**1) Explain what is JMeter?**

JMeter is a[Java](https://www.guru99.com/java-tutorial.html)tool, which is used for performance [Load Testing](https://www.guru99.com/load-testing-tutorial.html).

**2) Explain how JMeter works?**

JMeter acts like a group of users sending requests to a target server. It collects response from target server and other statistics which show the performance of the application or server via graphs or tables.

**3) Explain where you can use functions and variables?**

Variables and functions can be written into any field of any test component

**4) Mention what are regular expressions in JMeter?**

Based on the pattern, regular expression are used to search and manipulate text. JMeter is useful in interpreting forms of regular expression or patterns being used throughout a JMeter test plan.

**5) Explain what is Samplers and Thread groups?**

* Thread group: For any test plan, JMeter is the beginning part of thread group elements. It is an important element of JMeter, where you can set number of users and time to load all the users given in the thread group
* Samplers: Sampler generates one or more sample results; these sample results have many attributes like elapsed time, data size, etc. Samplers allow JMeter to send specific types of requests to the server, through samplers, thread group decides which type of request it need to make. Some of the useful samplers are HTTP request, FTP request, JDBC request and so on.

**6) Whether the test plans built using JMeter are OS dependent?**

Usually,[Test Plan](https://www.guru99.com/what-everybody-ought-to-know-about-test-planing.html)are saved in their XML format, so there is nothing to do with any particular O.S. It can be run on any OS where JMeter can run.

**7) Mention what are the types of processor in JMeter?**

The types of processor in JMeter are

* Pre-processor
* Post processor

**8) Explain what are Pre-processor Elements? List some of the pre-processor elements?**

A pre-processor is something that will happen before sampler executes. To configure the sample request prior to its execution or to update variables that are not extracted from response text pre-processor elements are used.

Some of the pre-processor elements are

* HTTP URL re-writing modifier
* HTTP user parameter modifier
* HTML link parser
* BeanShell PreProcessor

**9) Mention the execution order of Test Elements?**

The test plans elements execution order is

* Configuration elements
* Pre-processors
* Timers
* Samplers
* Post-processors
* Assertions
* Listeners

**10) What does “contain” and “matches” indicates in the regular expression?**

In the regular expression, contains indicates that the regular expression matched at least some part of the target. While matches means the regular expression matched the whole target. So, ‘alphabet’ is “matched” by ‘al.\*t.’

**11) Explain what is configuration elements?**

A configuration element works parallel with a Sampler. To set up defaults and variables for later use by samplers configuration elements can be used. At the start of the scope, these elements are processed before any samplers in the same scope.

**12) Explain what is a timer in JMeter and what are the types of it?**

A JMeter thread by default will send requests continuously without any pause. To get a pause between the request, Timers are used. Some of the Timers used are Constant Timer, Gaussian Random Timer, Synchronizing Timer, Uniform Random Timer and so on.

**13) Explain what is Test Fragment?**

Test fragment is also a type of element like Thread Group element. The only difference is test fragment is not implemented unless it is referenced by either a Module controller or an Include controller.

**14) Explain what is Assertion in JMeter? What are the types of assertion?**

Assertion helps to verify that your server under test returns the expected results

Some commonly used Assertion in JMeter are

* Response Assertion
* Duration Assertion
* Size Assertion
* XML Assertion
* HTML Assertion

**15) Explain how you can reduce the resource requirement in JMeter?**

To reduce the resource requirements in JMeter

* Use non-GUI mode: jmeter –n –t test.jmx –l test.jtl
* During the load, test doesn’t use “view results tree” or “view results in table” listeners, use them only during scripting phase
* Don’t use functional mode
* Instead of using lots of similar samplers, use the same sampler in loop and use variable to vary the sample

**16) Explain how you can perform spike testing in JMeter?**

By synchronizing, timer JMeter spike[Testing](https://www.guru99.com/software-testing.html)can be achieved. Synchronizing timer, blocks thread until a specific amount of threads has been blocked and then release them all together thus creating large instantaneous load.

