Jenkins

DevOps Tools:

➔ **Log Monitoring Tools:**

- •Splunk
- •SumoLogic
- •LogStash
- •Kibana
- •PaperTrail
- •Loggly

➔ **System Monitoring Tools:**

- •Graphite
- •Gangalium
- •Sensu

➔ **Network Monitoring Tools:**

- •Zabbix
- •Nagios
- •Cacti

➔ **Build & Test Tools:**

- •Jenkins
- •Gradle
- •Maven
- •Ant & Lavy
- •Cruise Control

➔ **Deployment and Configuration Tools:**

- •Puppet
- •chef
- •Rancid
- •CFEngine
- •Ansible

➔ **Version control:**

- •Git

🛈 Jenkins

- Start learning Jenkins from *https://www.tutorialspoint.com/jenkins/index.htm*

Jenkins

- Jenkins is a powerful free source and cross-platform application that allows **continuous integration and continuous delivery** of projects, regardless of the platform you are working on.
- It can be integrated with a number of testing and deployment technologies.
- It is a **server-based** system running in a servlet container such as Apache Tomcat.

➢ What Jenkins does?

The primary job of the Jenkins is to execute a list of steps configured which is based on a trigger.

Below are the list of steps/tasks performed by Jenkins when triggered :

1. Perform the code compilation & build the software with ANT, Maven or Gradle.

2. Run an internal shell script

3. Archive the resultant builds

4. At last start the execution of the integration tests.

5. Monitoring the execution of the above tasks.

6. Provision to stop the build in case of failure in any of the steps.

7. Notifying the user about the success or failure of each build.

➢ What Jenkins focus on?

Jenkins mainly focuses on 2 important activities :

1. **Continuous build of the software :** Nothing but the creating a continuous integration environment

2. **Monitoring execution of external jobs :** This monitors the jobs defined – CRON jobs.

➢ Features of Jenkins

1. Easy to install & configure.

2. Permanent link – readable URLs for the build details.

3. Email integration for notifications.

4. JUnit tests reporting.

5. Tagging support for every successful builds.

6. Distributed builds.

7. File fingerprinting for managing dependencies

8. Plugin support

Jenkins

How to Configure Jenkins:

System Requirements:

JDK	JDK 1.5 or above
Memory	2 GB RAM (recommended)
Disk Space	No minimum requirement. Note that since all builds will be stored on the Jenkins machines, it has to be ensured that sufficient disk space is available for build storage.
Operating System Version	Jenkins can be deinstalled Sun Windows, Ubuntu/Debian, Red Hat/Fedora/CentOS, Mac OS X, openSUSE, FReeBSD, OpenBSD, Gentoo.
Java Container	The WAR file can be run in any container that supports Servlet 2.4/JSP 2.0 or later. (An example is Tomcat 5).

The following prerequisites must be met for Jenkins setup :-

 ✔ Verifying Java Installation

 ✔ Set the JAVA_HOME environment variable

 ✔ Jenkins and Tomcat Setup

 ✔ Jenkins – Maven Setup

Installation process:

Step 1: Verifying Java Installation

• Open the cmd and execute the following java command.

java -version

• If Java has been installed properly on your system, then you should get one of the following outputs, depending on the platform you are working on.

Java version "1.7.0_60"

Java (TM) SE Run Time Environment (build 1.7.0_60-b19)

Java Hotspot (TM) 64-bit Server VM (build 24.60-b09, mixed mode)

- If you dont have it then download jdk from:

  http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

- Click on Jdk file according to your system configuration . The downloading will start.
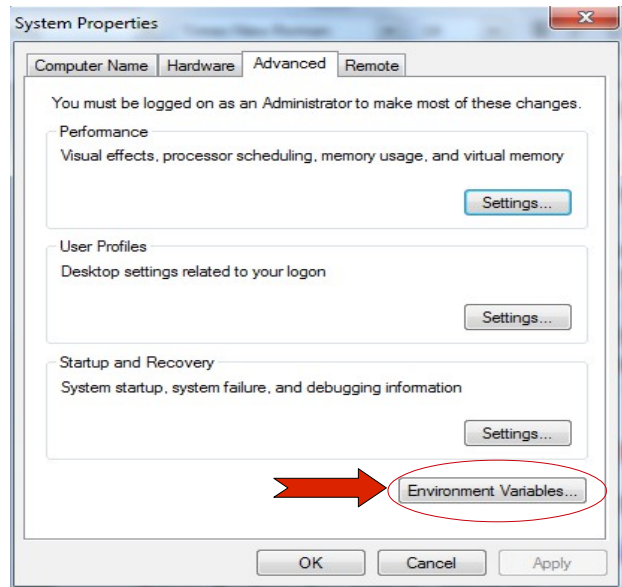
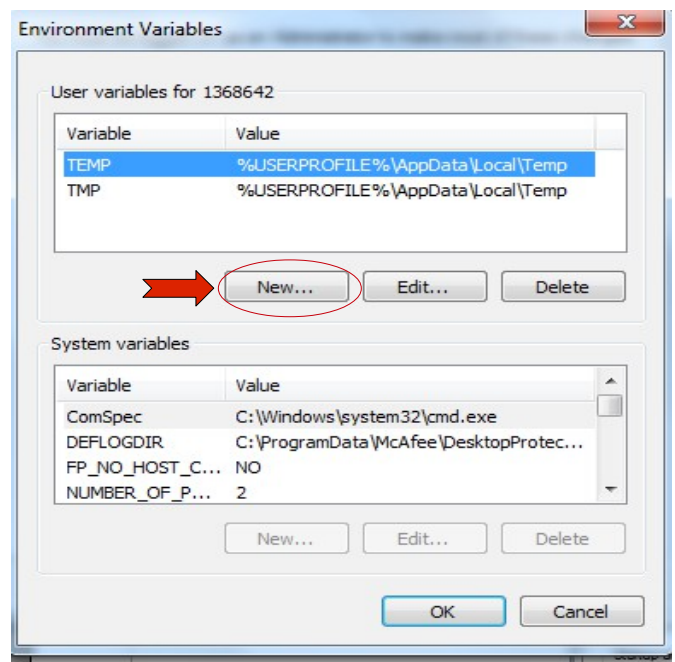- Install downloaded *.exe fle* at your desired location.

