

<Codingcompiler/>

Advanced Jenkins Interview Questions And Answers 2018

Interview Questions

Codingcompiler.com

Top 28 **Jenkins Interview Questions** And Answers For Experienced 2018. If you are looking for Jenkins interview questions with answers, then you are at right place. Here [coding compiler](#) sharing a list of 28 real-time interview *questions on Jenkins*. These [Jenkins](#) interview questions for devops will help you to crack your next **Jenkins job interview**. Happy reading and all the best for your future.

Jenkins Interview Questions

1. What is **Jenkins**?
2. Why do we use Jenkins?
3. What is Maven and what is Jenkins?
4. What is the difference between **Hudson and Jenkins**?
5. What is meant by *continuous integration in Jenkins*?
6. Why do we use **Jenkins with selenium**?
7. What are **CI Tools**?
8. What is a *CI CD pipeline*?
9. What is *build pipeline* in Jenkins?
10. What is a **Jenkins pipeline**?
11. What is a DSL Jenkins?
12. What is continuous **integration and deployment**?
13. What is the tool used for provisioning and configuration?
14. What is the difference between **Maven, Ant, and Jenkins**?
15. Which SCM tools Jenkins supports?
16. How schedule a *build in Jenkins*?
17. Why do we use Pipelines in Jenkins?
18. What is a **Jenkinsfile**?

19. How do you create Multibranch Pipeline in Jenkins?
20. What is the blue ocean in Jenkins?
21. What are the important **plugins in Jenkins**?
22. What are *Jobs in Jenkins*?
23. How do you create a Job in Jenkins?
24. How do you configuring **automatic builds in Jenkins**?
25. How to create a backup and copy files in **Jenkins**?

Jenkins Interview Questions And Answers

| Jenkins Interview Questions | |
|-----------------------------|---|
| Jenkins is an | Open source software |
| Jenkins is an | Automation server |
| Jenkins can | Help to automate the software development process. |
| Jenkins can | Automate the process with continuous integration and facilitate technical aspects of continuous delivery. |
| Jenkins developed by | Jenkins is a fork of a project called Hudson. |
| Jenkins License | MIT |
| Jenkins has written in | Java |

Jenkins Interview Questions # 1) What is Jenkins?

Answer # Jenkins is an open source automation server. Jenkins is a continuous integration tool developed in Java. **Jenkins** helps to automate the non-human part of software development process, with **continuous integration** and facilitating technical aspects of continuous delivery.

Jenkins Interview Questions # 2) Why do we use Jenkins?

Answer # Jenkins is an **open-source** continuous integration software tool written in the Java programming language for testing and reporting on isolated changes in a larger code base in real time. The **Jenkins software** enables developers to find and solve defects in a code base rapidly and to automate testing of their builds.

Jenkins Interview Questions # 3) What is Maven and what is Jenkins?

Answer # **Maven is a build tool**, in short a successor of ant. It helps in build and version control. However, **Jenkins is continuous integration system**, where in maven is used for build. Jenkins can be used to automate the deployment process.

Jenkins Interview Question # 4) What is the difference between Hudson and Jenkins?

Answer # **Jenkins is the new Hudson**. It really is more like a rename, not a fork, since the whole development community moved to Jenkins. (Oracle is left sitting in a corner holding their old ball “**Hudson**“, but it’s just a soul-less project now.). In a nutshell **Jenkins CI** is the leading open-source continuous integration server.

Jenkins Interview Questions # 5) What is meant by continuous integration in Jenkins?

Answer # **Continuous integration** is a process in which all development work is integrated as early as possible. The resulting artifacts are automatically created and tested. This process allows to identify errors as early as possible. **Jenkins is a popular open source tool** to perform continuous integration and build automation.

Interview Questions on Jenkins

Continuous Integration Interview Questions # 6) Why do we use Jenkins with selenium?

Answer # Running **Selenium tests in Jenkins** allows you to run your tests every time your software changes and deploy the software to a new environment when the tests pass. Jenkins can schedule your tests to run at specific time.

