**JENKINS**

1. What is Jenkins?

Jenkins is an open-source free automation tool used to build and test software projects. The tool makes it painless for developers to integrate changes to the project. Jenkins' primary focus is to keep track of the version control system and initiate and monitor a build system if there are any changes. It keeps an eye on the entire process and provides reports and notifications to alert.

Some typical reasons as to why Jenkins is so widely used are:

* Developers and testers use Jenkins to detect defects in the software development lifecycle and automate the testing of builds.
* They use it to continuously monitor the code in real-time and integrate changes into the build.
* Jenkins as it turns out, is a great fit for building a [CI/CD pipeline](https://www.simplilearn.com/tutorials/jenkins-tutorial/ci-cd-pipeline) because of its plugin-capabilities, and simple-to-use nature.

2. What are the features of Jenkins?

Some of the crucial features of Jenkins are the following:

* It is a free and open-source automation tool
* Jenkins provides a vast number of plugins
* It is easy to set up and install on multiple operating systems
* Provides pipeline support
* Fast release cycles
* Easy upgrades

3. What is Groovy in Jenkins?

* Apache Groovy is a dynamic object-oriented programming language used as a scripting language for [Java](https://www.simplilearn.com/tutorials/java-tutorial/what-is-java) platforms.
* Groovy is used to orchestrate the Jenkins pipeline and enables different teams to contribute to the work in different languages.
* Groovy's syntax is very similar to that of Java, making it more seamless with the Java interface.
* The language has several features like Java compatibility and Development support.

4. How do you install Jenkins?

Follow the steps mentioned below to [install Jenkins](https://www.simplilearn.com/tutorials/jenkins-tutorial/jenkins-installation-on-windows):

* Install Java
* Install Apache Tomcat Server
* Download Jenkins war File
* Deploy Jenkins war File

5. Which commands can be used to begin Jenkins?

Here are the commands used to start Jenkins:

* Open the command prompt
* After the command prompt opens, browse to the directory where Jenkins war is present
* Then run the following command:

D:\>Java -jar Jenkins.war

7. What are the differences between Continuous Integration, Continuous Delivery, and Continuous Deployment?

|  |  |  |
| --- | --- | --- |
| Continuous Integration | Continuous Delivery | Continuous Deployment |
| Continuous Integration (CI) is a DevOps software development practice that permits developers to combine/merge the changes to their code in the central repository to run automated builds and tests. | Continuous Delivery (CD) refers to the building, testing, and delivering improvements to the software code. The most critical part of the CD is that the code is always in a deployable state. | Continuous Deployment (CD) is the ultimate stage in the DevOps pipeline. It  refers to automatic release of any developer changes from the repository to the production stage. |

What is a CI/CD pipeline?

CI/CD Pipeline or Continuous Integration/ Continuous Delivery is considered the DevOps approach's backbone. The pipeline is responsible for building codes, running tests, and deploying new software versions.

9. What is a Jenkins pipeline?

* The pipeline represents the continuous delivery and continuous integration of all the jobs in the SDLC and DevOps life cycle.
* The Jenkins pipeline is a set of plugins that support implementation and integration of continuous delivery pipelines into Jenkins. It connects this pipeline in a particular format by Jenkins.
* The Jenkins pipeline solves several problems like the maintenance of thousands of jobs and maintaining deployment with needing to resort to other powerful methods.

10. Name the three different types of pipelines in Jenkins?

The three different types of Jenkins pipelines are:

* CI/CD pipeline
* Scripted pipeline
* Declarative pipeline

11. How can you set up a Jenkins job?

To set up a Jenkins job, you may follow these steps:

* Select New item from the menu
* Next, enter a name for the job and select a free-style job
* Click on OK to create a new job
* Hence, the next page that appears will allow you to configure your job.

12. What are the requirements for using Jenkins?

To use Jenkins, you require the following:

* A source code repository that can be accessed, for example, a Git repository.
* A build script, for example, a Maven script.