**17) Explain how you can capture the script of the authentication window in JMeter?**

Normally, you can capture script by recording.

* First you have to Threadgroup in Testplan and then make HTTPProxyServer in Workbench
* After that, set port number in Global Setting box (e.g., 8911) and modify your connection setting in IE as local host in address 8911 as in port Then you can start http proxy server in JMeter and run your application for login

**18) List out few JMeter Listeners?**

Some of the JMeter Listeners are

* Spline Visualizer
* Aggregate Report
* View Result Tree
* View Result in Table
* Monitor Results
* Distribution Graph
* BeanShell Listener
* Summary Report and so on

**19) What is distributed load testing? How it can be achieved?**

Distributed load testing is the process through which numerous systems can be used for simulating load of a large number of users. By using the master-slave configuration, JMeter can do distribute load testing.

**20) In JMeter is it necessary to call embedded resources explicitly?**

You can eliminate all embedded resources from being explicitly called. Requests have a checkbox at the bottom that says “retrieve embedded resources.” It would grab all CSS, JPG, etc. It is a brilliant way to find resources and broken link in a web App.

**21) Explain what is the role of Timer in JMeter?**

With the help of timer, JMeter can delay the time between each request, which a thread makes. It can solve the overload problem of the server.

**22) Explain what is Post-processor?**

To perform any action after making a request, Post- processor is used. For example, if JMeter sends an http request to the web server, and if you want JMeter to stop sending the request if the web server shows an error, then you will use post-processor to perform this action.

**23) What are the benefits that JMeter offers for performance testing?**

JMeter offers benefits on [Performance Testing](https://www.guru99.com/performance-testing.html) like

* It can be used to test performance for both, static resources as well as dynamic resources
* It can handle a maximum number of concurrent users then your website can handle
* It provides the graphical analyses of performance reports

**Q #1) Explain the architecture of Jmeter.**

**Ans:** Jmeter is a Java based open source application that is basically designed for the purpose of Load Testing.

It supports all major protocols that are supported in Load Runner. Unlike any browser, Jmeter works on levels of protocols and does not execute JavaScript present in HTML web pages.

**Q #2) Does Jmeter simulate actual browser behavior?**

**Ans:**No, Jmeter does not support the actual browser behavior. It does not render the HTML webpages as the normal browser does. Response can be viewed in HTML format but the actual timings are not present in the generated samples.

**Q #3) What is Distributed testing?**

**Ans:** Distributed Testing means using multiple machines for load testing in which one of the machine can be made master and others can be kept as slave. It is very important to note that all the machines should be on the  same network and should have the same version of Java and Jmeter

**Q #4) What is the use of Regular Expression in Jmeter?**

**Ans:** Regular Expression can be used for extracting some values dynamically from the responses so as to use to it in the subsequent request or save it for reporting purposes. Regular Expression is used in both Pre-Processors as well as Post Processors.

**Q #5) What are the types of processors in Jmeter?**

**Ans:** Basically there are two types of processors in Jmeter namely Pre-Processor and Post Processor.

Pre-Processors execute before the main sampler and can change the scope of the sampler whereas Post Processors execute after the main sampler and are applicable to all samplers in the same scope of Test Plan. They can be used to extract some fields from the server response and store them in variables.

**Q #6) What are the different ways of Data Parametrization in Jmeter?**

**Ans:** Data Parametrization makes the scripts reusable where the values need not be hardcoded for the same request with different parameters.

**Below is the data parametrization that is supported in Jmeter:**

* CSV Data Set Config
* User Defined Variables.

**Q #7) What are the maximum recommended threads on a single system?**

**Ans:** It depends on the hardware configuration of your system which includes a processor, JVM, allocated memory -Xmx, etc.

Other factors which impact this are the number of components in your test plan i.e. the number of config elements or processors and it also depends on whether you are using GUI/Non-GUI Mode.