Jenkins

Step 2: Set the JAVA_HOME environment variable :

1. Go to : Control Panel\All Control Panel Items\System\ advance system setting
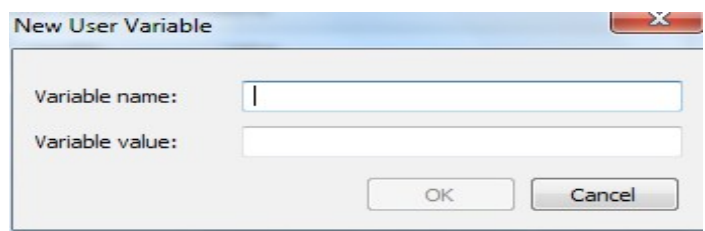
2. Click on environment variable

3. Click on new in user variables.

4. Set:

variable name: JAVA_HOME

variable value: your jdk location e.g. C:\ProgramFiles\java\jdk1.7.0_60

Jenkins

Step 3: Download and extracting Jenkins :

📦 Download :

    I) Open jenkins official website *https://jenkins.io/index.html*

    II) Click on *Download Jenkins*
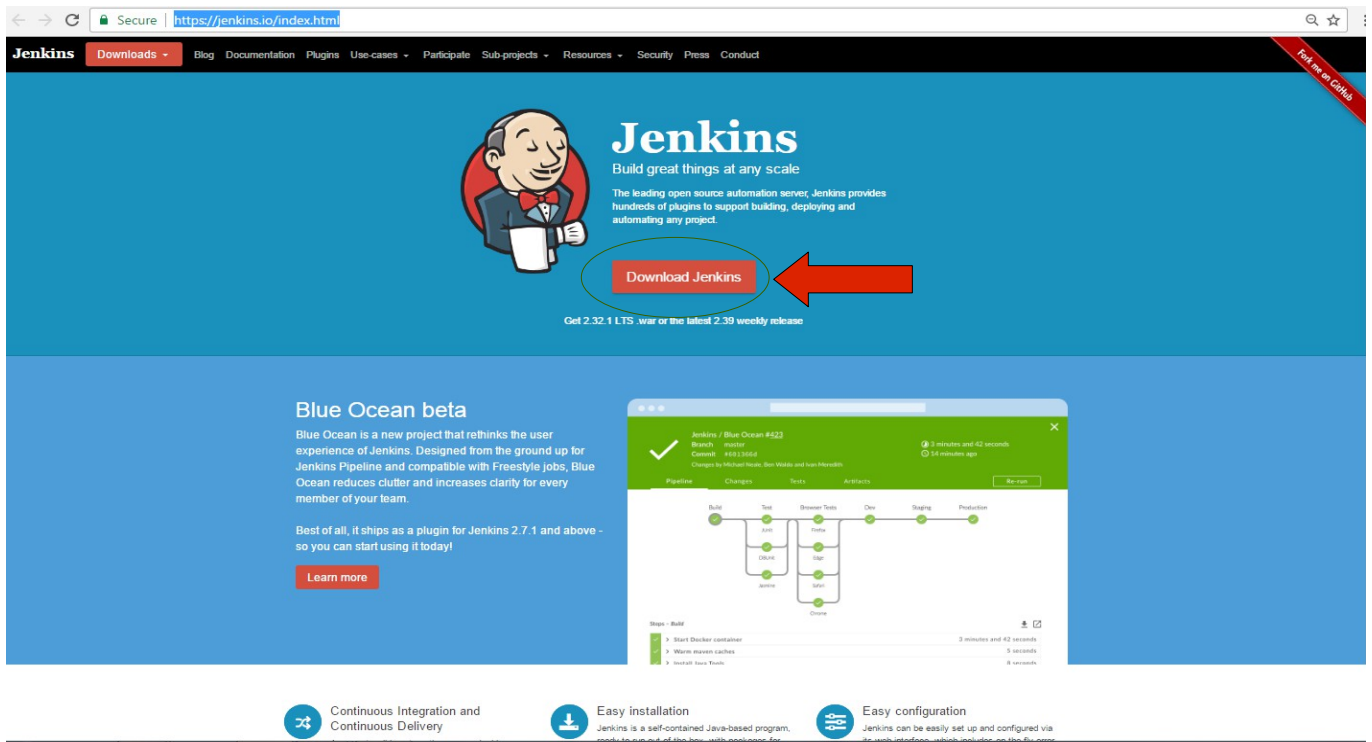
    III) Click on *LTS Release war file*
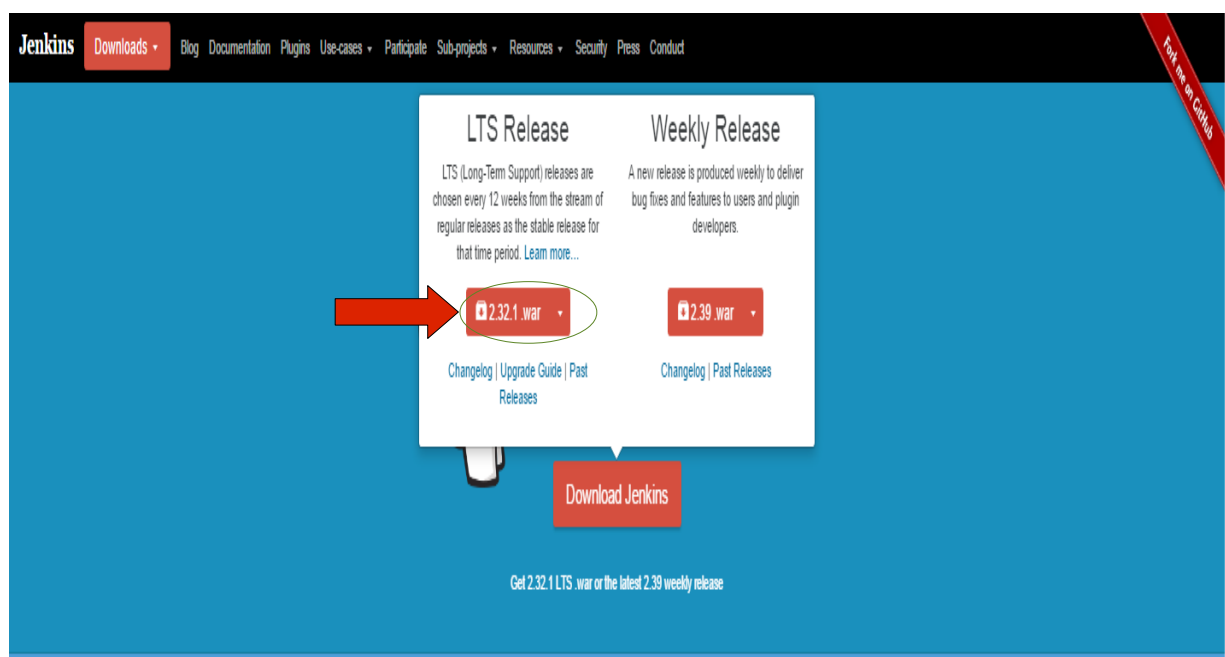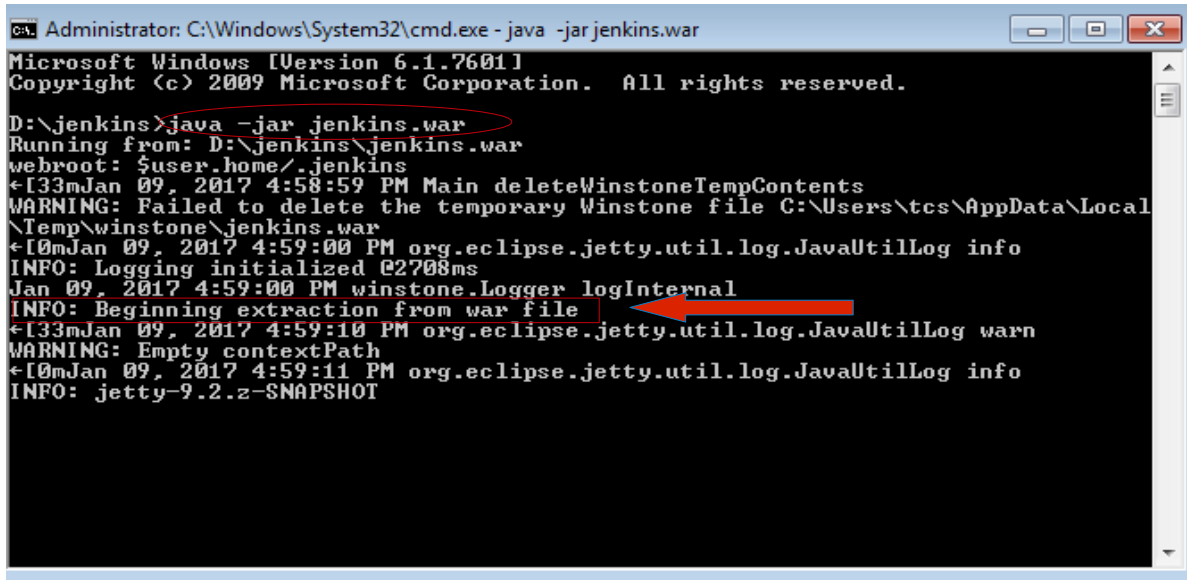
Jenkins