13. Name the two components that Jenkins is mostly integrated with.

Jenkins is typically integrated with these two components:

1. Version Control systems like Git and SVN (Apache Subversion)
2. Build tools like [Maven](https://www.simplilearn.com/tutorials/maven-tutorial/what-is-maven)

14. Name some of the useful plugins in Jenkins.

Some of the plugins in Jenkins include:

* [Maven 2 project](https://www.simplilearn.com/tutorials/maven-tutorial/maven-project-in-eclipse)
* [Amazon EC2](https://www.simplilearn.com/tutorials/aws-tutorial/aws-ec2)
* Copy artifact
* Join
* HTML publisher
* Green Balls

How can you deploy a custom build of a core plugin?

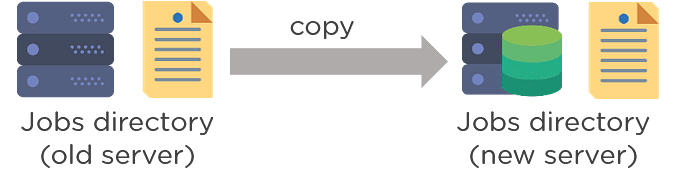
If you wish to deploy a custom build of a core plugin, you follow the following steps:

* Stop Jenkins
* Then copy the custom HPI to $Jenkins\_Home/plugins
* After that, delete the previously expanded plugin directory
* Next, make an empty file called <plugin>.hpi.pinned
* Finally, start Jenkins

17. What could be the steps to move or copy Jenkins from one server to another?

There are multiple ways to move or copy Jenkins from one server to another:

* You may move a job from one Jenkins installation to another just by copying the corresponding job directory.
* You may make a copy of an already existing job by making a clone of the job directory with an uncommon name.
* You may also just rename a current job by renaming a directory.



What is Jenkinsfile?

Jenkins file is a text file that has a definition of a Jenkins pipeline and is checked into the source control repository. It enables code review and iteration on the pipeline. It also permits an audit trail for the pipeline.

22. Differentiate between Maven, Ant, and Jenkins.

|  |  |  |
| --- | --- | --- |
| Maven | ANT | Jenkins |
| 1. Build tool 2. Perform build operations | 1. Build tool 2. Perform build operations | 1. Continuous Integration tool 2. Jenkins may run unit tests and deploy applications |

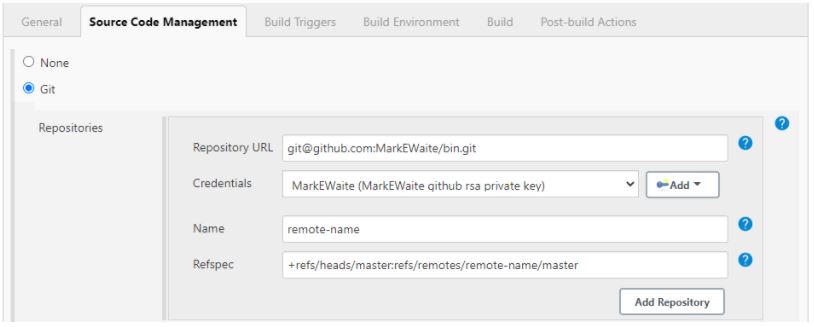
23. Differentiate between Bamboo and Jenkins?

|  |  |  |
| --- | --- | --- |
| S.no | Bamboo | Jenkins |
| 1 | Commercial tool | Open-source tool |
| 2 | Dedicated development team | Huge global community |
| 3 | Comparatively more user friendly | Less user-friendly |
| 4 | Many built-in features and plugins are available in the Atlassian marketplace. | Many plugins to perform different functions |

26. What is the process to integrate Git with Jenkins?

To integrate Git with Jenkins, you can follow the following steps:

* First, create a new Jenkins job and open the Jenkins dashboard.
* Now, enter the desired project name and select the job type.
* Click on OK.
* Then enter the project information.
* After that, visit the 'Source Code Management' tab.



Source: <https://plugins.jenkins.io/git/>

* If the Git plugin is pre-installed in Jenkins, there will be 'Git'.
* If it is not installed, you must reinstall the plugins (GitHub plugin, GitHub Branch Source plugin, GitHub API plugin, Git client plugin, etc.).
* After we install the plugins, restart Jenkins.
* To check if Git is installed, you can go to Command Prompt and type Git, and you would see various options like usage, version, help, etc.