**Q #8) Explain the difference between Gaussian and Poisson Timers.**

**Ans:** Both Gaussian and Poisson Timers work on a mathematical formula with some constant Delay and additional offset. Difference between the two lies in the fact how the lambda value is calculated in case of Poisson timer and how deviation is calculated in case of Gaussian Timer.

**Q #9) What are the major differences between Jmeter and Load Runner.**

**Ans:** Jmeter is considered as the major competitor of Load Runner in the  industry but there are some differences between the two.

**Mentioned below are some of the major differences:**

| **Load Runner** | **Jmeter** |
| --- | --- |
| Licensed Software | Open Source tool. |
| Developed by Mercury | Developed by Apache. |
| UI is very impressive | It lacks in UI |
| It has more technical capabilities. | Less technically sound as compared to Load Runner. |
| Supports SAP, Siebel and Peoplesoft. | Doesn’t support SAP and Siebel |

**Q #10) What is the use of co-relation in Jmeter?**

**Ans:** Co-relation is a process in which values can be extracted from the server response and stored in a variable and then can be used in any other request which is to follow.

**For Example,** for testing any login functionality if you have to use session ID/cookie ID , you can extract the value from the response of GET request of the login page and then dynamically use the same while making POST request for login.

**Q #11) What are the different types of listeners?**

**Ans:**Listeners are used for storing the execution results of load testing in different forms be it a table, graph, tree or in any other presentable format so that it can be presented to the client. There are different type of inbuild listeners in Jmeter and many others can be imported into it by using plugins as per the requirement.

**Below are some of the inbuild listeners:**

* View Results in Table
* View Results Tree
* Graph Results
* Aggregate Graph
* Aggregate Report
* Assertion Results
* Response Time Graph

**Q #12) Explain the flow of Test Script Recorder.**

**Ans:** HTTP(s) Test Script Recorder is used to record all the http(s) request going to the server from your application. There are some configurations which need to be done in Jmeter in order to make it work.

**Below are the steps followed to record https traffic:**

* Add HTTP(s) Test Script Recorder to WorkBench.
* Enter the port number which you want to start your proxy server from.
* Choose the Target either as “Workbench” only or add a Recording Controller in your test plan and select the same as Target so that all the recordings are stored under it.
* Start the Proxy Server now.
* Configure your browser with manual proxy settings pointing to the same port number used in the test script recorder.

**Q #13) Can Jmeter record actions from Mobile? If yes, how?**

**Ans:** Yes, Jmeter can record HTTP or https request going to the server from your mobile application also. Mobile and Jmeter should be on the same network.

**Below is the configuration required:**

* Configure your proxy server in JMeter to run at a specified port.
* Set up a proxy on your mobile wifi settings and enter the same port number that is used in the recorder.
* Install the Root CA certificate on your mobile.
* Hit server request from your mobile and observe it getting captured by the specified controller.

**Q #14) How to do master-slave configuration in Jmeter?**

**Ans:** Master-Slave configuration is a part of Distributed Testing in which more than one machine is used to perform load testing of the server under test.

It is very important that all machines are on the same network and all have the same version of Jmeter. In distributed testing, one machine is made as the Master and the others are kept as slaves by doing some configurations.

**The process is specified below:**

* On master machine, edit the jmeter.properties file and add the IP addresses of slave machines against the remote\_host field in the file.
* Save the file and open the Jmeter again.
* Now from the RUN menu in Jmeter, select Remote Start and choose the IP of the machine to be invoked.
* Choose RUN menu and select Remote Start all to start all the slave machines for your testing.

**Q #15) What are the Jmeter supported protocols?**

**Ans:** **Jmeter supports various standard protocols as listed below:**

* HTTP/HTTPs
* SOAP
* LDAP
* FTP
* SMTP
* TCP

**Q #16) Explain the Syntax of Jmeter variables and functions.**

**Ans:** Just as in any other programming language, variables and functions are used in Jmeter also in order to make the scripts reusable.

**Syntax of Variable** – ${var}

There are many inbuilt functions that are available in JMeter to perform various actions. Function string can be generated from the Function Dialogue Box itself.

