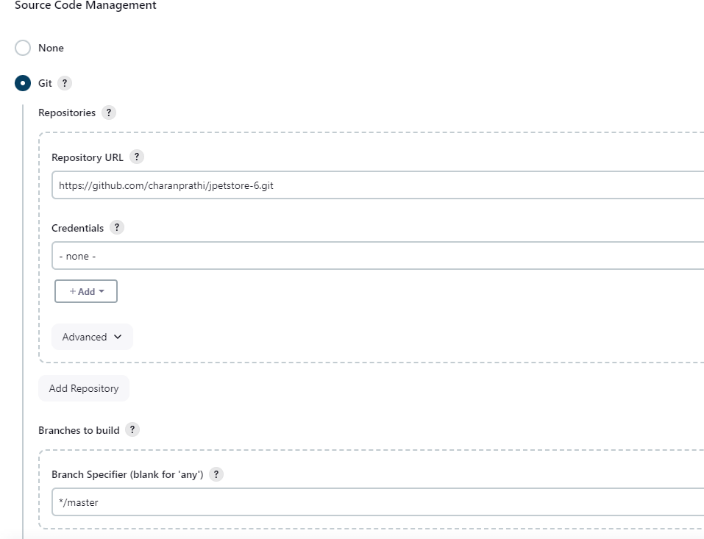
**Build periodically**

****

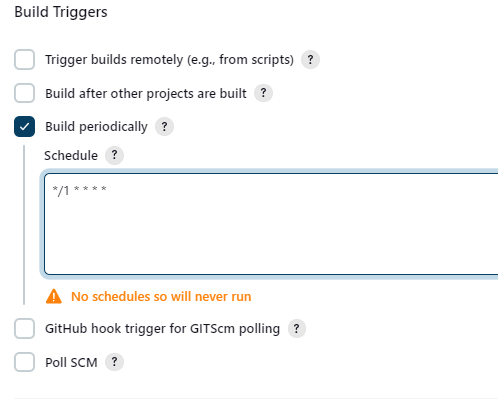
* **Create an item ( period build) using Freestyle project && click OK**
* **Now open the Git-Hub and copy the URL ( jpetstore-6 )**