🔲 Starting Jenkins :

From the command prompt, browse to the directory where the jenkins.war file is present and Run the following command:

D:\>Java –jar Jenkins.war

Note down the password for installation genrated during extration process of jenkins.war file.

Jenkins

🎁 Accessing Jenkins :

Once Jenkins is up and running, one can access Jenkins from the link— *http://localhost:8080.*
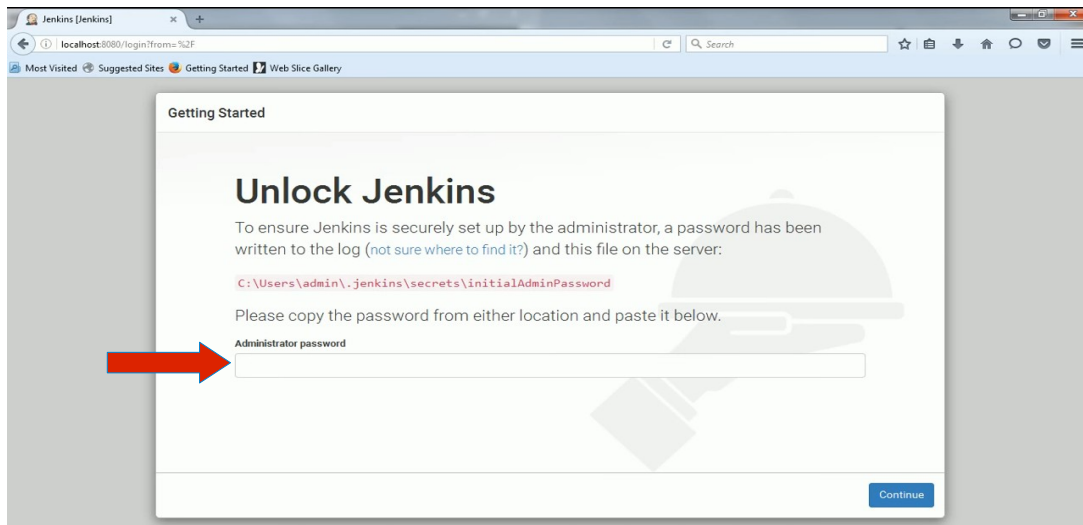