**For example,** if you want to get the machine IP stored in a machineIP variable, you can use the string ${\_\_machineIP(machineIP)}.

**Q #17) Why is it recommended to run Jmeter in GUI mode?**

**Ans:** Jmeter tests can be run both GUI as well as Non-GUI Mode.

It is highly recommended to run the load test in Non-GUI mode because AWT Event Thread can kill the tests in case of high load scenarios.

**There are various Non-GUI mode supported with Jmeter such as:**

* Command Line
* ANT Plugin
* MAVEN Plugin
* Jenkins

**Q #18) Is it possible to run Selenium scripts in Jmeter? If yes, how?**

**Ans:**Yes, it is possible to run selenium scripts in Jmeter to get some ideas on their performance.

There are two ways of doing it. Either you can use Junit libraries to build Selenium scripts and save as Jars and copy the same in Jmeter directory. And then add Junit sampler to your test plan and import the Jar file.

Otherwise, Webdriver sampler plugin can be added in the JMeter ext folder and then restart the Jmeter. Write your selenium code in the Webdriver sampler and then execute to see the performance.

**Q #19) How do you manage sessions and cookies in Jmeter?**

**Ans:** Sessions and cookies can be managed in Jmeter by using config elements such as HTTP Cache Manager which provides an option to clear the cookies in every iteration and also allows to add user-defined cookies.

HTTP Cache manager helps you in clearing cache after each iteration as per your requirement in the load tests and also limits the number of elements which can be stored in the cache. Both of these config elements can be attached to the HTTP sampler.

**Q #20) What are the important steps for testing JDBC request?**

**Ans:** JDBC Requests are used to establish a connection with the databases and then measure the response time of the queries.

**Important steps for testing JDBC requests are:**

* **Setting up Config Element** – JDBC Connection Configuration in which Database URL and JDBC Driver Class needs to be added as per the database which is being used. Also, add the variable name for this connection configuration so as to use it in sampler.
* Add JDBC Request, add the same variable name added above and write your queries to test.

**Q #21) What is BeanShell scripting?**

**Ans:** BeanShell is a lightweight Java scripting that is used in JMeter to perform some complex task.

BeanShell sampler can perform various functions with the use of coding. You can print the thread number, get the current sampler executed, fetch the cookies etc.

**Q #22) Can Jmeter measure the performance of a complete application? For example, you have multiple screens in your mobile app. Can Jmeter measure the time taken to flip the screens?**

**Ans:** No, Jmeter does not measure the transition time between the screens. It can only measure the server actions not the UI interactions.

**Q #23) What is a Root CA Certificate?**

**Ans:** HTTPS connection requires a certificate to authenticate the connections which get established when the browser hits the web server.

Jmeter generates it temporarily to intercept the SSL traffic in order to record the actions. For recording actions via mobile, you need to have this certificate in your mobile to record the actions.

**Q #24) Which factors decide the maximum threads that one should generate per system?**

**Ans:** It depends on the hardware of the system.

**For Example,** on a 2-3 GHz CPU, 400-600 threads can be generated. It also depends on the components in your test plan. More the processors and XML parsing elements, more the CPU load and hence less threads.  For high load, it is recommended to use multiple machines for load testing.

**Q #25) What is a Workbench and why is it required?**

**Ans:** Workbench is a storage area for adding some components which can be added to test plan if required.

Components of workbench does not get saved with the test plan automatically. They have to be saved separately as test fragments. Most important part of the Workbench is HTTP(s) Test script recorder which is highly useful in recording the https request and later load can be applied on them to measure the response time.

**Q #26) What is Tidy Parsing?**

**Ans:**Tidy Parsing is a type of parsing that is used in Xpath extractor.

If the response is in pure XML then tidy parsing is not required whereas, in case of XHTML, it is mandatory to check the tidy parsing option in order to fetch the correct results.