****

* **Past in the Source code management block it show the branch also**

****

* **Now select Build Periodically in the Build Triggers**

****

* **The Formate of time for Build periodically**

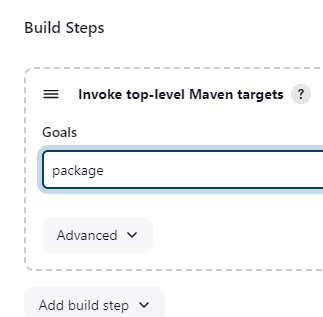
**Minute Houre DOM Month DOW**

**(0-59) (0-24) (1-30) (1-12) (0-7)**

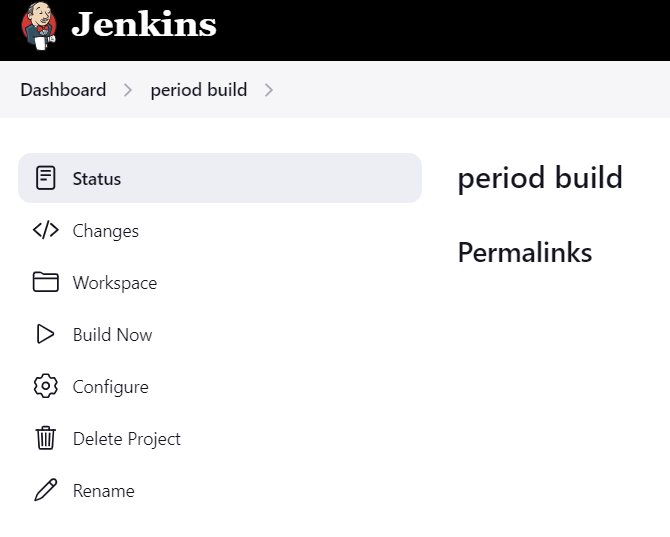
* **eg :- { \*/1 \* \* \* \* }**

**(Minutes) (Houre) (Data of Months) (Month) (date of week)**

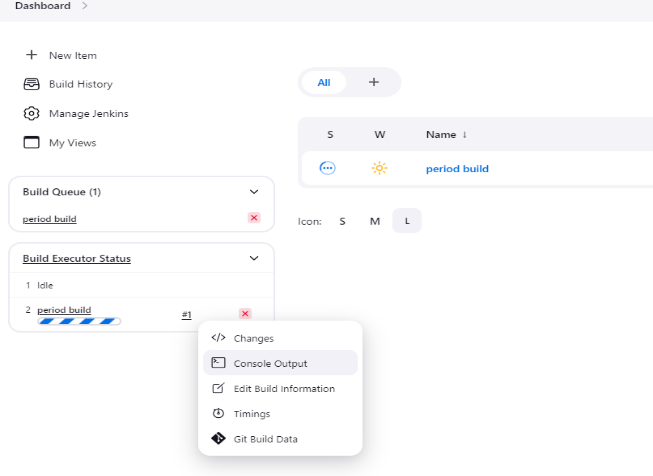
* **Next add the Build step 🡪 select the Invoke top-level maven target 🡪 goals 🡪 package**

****

* **And Apply & save**
* **Click on Build Now**

****

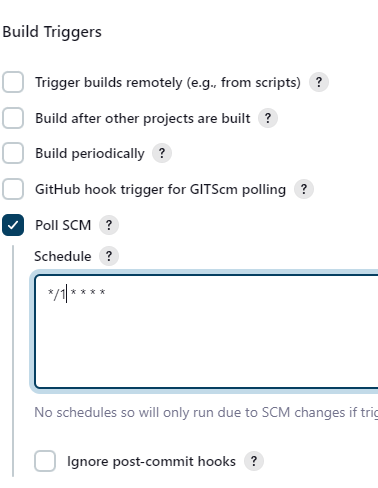
* **Next go to the Dashboard and click on Build Executor Status go the Console Output and check the Deploying of the application**

****

* **Go to console to check the code is success**
* **The Build periodically is a continues deploying it deploys for every 1 min which we have given time in Build Tiggers**

**PollSCM**

* **Create a New Item (Poll SCM)**
* **With the Freestyle project & click on ok**
* **Now go to the source code management & past the URL ( jpetstore-6 ) in the Repository URL**
* **Next in Build triggers 🡪 select Poll SCM**

****

* **The Formate of time for Poll SCM**
* **The Formate of time for Build periodically**