- To *Unlock Jenkins*, Enter the *password* noted during *extraction* of *jenkins.war file* and click on *continue.*
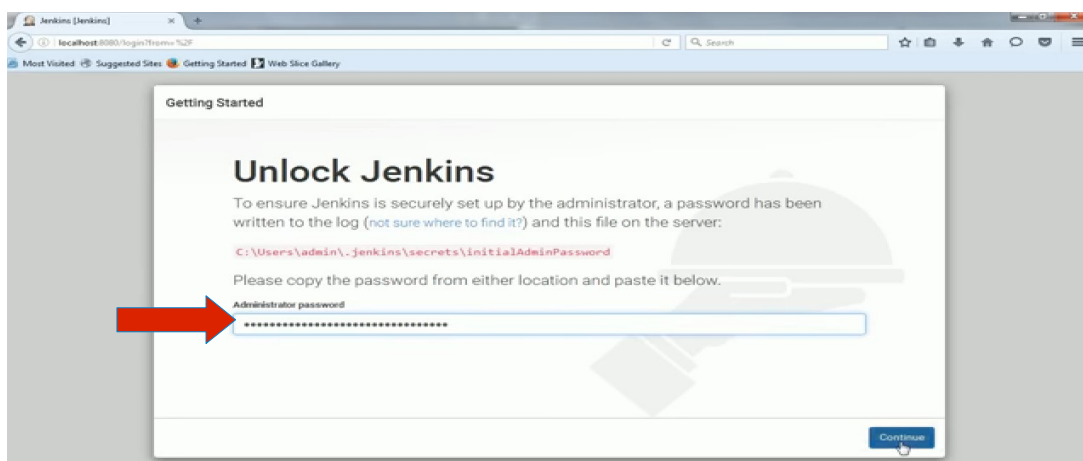
Click on *Install suggested plugins* options
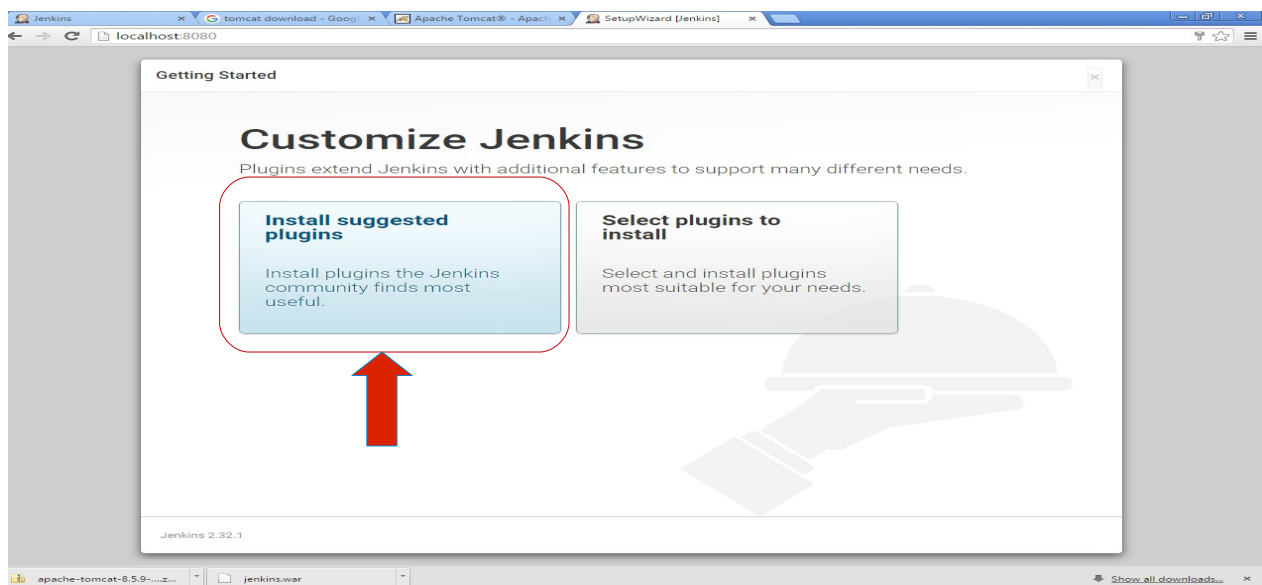
- Wait for installtion process done in dashboard as well as in background(cmd where jenkins.jar file is extracted)
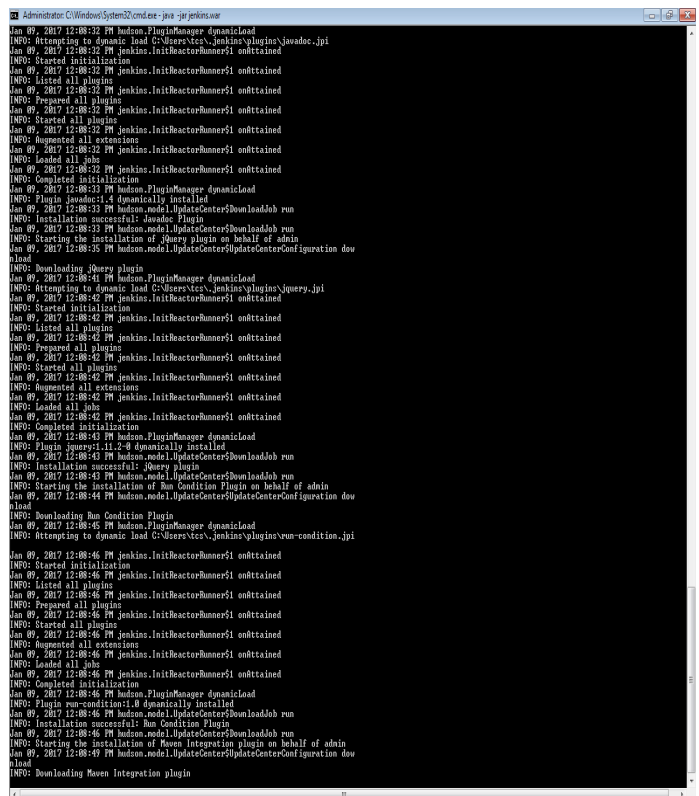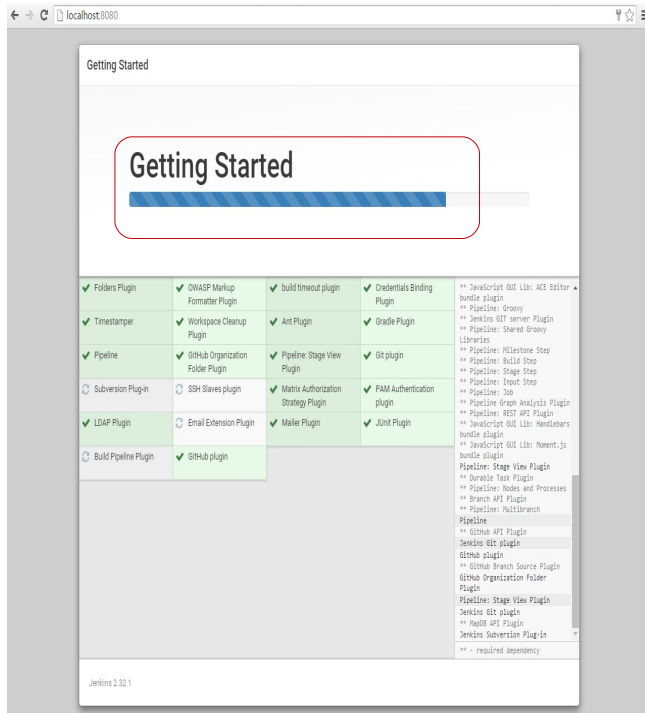
- **When the installation is completed :**