**Q #27) What are the important plugins that are supported in Jmeter?**

**Ans:** Jmeter supports different types of plugins which are helpful in generating high quality results.

**Below are the major plugins that are supported:**

* Thread Group Plugin – Stepping Thread Group Plugin.
* Samplers Plugins like Webdriver.
* Listeners plugins.

**Q #28) What are the types of controller in Jmeter?**

**Ans:** Controllers are used in Jmeter to control the flow of execution of requests.

**Below are the controllers that are used in JMeter:**

* Recording Controller
* IF Controller
* While Controller
* Transaction Controller
* Loop Controller
* Simple Controller
* Module Controller

**Angular4+**

**Q1. Explain component decorators in Angular4.**

A decorator is the core concept when developing an angular framework with version 2 and above. It may become a core language feature for JavaScript soon. In angular 4, decorators are used extensively and are also used to compile a code. There are 4 different types of decorators:

* Class decorators
* Property decorators
* Method decorators
* Parameter decorators

A decorator is a function that is invoked with a prefixed “@” symbol and is immediately followed by a class, parameter, method, or property. A decorator returns the same thing which was given as an input but in an augmented form.

**Q2. Write the CLI command to generate a component in Angular4.**

Components are just simple classes which are declared as components with the help of component decorators.

It becomes easy to create an application which already works, with the help of angular CLI commands. “Ng generate” is used to generate components, routes, services, and pipes. Simple test shells are also created with the help of this CLI command. For generating a component in angular4 with the help of CLI command, you need to follow the following syntax-

* ng generate component component\_name;

It generates the component and adds the component to module declarations.

**Q3. Explain the component directory structure of angular4.**

Here are the elements which are present in the component directory structure anf modules: –

* module.ts- in this, the angular module is declared. @NgModule decorator is used which initializes the different aspects of angular applications. AppComponent is also declared in it.
* components.ts- it simply defines the components in angular and this is the place where the app-root sector is also defined. A title attribute is also declared in the component.
* component.html- it is the template file of the application which represents the visual parts of our components.

**Q4.  Explain ngFor directive with an example.**

The ngFor directive instantiates a template for every element of the given iterator. The different local variables of the ngFor directive can be used in iterations. The ngFor directive can even be used with the HTML elements. It also performs various changes in DOM. Several exported values can be aliased to local variables with the help of ngFor directive. It allows us to build data presentation lists and tables in our HTML templates. Here’s an example of ngFor directive with the help of HTML:

<tr \*ngFor=”hero of heroes”>

<td>({hero.name})</td></tr>

**Q5. Explain property binding or one way binding in angular js.**

Basically, property binding means passing data from the component class and setting the value of a given element in the view. It is a one-way binding in which the data is passed from component to the class. It allows us to control the element property values from component to class. Property binding in angular can take place by three ways:

* Interpolation can be used to define a value, as long as the value being defined is a string.
* Wrapping brackets around the element property and binding it to the component property is the most common type of property binding.
* The third way is by adding “bind” before the element property.

**Q6. Explain ngIf directive with an example.**

The ngIf is a built-in template directive which is used to add or remove some parts of DOM. This addition or removal depends on the expression being true or false.

If the expression is evaluated to false, then the ngIf directive removes the HTML element. If the expression is evaluated to be true, then the element gets added to the DOM.

Syntax- \*ngIf=”<condition>”

Example- <ul \*ngFor=”let person of people”

\*ngIf=”person.age < 30”>

<li>{{person.name}}</li></ul>

**Q7. Write the difference between directive and component in angular js.**

In angular js, there are differences between the meta-data annotations. Some of the differences are:

* A directive is used to add behavior to an existing element. Whereas, a component is used to create a component with attached behavior.
* “@directive” is used to create a directive. Whereas, “@component” is used to create a component.
* A directive is used to attach different behaviors to an existing DOM element. Whereas, with the help of component, we can break our application into smaller components.
* A directive is used to create reusable behavior. Whereas, a component is used to create reusable components.
* A directive does not require a view. Whereas, a component needs a view via @view.