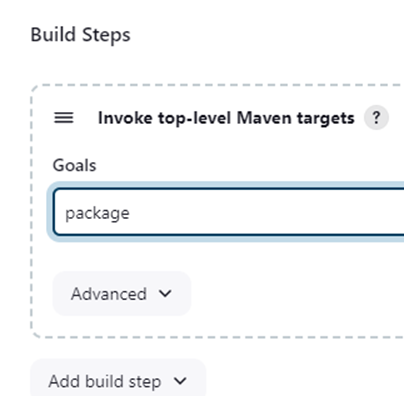
**Minute Houre DOM Month DOW**

**(0-59) (0-24) (1-30) (1-12) (0-7)**

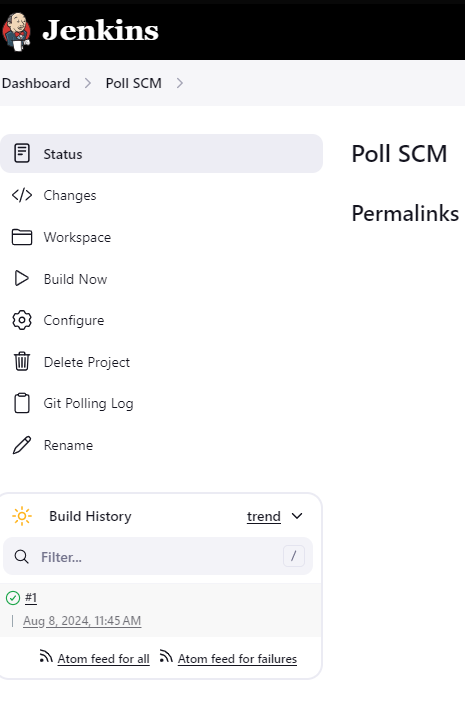
* **eg :- { \*/1 \* \* \* \* }**

**(Minutes) (Houre) (Data of Months) (Month) (date of week)**

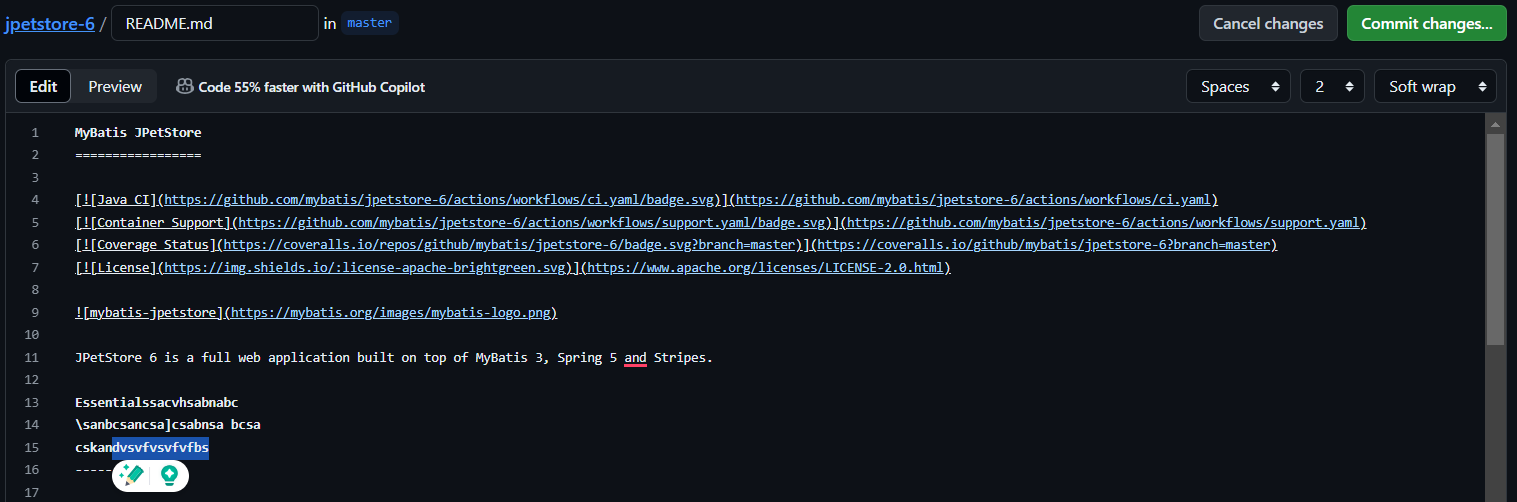
* **Next click on Build Steps 🡪 ( Invoke top-level maven targets ) 🡪 gaols (package)**

