

# JENKINS PRODUCT BASED COMPANY QUESTIONS

## 1) What do mean by Build?

Build is nothing but compiling of sources into distributed artifacts. So developers will be writing the source code...and it needs to be compiled and packaged so then it can be distributed to the customers or users for using it..... So this build might be happening after development....it can happen during development or it can happen on a periodic basis like every week or every month...

## 2) What is CI?

CI is nothing but a software development practice where members of each team integrate their work daily....by integrating their work daily, they can test it every day and it reduces the problems and it can identify the problems earlier and they can fix it earlier...then we can make the product more stable... That is the main use of CI ....If we start the building at the end of the project we will identify so many issues, those are integration issues because code will be developed by the multiple developers.....if we start integrating them daily so you identify issues earlier and we can fix them earlier....

So that our code quality will get increased and will not see any integration issues during release time.....

### 3) How does CI works? How do people work day to day tasks?

For any CI there will be a-----VCS where it keeps tracking of source code, and Build script where we have given instructions to build, to deploy, and to test and CI server to manage all these things , and developers who are writing/modifying the source code.....

- Developers do they work and send changes to their VCS repository.....that's their job
- And CI server will continuously checks is their any change to VCS
- If there's a change pull that change .....& calls the build script.....build it, deploy it and test it....and gives the feedback
- Developers will make a change .....
- Feedback is generated by CI server about the quality of code

change.....something bad he has done its available for him right

away.....that is why continuous integration is so powerful....

- It integrates VCS .....It integrates build and deployment systems.....and consolidate whole things & give feedback on the quality of change.....it brings all systems together.....

#### 4) What are the Prerequisites for CI

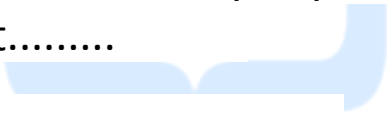
we should have a project with your buildscript it might be written in ant or it can be a maven too...and we have to maintain a source repository...like svn, git, tfs etc.....

#### 5) Which tool do you use for CI? And how it works?

We use Jenkins as a CI tool.....and coming to work process.....multiple developers are working on the project.....svn is the source repository which keeps the versioning of our files and maintaining our files data.....and jenkins is the CI system which pole the source repository and starts the build...so the build script is provided in AnT.....so, it will execute the build script and it will generates the artifact..... Artifact is nothing but the output of our source code... So after compiling and packaging our source code the artifact will be generated... So once the artifact is generated from the given build script.....and we deploy the artifacts in to the tomcat server and then QA guys will go for testing and validate it...and if there are any automated tests they can run and reports will be published from the jenkins dashboard or CI system dashboard.....So, this is the process how jenkins works.....

## 6) What is the difference between SVN and Jenkins?

Svn is just keep track of our source code versioning.....  
But Jenkins will manage the running builds and will manage your builds...so, once we configure the project with Jenkins ....it will take care of building it whenever changes happen in your source code and publish if any build failure or if there are any issues with the test cases... So, as multiple developers working on the same source code better to integrate daily so that we can identify any dependency issues or any issues earliest.....



## 7) What kind of plugins you have installed and used?

subversion/git , cobertura, clone, email, shelve, junit, thin backup...these are the plugins I have installed .....

In the Jenkins Dashboard left side panel there is option manage Jenkins ---> click on manage plugin ---> go to available option ----> We can search for the plugin what we have to integrate with Jenkins for ex: git we just type the git plugin in the filter option and it produces the git related

plugins then we have to select it click on Install with or without restart.....

Basically Git plugin have the dependencies also one is git client plugin and scm API plugi

After installation it changes from pending state to success state....

And then once again restart the jenkins and now we can configure the job....

### 8) Have u create jenkins job from the scratch

Yes.....Before creating a job i will do some system configuration bcz i have to configure the jdk subversion ant version details and path of that tools in the related sessions and i also configured the E-mail notification session and so on.....and i will also install if any new plugins needed and after that i will create the job and i configured that job session and i trigger a build and i view console output and i send the status of the builds to the dev team.....

All the configurations of jenkins server will store in config.xml file

### 10) Have you set up jenkins? How did you install jenkins?

First install jenkins and go to jenkins installed folder and copy jenkins war file and paste it in tomcat webapps....

Next type :

```
$ Java -jar jenkins.war $ Jenkins.rpm
```

And now go to tomcat path ---->ls \$cdbin

```
$ chmod 777 catalina.sh
```

```
$ sh startup.sh
```

Now open browser type the default <http://localhost:8080> :  
then it will shows tomcat homepage

If we want to open jenkins homepage: type  
<http://localhost:8080/jenkins>..... Thats it....