    Create the *First ADMIN User* by providing following details and click on *save and finish.*:

Jenkins

- Click on **Start using Jenkins**
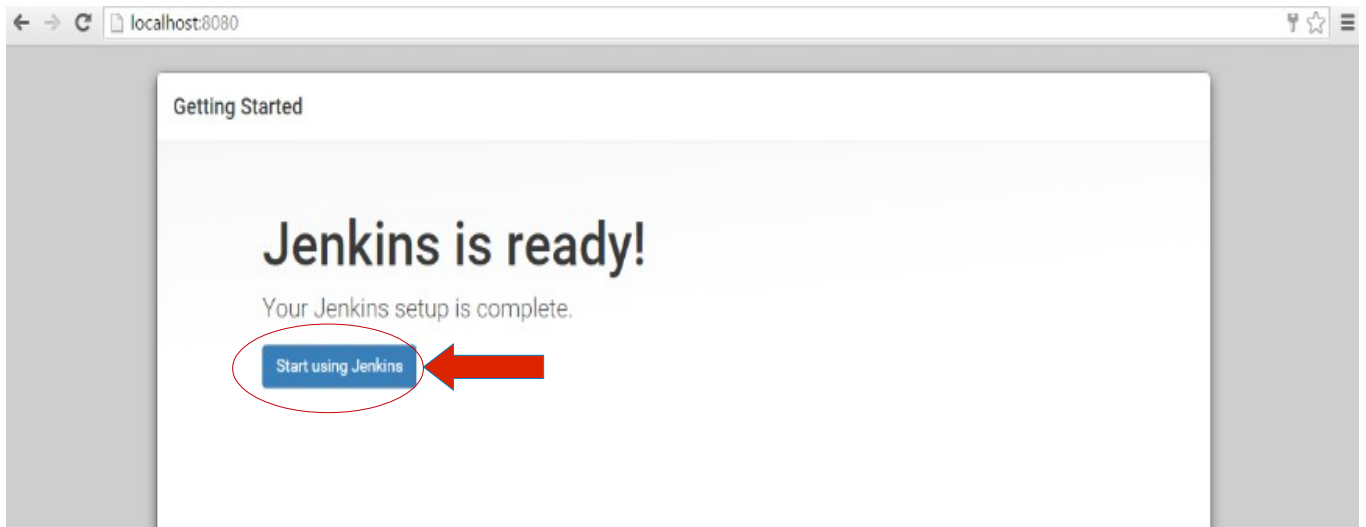
- Now Jenkin's **dashboard screen** *will appear as:*
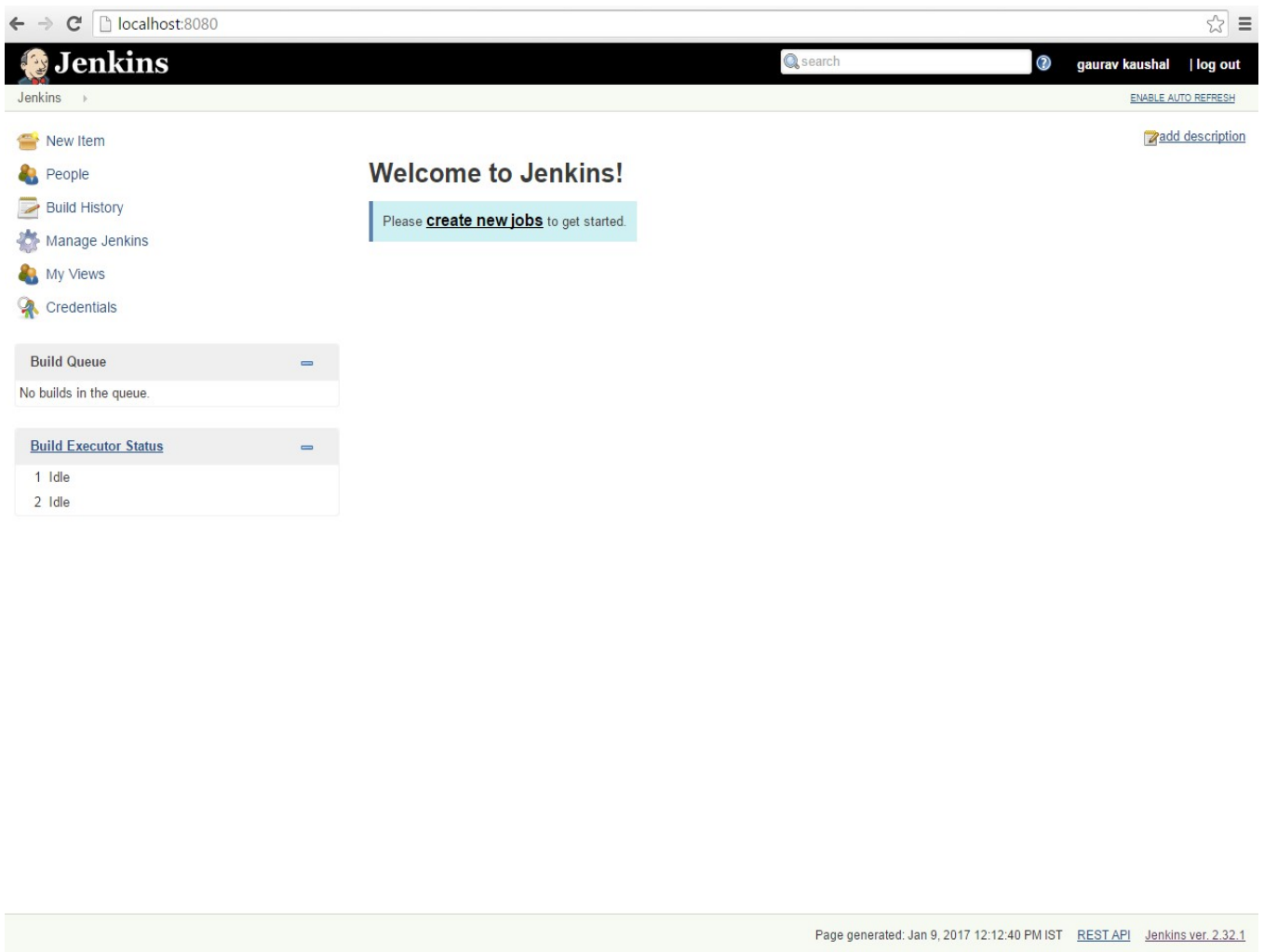
Jenkins