Jenkins CI CD Interview Questions # 7) What are CI Tools?

Answer # Here is the list of the top 8 **Continuous Integration tools**:

- Jenkins
- TeamCity
- Travis CI
- Go CD
- Bamboo
- GitLab CI
- CircleCI
- Codeship

Jenkins Pipeline Interview Questions # 8) What is a CI CD pipeline?

Answer # A **continuous integration** and deployment pipeline (CD/CI) is such an important aspect of a software project. It saves a ton of manual, error-prone deployment work. It results in higher quality software for continuous integration, **automated tests**, and code metrics.

Jenkins Tough Interview Questions # 9) What is build pipeline in Jenkins?

Answer # Job chaining in **Jenkins** is the process of automatically starting other job(s) after the execution of a job. This approach lets you build **multi-step build pipelines** or trigger the rebuild of a project if one of its dependencies is updated.

Jenkin Interview Questions # 10) What is a Jenkins pipeline?

Answer # The **Jenkins Pipeline plugin** is a game changer for Jenkins users. Based on a *Domain Specific Language (DSL)* in Groovy, the Pipeline plugin makes pipelines scriptable and it is an incredibly powerful way to develop complex, multi-step **DevOps pipelines**.

Jenkins Interview Questions And Answers For Experienced

Jenkins Interview Questions # 11) What is a DSL Jenkins?

Answer # The Jenkins “Job DSL / Plugin” is made up of two parts: The Domain Specific Language (DSL) itself that allows users to describe jobs using a Groovy-based language, and a Jenkins plugin which manages the scripts and the updating of the Jenkins jobs which are created and maintained as a result.

Jenkins Interview Questions For Devops # 12) What is continuous integration and deployment?

Answer # **Continuous Integration (CI)** is a development practice that requires developers to integrate code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early.

Jenkins Real Time Interview Questions # 13) What is the tool used for provisioning and configuration?

Answer # Ansible is an agent-less configuration management as well as orchestration tool. In Ansible, the configuration modules are called “Playbooks”. Like other tools, Ansible can be used for cloud provisioning.

Jenkins Questions And Answers # 14) What is the difference between Maven, Ant and Jenkins?

Answer # Maven and ANT are build tool but main difference is that maven also provides dependency management, standard project layout and project management. On difference between Maven, ANT and Jenkins, later is a continuous integration tool which is much more than build tool.

Jenkins Questions # 15) Which SCM tools Jenkins supports?

Answer # Jenkins supports version control tools, including AccuRev, CVS, Subversion, Git, Mercurial, Perforce, ClearCase and RTC, and can execute Apache Ant, Apache Maven and sbt based projects as well as arbitrary shell scripts and Windows batch commands.

Jenkins Interview Questions For Testers

Devops Interview Questions Jenkins # 16) How schedule a build in Jenkins?

Answer # In Jenkins, under the job configuration we can define various build triggers. Simple find the ‘Build Triggers’ section, and check the ‘Build Periodically’ checkbox. With the periodically build you can schedule the build definition by the date or day of the week and the time to execute the build.

The format of the ‘Schedule’ textbox is as follows:

MINUTE (0-59), HOUR (0-23), DAY (1-31), MONTH (1-12), DAY OF THE WEEK (0-7)

Continuous Integration Interview Questions # 17) Why do we use Pipelines in Jenkins?

Answer # Pipeline adds a powerful set of automation tools onto Jenkins, supporting use cases that span from simple continuous integration to comprehensive continuous delivery pipelines. By modeling a series of related tasks, users can take advantage of the many features of Pipeline:

- **Code:** Pipelines are implemented in code and typically checked into source control, giving teams the ability to edit, review, and iterate upon their delivery pipeline.
- **Durable:** Pipelines can survive both planned and unplanned restarts of the Jenkins master.
- **Pausable:** Pipelines can optionally stop and wait for human input or approval before continuing the Pipeline run.
- **Versatile:** Pipelines support complex real-world continuous delivery requirements, including the ability to fork/join, loop, and perform work in parallel.
- **Extensible:** The Pipeline plugin supports custom extensions to its DSL and multiple options for integration with other plugins.