### 11) How do you start jenkins as a service?

Install yum---->install jenkins.war

### 12)How do you start / stop /restart a jenkins

First we have go tomcat installed path ---> bin ---->

To start jenkins type: \$ sh startup.sh

To stop jenkins type: \$ sh shutdown.sh

And to restart type: stop the server and then start again...

If we want to know whether jenkins server is in on or not:  
just type: ps -ef | grep tomcat

### 13)How do you change your port no for your jenkins and what is the default port?

Ans: go to the tomcat installation folder and from there go to conf folder and open service.xml there change the port (connector) to 8081 or any other port no from

8080...and go to bin and double click on service.bat and click on sh startup.sh/ startup.bat and run jenkins on the new port number...

#### 14) Where do you find default jenkins logs?

system log will captures the output from java.util.loggingoutputrelatedto jenkins.

select the "system log" from the "Manage Jenkins" page:

From there, you can create a custom log recorder, which helps you group relevant logs together while filtering out the noise.

A screen will open like log recorders-----> All jenkins logs--->click on add new recorder----> we have to choose/enter the name that makes sense to us----->click on ok

---> and then we have to configure the logger details ex:udson.plugin.disk\_usage and select the log level to find and click on add....and click on save....

#### 15)How do find jenkins server is running or not?

Ans: go to console --->ps -ef | grep jenkins

### 16)How do you stop jenkins when some build jobs are in progress

(jenkins\_url)/safeRestart

- Allows all running jobs to complete.New jobs will remain in the queue to run after the restart complete.

(jenkins\_url)/restart

- Forces a restart without waiting for builds to complete

### 17) How do you migrate jenkins from one machine to another

1. Install the fresh jenkins in the new server
2. Be sure old and new jenkins are stopped
3. Archive all the content of the jenkins\_home(.jenkins) of the old jenkins instance
4. Extract the old jenkins in to the new jenkins\_home directory
5. Launch the new jenkins
6. Do not forget to change documentation / links to your instance of your new jenkins

### 18)Where does jenkins store its configuration data

On the jenkins home directory there are some common folders we found like .jenkins, jobs and plugins folders etc . In jobs folder we found all the configuration data will be stored...and in the jobs folder the 3



important subfolders are builds workspace folders and **config.xml** file....

19) What is the default jenkins home directory? How do you change it?

Dot jenkins is the default home directory.... To change the jenkins home directory goto:

```
vi.bash rc export jenkins_home=path....
```

20) Where does jenkins store global configuration and job related configuration

Jenkins stores the configuration for each job within an anonymous directory in jobs/. The job configuration file is config.xml, the builds are stored in builds/, and the working directory is workspace/.

Or

1. 2.

Or

The JENKINS\_HOME directory is laid out like this:

\$JENKINS\_HOME contains system level configuration files such as the Jenkins master config.xml and the credentials.xml

\$JENKINS\_HOME/jobs contains all the individual job configuration and build data in a separate subdirectory for

each job, e.g. for the job "myjob" you'll have `$JENKINS_HOME/jobs/myjob/config.xml` plus other files that store the data for individual jobs.

## 21) How do you restore the system configuration and all the jobs?

----->So if you want to restore the system configuration and all the jobs without the build history, you'll need to copy:

```
$JENKINS_HOME/.xml $JENKINS_HOME/jobs/*/config.xml
```

(using Antglob notation, `**` denotes all subdirectories under `$JENKINS_HOME/jobs`)

---->You'll also need `$JENKINS_HOME/plugins/*` if you installed any extra plugins. If you want to restore all the build history, copy `$JENKINS_HOME/jobs/*` instead of just the `config.xml` files.

## 22) How many jobs files in jenkins \

There are four types of jobs we can create....

1. Free style project
2. Maven project
3. Build multi configuration project
4. External jobs

Mostly we will create a job in freestyle projects/pipelines....because it supports all kinds of tools.

### 23) How do you create a Jenkins job ii) How did you setup build and deployment using jenkins for your project?

Ans: i) in jenkins dashboard---> in the left side panel ----> there will be an option called "new job" click on it and create the job.....

li) go to manage jenkins ----> configure system----> there we have to configure svn path and ant path and save it and now go to project and click on configure and there we invoke ant for build and for deployment we copy the artifacts in to tomcat server...

### 24) What is parameterised builds

Go to project -----> go to configure page ----> there we will see the option "this build is parameterised" check it that box---> we have to click on dropdown list "add parameter"-----> here we have different modules.....based on our need we will select one....i.e., boolean, cvs, parameter etc....