- **Now set the Jenkins as Windows Service**

  Go to *Jenkin's Dashboard >> Manage Jennkins tab >>* **Install as Windows Service:**

- Click *on Install*

Jenkins

- Click on *Yes*,and wait for few seconds while jenkins is restarting

🛈 Jenkins

- Now enter your credentials to login to Jenkins

- On Click on login button with correct credentials, Dashboard screen will shown:

Jenkins

Step 4: Jenkins and Tomcat Setup

I) Download Tomcat :

- Go to official website of tomcat *http://tomcat.apache.org/download-80.cgi*

- Go to the '**Binary Distributions'** section. **Download the Windows zip file** according to your system.

- Unzip the contents of the downloaded zip file at your desired location.

II) Setting up Jenkins and Tomcat

- Copy the *Jenkis.war file* which was downloaded in *Step 2(I)* and copy it to the *webapps folder* in the *tomcat folder(Tomcat installation location).*

- Now open the command prompt. From the command prompt, browse to the directory where the tomcat folder is located. Browse to the bin directory in this folder and *run* the *start.bat file*. **e.g.**

```
E:\Apps\tomcat8\bin>startup.bat
```

Jenkins

- Once the processing is complete without major errors, the following line will come in the output of the command prompt.

INFO: Server startup in 1302 ms

- Open the browser and go to the link − **http://localhost:8080/jenkins**. Jenkins will be up and running on tomcat.

Jenkins

Step 5: Jenkins – Maven Setup

I) Download Maven :

- Go to the official website of Apache Maven project *https://maven.apache.org/download.cgi*

- While browsing to the site, go to the Files section and download the link to the *Binary.zip file.*

- Once the file is downloaded, extract the files to the relevant application folder. For this purpose, the maven files will be placed in E:\Apps\apache-maven-3.3.3.