Questions on Jenkins # 18) What is a Jenkinsfile?

Answer # A Jenkinsfile is a text file that contains the definition of a Jenkins Pipeline and is checked into source control.

Creating a Jenkinsfile, which is checked into source control, provides a number of immediate benefits:

1. Code review/iteration on the Pipeline
2. Audit trail for the Pipeline
3. Single source of truth for the Pipeline, which can be viewed and edited by multiple members of the project.

Interview Questions on Jenkins # 19) How do you create Multibranch Pipeline in Jenkins?

Answer # The Multibranch Pipeline project type enables you to implement different Jenkinsfiles for different branches of the same project. In a Multibranch Pipeline project, Jenkins automatically discovers, manages and executes Pipelines for branches which contain a Jenkinsfile in source control.

Devops Jenkins Interview Questions # 20) What is blue ocean in Jenkins?

Answer # Blue Ocean is a project that rethinks the user experience of Jenkins, modelling and presenting the process of software delivery by surfacing information that's important to development teams with as few clicks as possible, while still staying true to the extensibility that is core to Jenkins.

Jenkins Interview Questions For Automation Testers

Jenkins Interview Questions For DevOps # 21) What are the important plugins in Jenkins?

Answers # Here is the list of some important **Plugins in Jenkins**:

1. Maven 2 project
2. Git
3. Amazon EC2
4. HTML publisher
5. Copy artifact
6. Join
7. Green Balls

Interview Questions on Maven and Jenkins # 22) What are Jobs in Jenkins?

Answer # **Jenkins** can be used to perform the typical build server work, such as doing continuous/official/nightly builds, run tests, or perform some repetitive batch tasks. This is called “**free-style software project**” in Jenkins.

Jenkins Advanced Interview Questions # 23) How do you create a Job in Jenkins?

Answer # Go to **Jenkins** top page, select “New Job”, then choose “Build a free-style software project”. This job type consists of the following elements:

optional **SCM**, such as **CVS** or **Subversion** where your source code resides.

optional triggers to control when Jenkins will perform builds.

some sort of build script that performs the build (**ant**, **maven**, **shell script**, **batch file**, etc.) where the real work happens optional steps to collect information out of the build, such as archiving the artifacts and/or recording javadoc and test results.

optional steps to notify other people/systems with the build result, such as sending e-mails, IMs, updating issue tracker, etc.

Selenium Jenkins Interview Questions # 24) How do you configuring automatic builds in Jenkins?

Answer # **Builds in Jenkins** can be triggered periodically (on a schedule, specified in configuration), or when source changes in the project have been detected, or they can be automatically triggered by requesting the URL:

<http://YOURHOST/jenkins/job/PROJECTNAME/build>

Jenkins CI Interview Questions And Answers # 25) How to create a backup and copy files in Jenkins?

Answer # To create a backup, all you need to do is to periodically back up your `JENKINS_HOME` directory. This contains all of your build jobs configurations, your slave node configurations, and your build history. To create a back-up of your Jenkins setup, just copy this directory.

Jenkins Real-Time Interview Questions

26) What is the trustAnchors parameter must be non-empty error and how can you solve it?

A) This trustAnchors parameter must be non-empty error means that the truststore you specified was not found, or couldn't be opened due to access permissions for example.

EJP basically answered the question (and I realize this has an accepted answer) but I just dealt with this edge-case gotcha and wanted to immortalize my solution. I had the `InvalidAlgorithmParameterException` error on a hosted jira server that I had previously set up for SSL-only access.

The issue was that I had set up my keystore in the PKCS#12 format, but my truststore was in the JKS format. In my case, I had edited my `server.xml` file to specify the `keystoreType` to PKCS, but did not specify the `truststoreType`, so it defaults to whatever the `keystoreType` is. Specifying the `truststoreType` explicitly as JKS solved it for me.