****

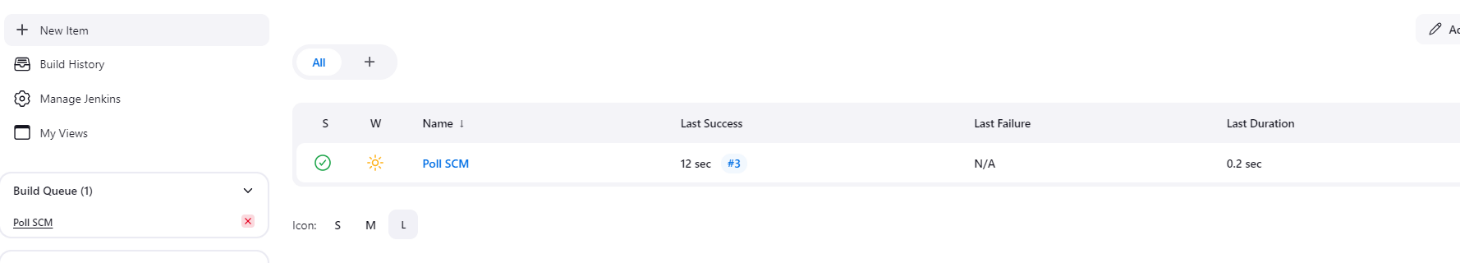
* **Now click on Apply & save**
* **Click on build now and it starts developing automatically**

****

* **And next go to Git-Hub ( jpetstore-6 ) – repository and change the (readme.md) file that means edit a data**

****

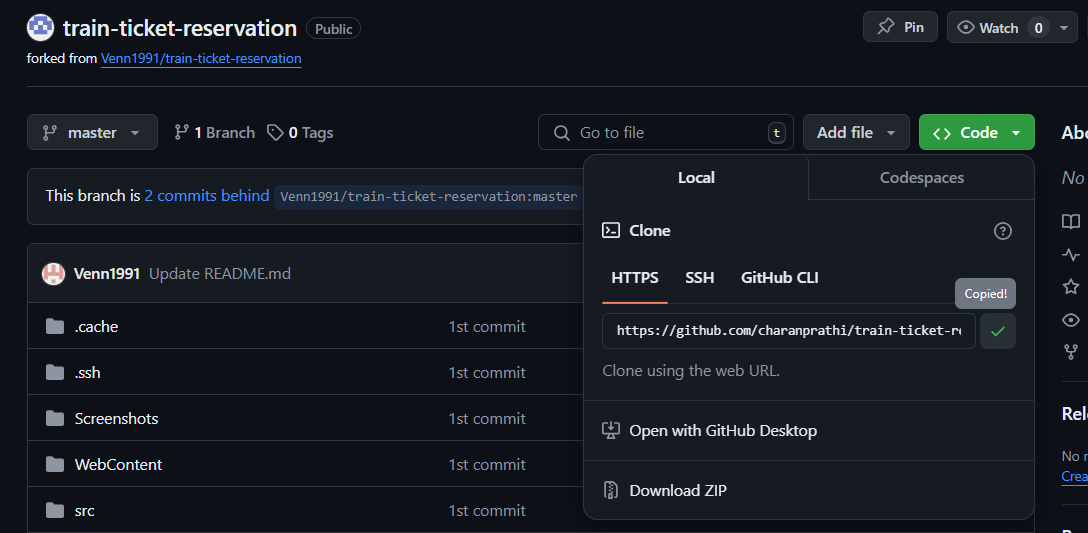
* **Commit changes**
* **Now go back Jenkins dashboard and check the deploying starts within the next minute**

****

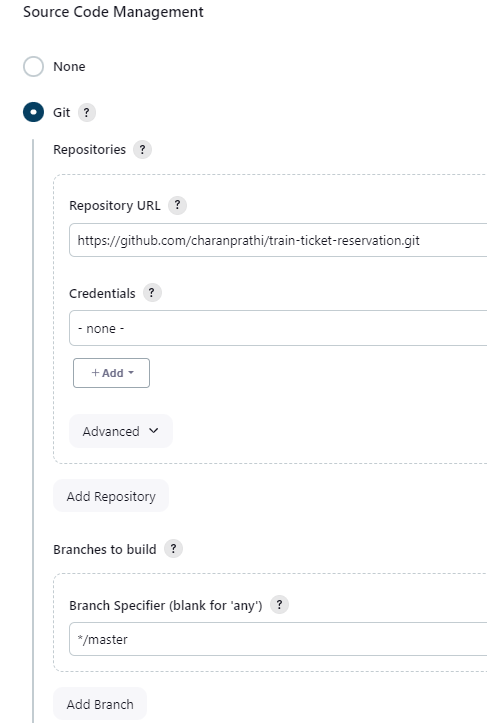
* **Here the application start Queue sharply in the next minute of your changes are applied in the Git-hub repository**
* **Mainly this is because Poll SCM means it automatically checks the code when you have commited changes in the Git-hub repository in the given interval of time**

