**How to use CLI with Jenkins**

1. Goto Manage Jenkins 🡪 Configure Global Security 🡪 Enable security should be enabled

2. Goto http://localhost:8080/cli/

3. Download Jenkins-cli.jar in any location

4. Test Jenkins CLI. Run this command in terminal

$ java -jar [jenkins-cli.jar](http://localhost:8080/jnlpJars/jenkins-cli.jar) -s http://localhost:8080/ help --username {} --password {}

Issue: Run into ERROR: anonymous is missing the Overall/Read permission

Solution: Goto Configure Global security 🡪 Authorization 🡪 enable anyone can do anything

Restart your command

**How to create Users + Manage + Assign roles**

1. Create new roles

Goto Manage Jenkins 🡪 Manage users 🡪 Create user

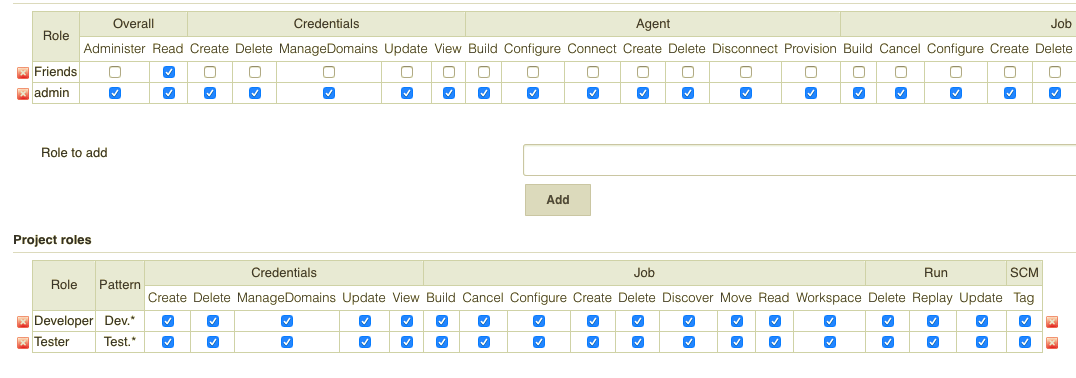
2. Managing the roles

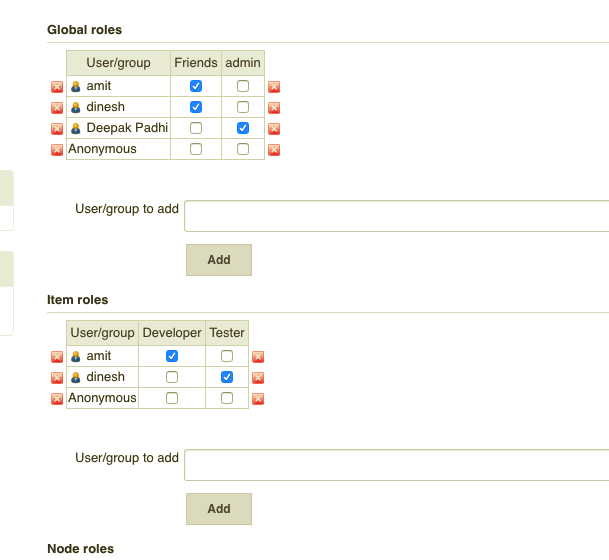
Install Role-Based Authorization strategy from Manage plugin and then restart

Login as Admin 🡪 Goto Configure Global security 🡪 Authorization 🡪 enable Role-Based strategy

Go to Manage roles 🡪 create global roles eg: employee role and project roles eg: developer

Go to assign roles 🡪 select user and role like employee and/or developer

****



**Basic Configurations**

1. Manage Jenkins 🡪 Configure System

2. Understanding of Basic Configuration

System Message: message provided will be displayed on the dashboard

# of executors: No of parallel jobs a Jenkins instance can execute

Label: To specify on which node a job should run

Quiet period: Waiting period before Jenkins will start job execution

SCM checkout retry count: 🡪 under build triggers >> Poll SCM- It will connect to repository and get the latest code. If it fails, then what does the maximum retry count is given in scm checkout.