27) What are the feature differences between Jenkins and Hudson?

A) Jenkins is the recent fork by the core developers of Hudson. To understand why, you need to know the history of the project. It was originally open source and supported by Sun. Like much of what Sun did, it was fairly open, but there was a bit of benign neglect. The source, trackers, website, etc. were hosted by Sun on their relatively closed `java.net` platform.

Then Oracle bought Sun. For various reasons Oracle has not been shy about leveraging what it perceives as its assets. Those include some control over the logistic platform of Hudson, and particularly control over the Hudson name. Many users and contributors weren't comfortable with that and decided to leave.

So it comes down to what Hudson vs Jenkins offers. Both Oracle's Hudson and Jenkins have the code. Hudson has Oracle and Sonatype's corporate support and the brand. Jenkins has most of the core developers, the community, and (so far) much more actual work.

In fact, arguably it was Oracle who did the forking! And technically, too, that's kinda what happened.

It's interesting to see what comes out of "Hudson" though. While the "Winston summarizes the state and rosy future of the Hudson project" stuff they posted on the (new) Hudson website originally seemed like odd humour to me, perhaps this was a purposeful takeover, and the Sonatype guys actually have some big ideas up their sleeve. This analysis, suggesting a deliberate strategy by Oracle/Sonatype to oust Kohsuke and crew to create a more "enterprisy" Hudson is a very interesting read!

In any case, this brief comparison a fortnight after the split—while not exactly scientific—shows Jenkins to be by far more active of the two projects.

Jenkins has continued the path well-trodden by the original Hudson with frequent releases including many minor updates.

Oracle seems to have largely delegated work on the future path for Hudson to the Sonatype team, who has performed some significant changes, especially with respect to Maven. They have jointly moved it to the Eclipse foundation.

I would suggest that if you like the sound of:

Less frequent releases but ones that are more heavily tested for backwards compatibility (more of an "enterprise-style" release cycle)

A product focused primarily on strong Maven and/or Nexus integration (i.e., you have no interest in Gradle and Artifactory etc)

Professional support offerings from Sonatype or maybe Oracle in preference to Cloudbees etc

You don't mind having a smaller community of plugin developers etc.
, then I would suggest Hudson.

Conversely, if you prefer:

More frequent updates, even if they require a bit more frequent tweaking and are perhaps slightly riskier in terms of compatibility (more of a "latest and greatest" release cycle)

A system with more active community support for e.g., other build systems / artifact repositories

Support offerings from the original creator et al. and/or you have no interest in professional support (e.g., you're happy as long as you can get a fix in next week's "latest and greatest")

A classical OSS-style witches' brew of a development ecosystem

then I would suggest Jenkins.

Jenkins CI Interview Questions

28) How to trigger a build remotely from Jenkins? How to configure Git post commit hook?

The requirement is whenever changes are made in the Git repository for a particular project it will automatically start Jenkins build for that project.

A) As mentioned in "Polling must die: triggering Jenkins builds from a git hook", you can notify Jenkins of a new commit:

With the latest Git plugin 1.1.14 (that I just release now), you can now do this more >easily by simply executing the following command:

```
curl http://yourserver/jenkins/git/notifyCommit?url=<URL of the Git repository>
```

This will scan all the jobs that's configured to check out the specified URL, and if they are also configured with polling, it'll immediately trigger the polling (and if that finds a change worth a build, a build will be triggered in turn.)

This allows a script to remain the same when jobs come and go in Jenkins. Or if you have multiple repositories under a single repository host application (such as Gitis), you can share a single post-receive hook script with all the repositories. Finally, this URL doesn't require authentication even for secured Jenkins, because the server doesn't directly use anything that the client is sending. It runs polling to verify that there is a change, before it actually starts a build.

