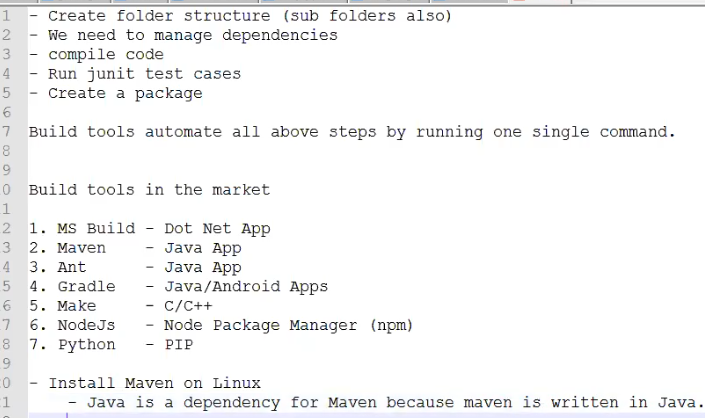
**Video 5:**

How to build a project

There is java based app, it have lot of files , our job is we need to build and deploy, most of java based apps they use maven build tool.

Some one ask me to build

* Open git bash – $ git clone paste url – $ cd my-app/
* To get latest code: git pull
* $mvn package: reads the source code pom.xml (config file used by maven ) and it creates a package
* War file is created – deploy this war file on a server and give it to testing- give it to UAT – staging – finally move it to production





Launch Ec2

git clone paste url

cd my-app

mvn package

maven uses pom.xml which is configuration file to build a project

in order to build java project using maven your java project must follow maven directly structure. Otherwise you can’t build.

Developers will take care of creating Maven based java projects.

**Maven dependencies:**

Dependency is an external jar file that our project depends on.

The dependency we want to use in the project is configured in pom.xml (Copy the dependency and paste it in pom.xml)

Maven automatically downloads dependency from repositories

**From where depencencies are downloaded :**

Maven uses diff repositories

* Central repository
* Local repository
* Remote repository

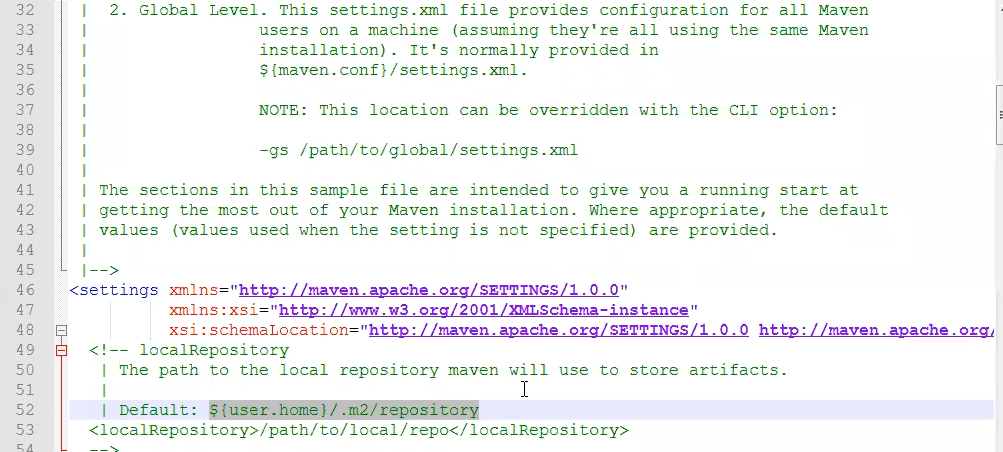
**Central repository** : it’s a repository maintained by maven itself over internet (server is maintained by maven, who upload dependencies in to that - organisation maintains dependencies, ex :apache maintains it.)

**Local repository** : it is a repository on local machine i.e the machine where we are running maven build commands.

**IQ): What is maven local repo path? or what is default location of aven repository**

C:/users/beerelokesh/.m2/repository (C:\Users\Beere Lokesh\.m2\repository)

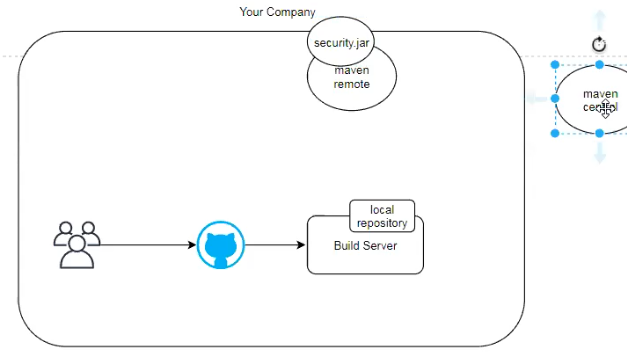
If we want to chang path change path there (( C:\apache-maven-3.6.2\conf -seeting.xml page)



**Remote repository**:

Remoterepository is a server which is maintained within company N/W.