**Webhook**

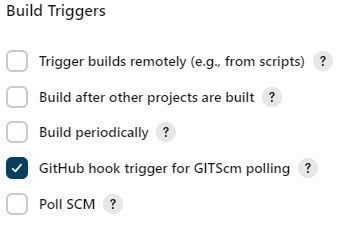
* **Create a New Item (Web hook)**
* **With Freestyle project & click on OK**
* **Open Git-hub and copy the URL (Train-trick-reservation)**

****

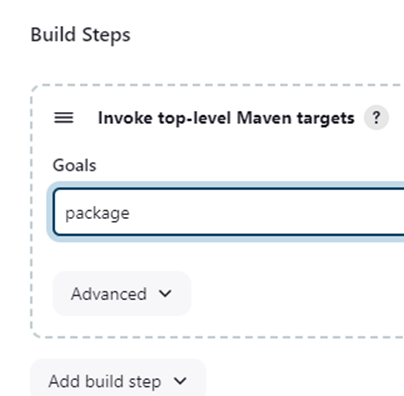
* **Now go to the source code management and past the URL**

****

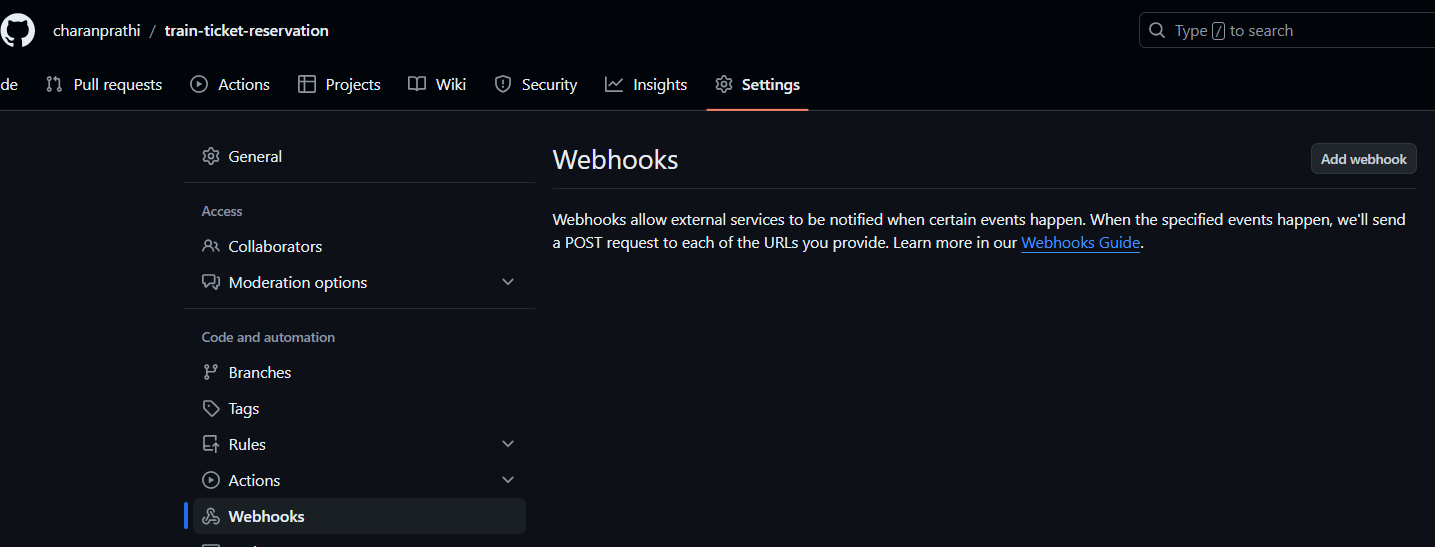
* **And select GitHub hook trigger for GITscm polling in the build triggers**