Jenkins

II) Setting up Jenkins and Maven

- In the Jenkins dashboard (Home screen), click Manage Jenkins from the left-hand side menu.
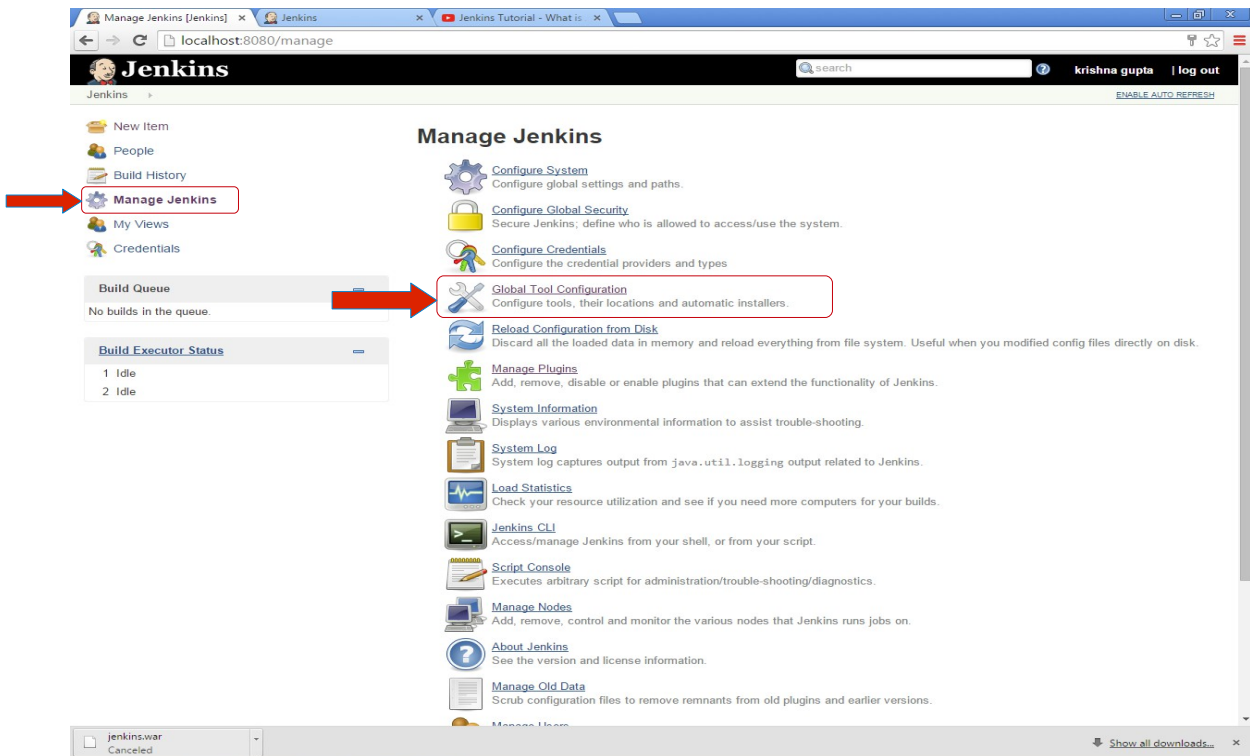
- Then, click on *'Global Tool Configuration'* from the right hand side.

- In the Configure system screen, scroll down till you see the Maven section and then click on the *'Add Maven'* button.

- Uncheck the *'Install automatically'* option.

- Add any name for the setting and the location of the MAVEN_HOME.

- Then, click on the *'Save'* button at the end of the screen.

Jenkins

- You can now create a job with the '**Maven project**' option. In the Jenkins dashboard, by clicking the *New Item option*.

🧔 **Jenkins**

Configure Jenkins :

■ Configuration of JDK location :

- In the Jenkins dashboard (Home screen), click Manage Jenkins from the left-hand side menu.
- Then, click on *'Global Tool Configuration'* from the right hand side.
- In the Configure system screen, scroll down till you see the JDK section and then click on the *'Add JDK'* button.

- Uncheck the *'Install automatically'* option.

- Add jdk name for the setting and the location of the JAVA_HOME.

- Then, click on the *'Save'* button at the end of the screen.

■ Configuration the GIT location :

- In the Jenkins dashboard (Home screen), click Manage Jenkins from the left-hand side menu.
- Then, click on *'Global Tool Configuration'* from the right hand side.
- In the Configure system screen, scroll down till you see the JDK section and then click on the *'Add JDK'* button.

- Uncheck the *'Install automatically'* option.

- Add jdk name for the setting and the location of the JAVA_HOME.

- Then, click on the *'Save'* button at the end of the screen.

In Addition you can also do configuration of :

- **Secure Jenkins**
- **Generate ssh key for Jenkins user**

Jenkins

Jenkins management

- **Plug-in management**
- **Restart your Jenkins**

Jenkins Plugins:

Jenkins has over 1000+ plugins,that are classified as:

- ### Software configuration management

Version Control Systems

1.Accurev	2.Bazaar	3.BitKeeper	4.ClearCase	5.Darcs	6.Dimensions	7.Git
8.Harvest	9.MKS	10.Integrity	11.PVCS	12.StarTeam	13.Subversion	14.Team
15.Foundation Server		16.Visual SourceSafe etc...				

- ### Builders(Build and Test)

Build Tools		Test Frameworks	
1.Ant	2.Maven	1.Junit	2.Nunit
3.MSBuild	4.Cmake	3.MSTest	4.Selenium
5.Gradle	6.Grails	5.Fitnesse	
7.Scons	8.Groovy etc...	etc...	

- ### Virtual Machine Controllers

1.Amazon EC2	2.VMWare	3.VirtualBox	4.Xen	5.Libvirt

- ### Notifiers

1.Twitter	2.Campfire	3.Google Calendar	4.IM	5.IRC
6.Lava Lamp	7.Sounds	8.Speak etc...		

- ### Static Analyzers

1.Checkstyle	2.CodeScanner	3.DRY	4.Crap4j	5.Findbugs
6.PMD	7.Fortify	8.Sonar	9.FXCop etc...	

- ### Authorization

1.Active Directory	2.LDAP etc...