As mentioned here, make sure to use the right address for your Jenkins server:

since we're running Jenkins as standalone Webserver on port 8080 the URL should have been without the /jenkins, like this:

```
http://jenkins:8080/git/notifyCommit?url=git@gitserver:tools/common.git
```

To reinforce that last point, ptha adds in the comments:

It may be obvious, but I had issues with:

```
curl http://yourserver/jenkins/git/notifyCommit?url=<URL of the Git repository>.
```

The url parameter should match exactly what you have in Repository URL of your Jenkins job.

When copying examples I left out the protocol, in our case ssh://, and it didn't work.

You can also use a simple post-receive hook like in "Push based builds using Jenkins and GIT"

```
#!/bin/bash  
/usr/bin/curl -user USERNAME:PASS -s \
```

`http://jenkinsci/job/PROJECTNAME/build?token=1qaz2wsx`

Configure your Jenkins job to be able to “Trigger builds remotely” and use an authentication token (1qaz2wsx in this example).

However, this is a project-specific script, and the author mentions a way to generalize it.

The first solution is easier as it doesn’t depend on authentication or a specific project.

I want to check in change set whether at least one java file is there the build should start.

Suppose the developers changed only XML files or property files, then the build should not start.

Basically, your build script can:

put a ‘build’ notes (see git notes) on the first call

on the subsequent calls, grab the list of commits between HEAD of your branch candidate for build and the commit referenced by the git notes ‘build’ (git show refs/notes/build): `git diff --name-only SHA_build HEAD`.

your script can parse that list and decide if it needs to go on with the build.

in any case, create/move your git notes ‘build’ to HEAD.

RELATED INTERVIEW QUESTIONS

1. [Chef Interview Questions](#)
2. [Puppet Interview Questions](#)
3. [DB2 Interview Questions](#)
4. [AnthillPro Interview Questions](#)
5. [Angular 2 Interview Questions](#)
6. [Hibernate Interview Questions](#)
7. [ASP.NET Interview Questions](#)
8. [PHP Interview Questions](#)

9. [Kubernetes Interview Questions](#)
10. [Docker Interview Questions](#)
11. [CEH Interview Questions](#)
12. [CyberArk Interview Questions](#)
13. [Appian Interview Questions](#)
14. [Drools Interview Questions](#)
15. [Talend Interview Questions](#)
16. [Selenium Interview Questions](#)
17. [Ab Initio Interview Questions](#)
18. [AB Testing Interview Questions](#)
19. [Mobile Application Testing Interview Questions](#)
20. [Pega Interview Questions](#)
21. [UI Developer Interview Questions](#)
22. [Tableau Interview Questions](#)
23. [SAP ABAP Interview Questions](#)
24. [Reactjs Interview Questions](#)
25. [UiPath Interview Questions](#)
26. [Automation Anywhere Interview Questions](#)
27. [RPA Interview Questions](#)
28. [RPA Blue Prism Interview Questions](#)
29. [Ranorex Interview Questions](#)
30. [AWS Interview Questions](#)

3 thoughts on “Advanced Jenkins Interview Questions And Answers 2018”

BestAnastasia

February 18, 2018 at 10:48 am

I have noticed you don't monetize your website, don't waste your traffic, you can earn additional bucks every month because you've got high quality content. If you want to know how to make extra bucks, search for: Ercannou's essential tools best adsense alternative

Mouni

March 16, 2018 at 11:21 am

Hello There,

Very cool to see your post come up when I was searching this topic!
Is the UiPath SDK supported anymore? I click the link on the homepage and it goes nowhere. I can't find much information about it. Seems like there is an effort to hide it now.
I think writing complex logic using UIPath Studio would result in a bloated mess. Writing even a simple loop with a few assign statements is comically bloated.
Appreciate your effort for making such useful blogs and helping the community.

Thank you,
Mouni

Sunil

March 24, 2018 at 6:56 am

I'm happy with the content. Content present here is lot more than a top websites.

Thank you

Programming Tutorials | Interview Questions | Coding Compiler / Proudly powered by WordPress