****

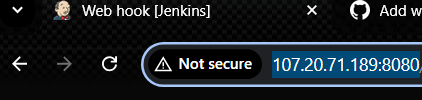
* **Next add the Build step 🡪 select the( Invoke top-level maven targets) 🡪 goals( package)**

****

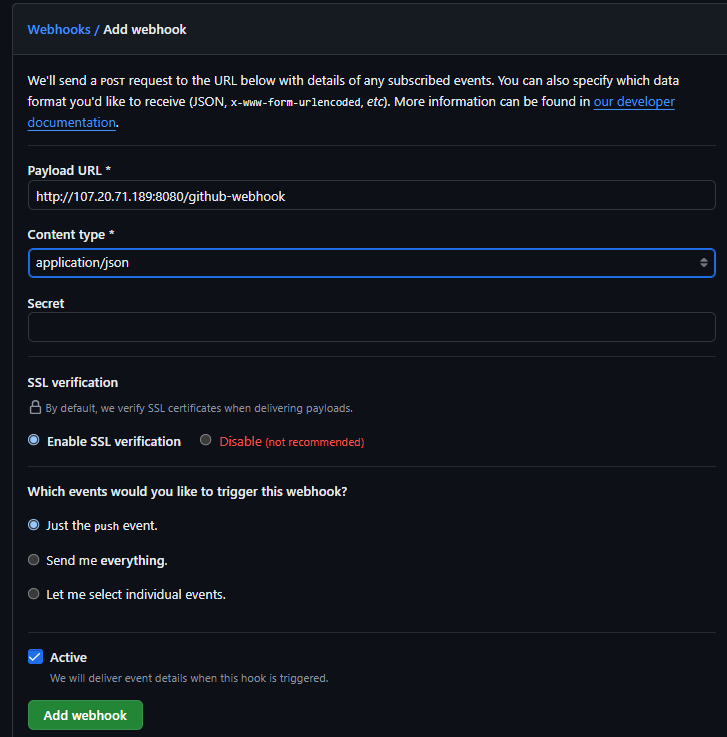
* **Then Apply & save**
* **Go to Git-Hub (Train-ticket-reservation) 🡪 settings 🡪 click on Web hook**

****

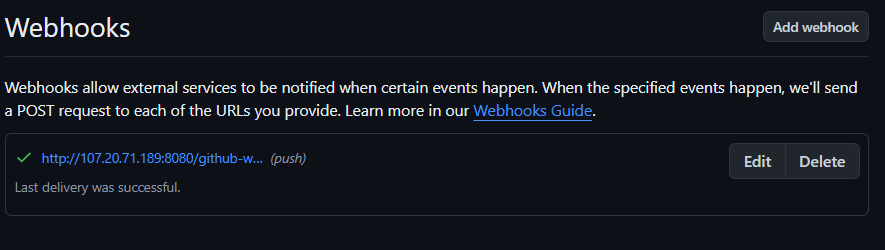
* **Click on add Webhook & use password to confirm**
* **And go to the Jenkins page copy the URL till 8080**