### 25) What is cron tab and cron scm

In jenkins --> go to project configuration page and there the option "Build periodically" that is the cron tab...whether there is change in scm or not it will go and build periodically...

Poll scm : If there is any change in svn it will automatically build....if there is no changes done in vcs then it will remains constant without doing any kind of build....

## 26)How do you configure security user data base for your jenkins

Go to the Configure Global Security screen and choose "enable security"

Select "Jenkins's own user database" as the security realm

Place a checkmark next to "Allow users to sign up"

Select "Matrix-based security" as the authorization

Give anonymous user the read access

In the text box below the table, type in your user name (you'd be creating this later) and click "add"

Give yourself a full access by checking the entire row for your user name

Scroll all the way to the bottom, click "save"

At this point, you'll be taken back to the top page, and Jenkins is successfully secured.

Now you need to create an user account for yourself.

1. Click "login" link at the top right portion of the page

2. Choose "create an account"

3. Use the user name you've used in the above step, and fill in the rest.

## 27) Are you using LDAP for build organisation?

No but i have an idea on LDAP...

If Jenkins is running on a Windows server then it is better to install the [Active Directory plugin](#). On a Linux host you have an option to either use the [Active Directory plugin](#) or an LDAP based authentication. To configure the LDAP to work with Active Directory, provide the following:

Server

Root DN

User Search Filter

Manager DN

Manager Password

## 28) Do you take backup of jenkins if yes how?

Yes we take backup of jenkins daily....for that we have to copy .jenkins folder to backup....this we will set in cron tab....

## 29) Do you take backup only job related configurations other than workspace

In jenkins home ---> .jenkins-----> jobs-----> builds and config.xml

We will configure this like: \$ cd .jenkins

\$ls

\$ cd jobs

\$ls

\$ builds config.xml

### 30)How do you install jenkins plugin manually?

Save the downloaded \*.hpi/\*.jpi file into the \$JENKINS\_HOME/plugins directory. You will then need to restart Jenkins (many containers let you do this without restarting the container)

### 31)How do you setup distributed builds

Jenkins supports the "master/slave" mode, where the workload of building projects are delegated to multiple "slave" nodes, allowing a single Jenkins installation to host a large number of projects, or to provide different environments needed for builds/tests. This document describes this mode and how to use it.

### 32)How do you add a node to your master or how to create nodes?

go to jenkins dashboard---->manage jenkins----->and then click on nodes----->  
and now we can add a new node.

33)How do you see the process id and whether jenkins server is running or not

```
ps -ef | grep jenkins
```

```
sudo kill -kill <pid>
```

34)What are the best practise do u follow in jenkins

Always secure Jenkins.

**This best practice is around authenticating users and enforcing access control on a Jenkins instance**

In the default configuration, Jenkins doe's not perform any security checks. This means any person accessing the website can configure Jenkins and jobs, and perform builds. While this configuration is normally acceptable for intranet use and quick setup, it introduces high security risks, like someone accidentally deleting your build jobs, reconfiguring your job to run every minute, kicking off too many builds at the same time, reconfiguring your build instance, etc.

Backup Jenkins Home regularly.

Use "file fingerprinting" to manage dependencies.

When you have interdependent projects on Jenkins, it often becomes hard to keep track of which version of this is used by which version of that. Jenkins supports "file fingerprinting" to simplify this, so make best use of it.

The most reliable builds will be **clean builds**, which are built fully from Source Code Control.

To ensure a build can be reproducible, the build must be a **clean build**, which is built fully from Source Code Control. This practice also implies that all code including third-party jars, build scripts, release notes, etc. must be checked into Source Code Control.

Integrate tightly with your issue tracking system, like JIRA or bugzilla, to reduce the need for maintaining a Change Log

The integration helps to track changes as they are made, including build status, what build has been performed for this requirement or defects, and the link to the actual build results and artifacts.

Integrate tightly with a repository browsing tool like FishEye if you are using Subversion as source code management tool

Repository browsing provides a quick update on what happens on a Subversion repository. It also provides a graphical diff on what changes have been made from the previous build.

Always configure your job to generate trend reports and automated testing when running a Java build

Trends helps project managers and developers quickly visualise current project progress status. Moreover, unit testing is often not enough to provide confidence that the delivered software complies to the desired quality.



Set up Jenkins on the partition that has the most free disk-space

Archive unused jobs before removing them.

Setup a different job/project for each maintenance or development branch you create

Set up email notifications mapping to ALL developers in the project, so that everyone on the team has his pulse on the project's current status.

Take steps to ensure failures are reported as soon as possible.