* Security
* Company might have its own framework which is internal to company those dependencies can be managed by remote repository



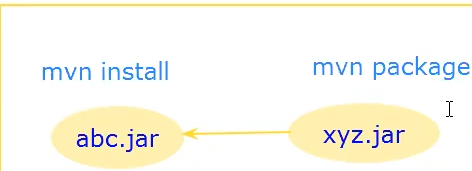
Inorder to use remote repository we use

* sonar type nexus (we have pen sorce and enterprise version)
* jfrog artifactory

**IQ) what is maven buid life cycle: imp ( VCTPVID)**

* Validate: it validates pom .xml
* Compile: it compiles source code (to compile source code it requires dependencies)
* Test: Executes **Junit** test cases. (
* Package: it is going to package your software by creating jar or war or ear
* Verify: To verify integration test results
* Install: copies package into local repository

(We have 2 projects xyz proj depends on abc proj (both are my projects) if we want to build xyz proj we need abc, I run mvn install here , it validates,compile, test ,package and copy that package to local repository then I will use mvn package. In my pom.xml of xyz I say group id, artifact id, dependency, version all that , it picks the jar file from local file so when you have dependency modules or proj install that such that your jar file dependency will be available in local repo that is used by other project then only we use install)



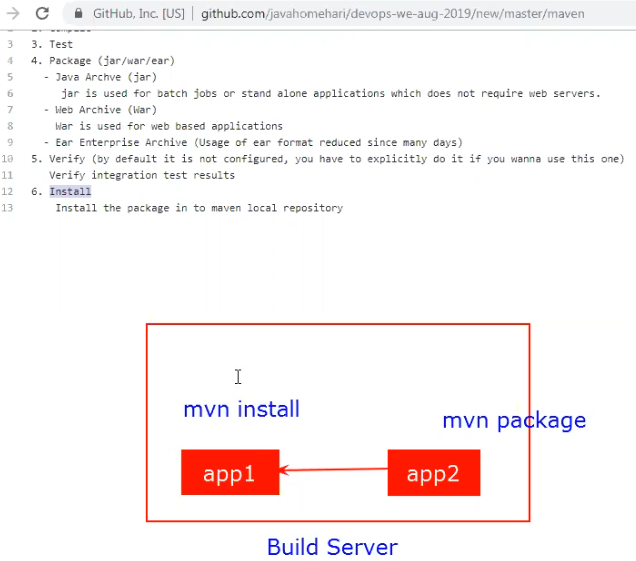
* Deploy**:**  it uploads artifacts to remote repository

If we have 2 projects, if 2 projects are inside same machine, you are building them together and depend on each other and we use install and we keep it on local

Let say I am working on machine 1 and you are working on machine 2 then you depend on my dependency, then requires we keep it in remote. I will deploy my packge to remote. Then maven downloads package from remote.

**Video 6 oct 8**

**Github.com/javahomehari/devops-we-aug-2019/new/master/maven**





**Deploy : if you want to run deploy we must have remote repository (ex :nexus ) installed and configured and we must tell maven that location of that remote repo)**

**----------------------------------------------------------------------------------------------------------------------------**

1. Validate
2. Compile
3. Test
4. Package (jar/war/ear)
   * Java Archve (jar) jar is used for batch jobs or stand alone applications which does not require web servers.
   * Web Archive (War) War is used for web based applications
   * Ear Enterprise Archive (Usage of ear format reduced since many days)
5. Verify (by default it is not configured, you have to explicitly do it if you wanna use this one) Verify integration test results
6. Install Install the package in to maven local repository
7. Deploy Uploads you package to remote repository(Sonatype Nexus/ Jfrog Artifactory)

-------------------------------------------------------------------------------------

**Maven SNAPSHOT and RELEASE version**

SNAPSHOT and RELEASE is maven conventions

**RELEASE Version**

According to maven convention dependency with RELEASE version will not have new changes, maven downloads it from central/remote ones and depends on local copy next time onwards. if version is not ending with -SNAPSHOT then it is release

**SNAPSHOT Version**

According to maven convention dependency with SNAPSHOT version will have new changes, maven downloads it from central/remote every time you build your application.

if version is ending with -SNAPSHOT then it is SNAPSHOT

----------------

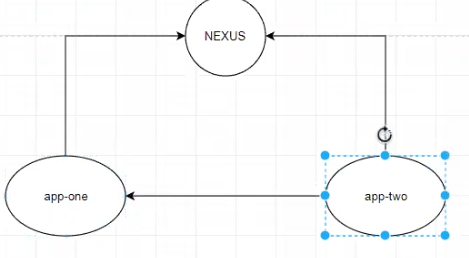
when they do mvn deploy (app one ) a new snapshot is uploaded to Nexus and when they (i.e app-two) run mvn package that depency is coming from Nexus

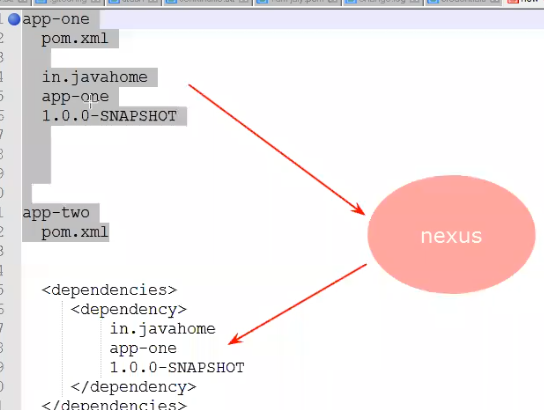
once development is done they change snapshot as a release)

in dependency also they remove – snapshot which starts pointing to a release

if we have 10 snapshotos it always picks the latest snapshot based on time stamp

accordingly who ever is developing they update pom.xml





**Maven clean package :** It deletes the target folder

Maven organises all files for creating a package in target directory. target is called as maven o/p folder it is safe to delete we can be recreated using maven goals (ex:install, package,deploy etc).

How to run junit test cases?

ANS): It executes by default