**How to run a job**

*First create a job by clicking on new item and then giving name and selecting freestyle*

**Periodically**

1. Click on project e.g.- test1. Click on configure 🡪 Build Triggers 🡪 enable build periodically

Format: MINUTE HOUR DOM MONTH DOW

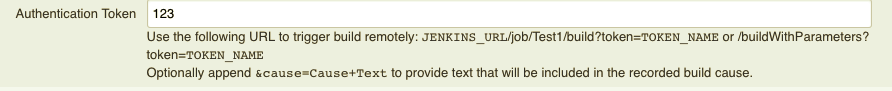
|  |  |
| --- | --- |
| MINUTE | Minutes within the hour (0–59) |
| HOUR | The hour of the day (0–23) |
| DOM | The day of the month (1–31) |
| MONTH | The month (1–12) |
| DOW | The day of the week (0–7) where 0 and 7 are Sunday. |

|  |  |
| --- | --- |
| **Remotely**  1. Click on project e.g.- test1. Click on configure 🡪 Build Triggers |  |

Under Build Triggers 🡪 Trigger builds remotely

Authentication code: we need to give, and it can be anything which will act as password

URL: localhost:8080/job/Test1/build?token=123 {according to picture}



**Chain Jobs**

Let us take 3 different jobs: test1, test2, test3

Now the test2 job should build or execute after test1 and after test2 gets executed successfully test3 should execute. How is this done?

Sol: Goto test2 configure 🡪Build Triggers 🡪 Build after other projects are built

Provide test1 as projects to watch and enable Trigger only if build is stable

*Now test2 will execute after test1 gets executed successfully*

Under Post-build action 🡪 test3 as project to build

This will execute as test3 after test2 execution

**Jenkins Integration with GIT (SCM)**

**Without GitHub**

1. Create a job 🡪 configure 🡪 Build (execute shell) 🡪 provide the path and command 🡪 save

2. Build now 🡪 check output

Github

Commands:

🡪 git init 🡪 git status 🡪 git add {file} 🡪 git commit [optional message (-m “comment”)]

🡪 git push -u origin master

**CATLIGHT**

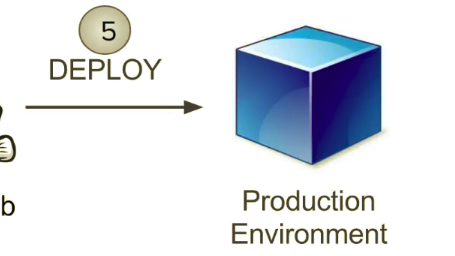
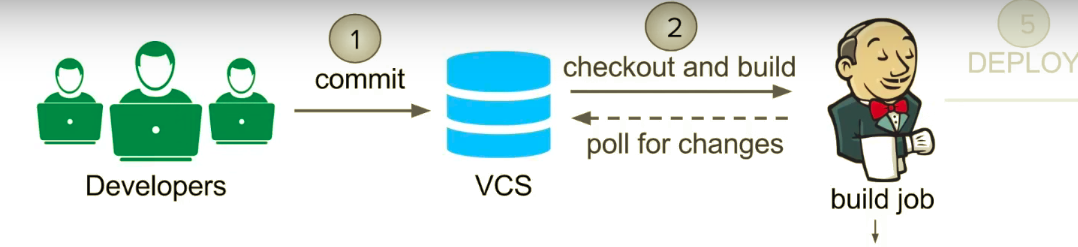
Catlight is used to monitor jobs on Jenkins and whenever there any status change or need attention it will pop up and notify you

**Automated Deployment - A part of continuous delivery**

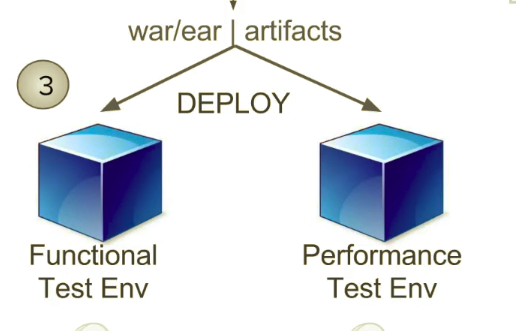
**It is a process of automating the deployment process in continuous delivery system.**

**BUILD 🡪 DEPLOY 🡪 TEST 🡪 RELEASE**

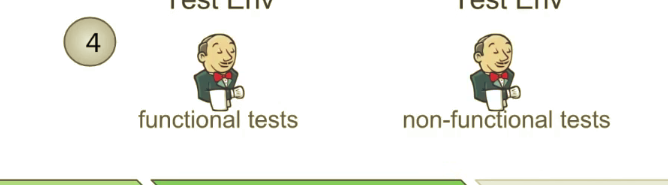
**BUILD Release**

****

**Deploy**

****

**Test**

****