****

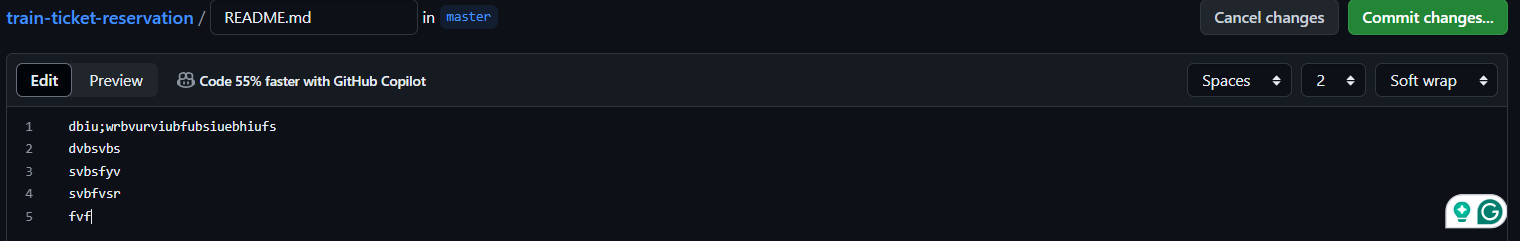
* **Next go to the Git-Hub 🡪 setting 🡪 webhook & past in the Payload URL and type github-webhook & the content type (application json)**

****

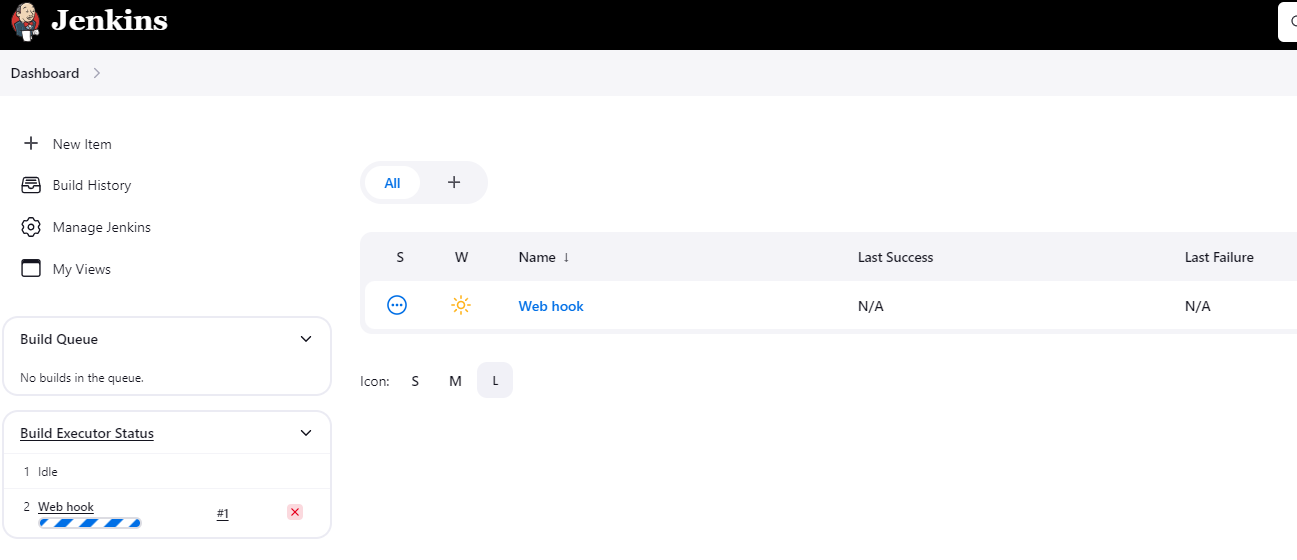
* **Click on add webhook repeat the process until you get the Tick mark on the webhook**

****

* **Now change the (Readme.md) file in the repository & commit changes**

****

* **Next go and check the Jenkins dashboard immediately after commiting changes the wedhook starts deploying**

****

* **This is because webhook is immediate reaction when ever you change the code of repository which is (web-hook) to the Jenkins it spontaneously start deploying right after you commit changes in the Git-hub repository without any delay of time intervals**