27. Explain Kubernetes, and how can you integrate Jenkins with Kubernetes?

* [Kubernetes](https://www.simplilearn.com/tutorials/kubernetes-tutorial/what-is-kubernetes) is a portable and open-source platform that is used for managing workloads and services that are containerized.
* With the help of Kubernetes, the group of hosts running the Linux containers can be easily and efficiently managed.
* To manage a Continuous Delivery (CD) pipeline, the most efficient way is to deploy Jenkins with Kubernetes Engine.
* Kubernetes enables the creation of multiple container instances to satisfy more fault tolerance.
* Kubernetes deploy plug may be used with Jenkins for Continuous Deployment.

29. What is the process to configure Third-party tools in Jenkins?

The process to configure Third-party tools in Jenkins can be seen in four significant steps:

* Install the third-party software
* Then install a Jenkins plugin supporting the third-party tool
* Now, configure the tool from the Manage Jenkins section
* Finally, your plugin is ready to be used

30. What are some of the default environmental variables in Jenkins?

Some of the Jenkins environmental variables are:

* $JOB\_NAME - The name that you give your job when it is first set up.
* $NODE\_NAME - This is the name of the node on which the current build is running.
* $WORKSPACE - Refers to the path of the workspace
* $BUILD\_URL - Indicates the URL where the results of the builds can be found.
* $JENKINS\_URL - This is set to the URL of the Jenkins master that is responsible for running the build.

### 32. Let's say there is a broken build in the Jenkins project, then what can be done?

Initially, you will have to open the console output where the broken builds are created and then figure out if there are any file changes that were missed. In case there are no issues found there, then you will need to update your local workspace, replicate the problem, and then try to solve it.

33. How to deploy a custom build of a core plugin?

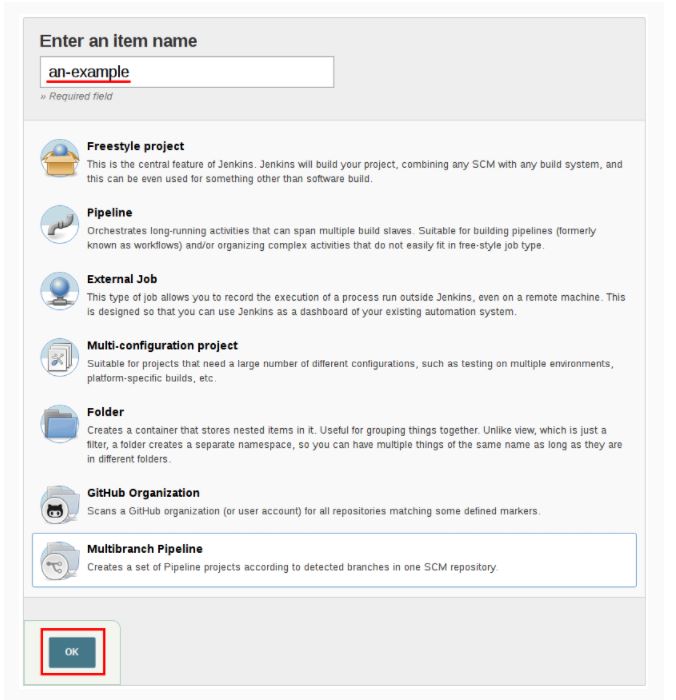
The steps to deploy a custom build of a core plugin are:

* First, copy the .hpi file to $JENKINS\_HOME/plugins
* Then remove the plugin's development directory
* Next, create an empty file called <plugin>.hpi.pinned
* Finally, restart Jenkins and use your custom build of a core plugin

34. What is the process of making a Multibranch Pipeline in Jenkins?

To create a Multibranch Pipeline in Jenkins, follow the following steps:

* Open the Jenkins dashboard and create a new item by clicking on 'new item'
* Enter the project name and, from the options, select 'Multibranch pipeline'
* Click on OK



Source: <https://www.jenkins.io/doc/book/pipeline/multibranch/>

* Then select the repository location, branch source (GitHub/Bitbucket), and add the branch source credentials.
* Save the project
* Now, Jenkins automatically creates new Multibranch Pipelines for repositories
* Then to connect to the GitHub repo, we need the HookURL
* To get this URL from the repository settings, add this HookURL to the Webhooks section
* Once the jobs are created, Jenkins will automatically trigger the build