**Prepare Latest**[**React js Interview questions**](https://www.onlineinterviewquestions.com/react-js-interview-questions/)**2018**

**Q8. What do you understand by Isolated Unit Tests?**

As the name implies, unit test is all about testing individual units of code. In order to answer some questions, isolating the unit of code under test is really important. When we do this, we are not forced into creating related pieces such as DOM elements for sorting. With the help of isolated unit tests, it becomes easier to implement everything. To simulate the requests, dependency injections are also provided. The individual sort function can be tested in isolation. And not only the sort function, any function can be tested in isolation.

**Q9. What is a routing in angular js?**

ngRoute module is used when you want to navigate through different pages of your application but you also want your application to be a single page application. This ngRoute module navigates through different pages of your application without reloading the entire application. The angular js route module should be included to make your application ready for routing. The ngRoute is added as a dependency in the application. The routing engine captures the specific URL requested by the user and renders the view based on the defined routing rules.

**Q10. What do you understand by services with reference to angular js?**

Services in angular js are used to organize and share code across your application. These are the suitable objects which are wired together with the help of dependency injection. The angular js services are lazily instantiated. The service is only instantiated by angular js only when the application component depends on it. In angular js, new services can be made or can even be used in other built-in services. Over 30 built-in services are present in angular js.

**11. What are DSL Animation Functions in Angular js. List them.**

DSL Animation functions are used for crafting animations in Angular js. Below are list of DSL Animation functions in angular js.

* trigger()
* state()
* transition()
* group()
* sequence()
* style()
* animate()
* keyframes()

**Docker:**

## **Docker Interview Questions**

Docker containers can be considered as the new era of virtualization. Docker containers are replacing Virtual Machines, this is a well known fact now. So be prepared to answer few questions on Docker in your next interview. If you aspire to outshine yourself in your upcoming DevOps interview, going through these top Docker interview questions is a must. These questions are very carefully handpicked after discussing with many interviewers and experts. Curious to know more about Docker [***check out this Docker blog series***](https://www.edureka.co/blog/docker-tutorial).

This Docker Interview Questions blog is a part of parent blog [***DevOps Interview Questions***](https://www.edureka.co/blog/interview-questions/top-devops-interview-questions-2016/). It includes all the DevOps Stages.

Before moving ahead, you may go through the webinar recording of DevOps Interview Questions where our instructor has shared his experience and expertise on Docker that will help you to answer any Docker question in your Interview:

## **DevOps Interview Questions and Answers | DevOps Training | Edureka**

First question in this blog on Docker Interview Questions has to be:

### **Q1. What is Docker?**

I will suggest you to start with a small definition of Docker.

Docker is a containerization platform which packages your application and all its dependencies together in the form of containers so as to ensure that your application works seamlessly in any environment be it development or test or production.

Now you should explain Docker containers.

Docker containers, wrap a piece of software in a complete filesystem that contains everything needed to run: code, runtime, system tools, system libraries etc. anything that can be installed on a server. This guarantees that the software will always run the same, regardless of its environment.

You can refer the diagram shown below, as you can see that containers run on a single machine share the same operating system kernel, they start instantly as only apps need to start as the kernel is already running and uses less RAM.

**Note: Unlike Virtual Machines which has its own OS Docker containers uses the host OS**

As you have mentioned about Virtual Machines in your previous answer, so the next question in this Docker Interview Questions blog will be related to the differences between the two.

### **Q2. What are the differences between Docker and Hypervisors?**

You can refer the below differences:

Next set of Docker interview questions will focus on various components of Docker.

### **Q3. What is Docker image?**

I will suggest you to go with the below mentioned flow:

Docker image is the source of Docker container. In other words, Docker images are used to create containers. Images are created with the build command, and they’ll produce a container when started with run. Images are stored in a Docker registry such as registry.hub.docker.com because they can become quite large, images are designed to be composed of layers of other images, allowing a minimal amount of data to be sent when transferring images over the network.  
**Tip: Be aware of Dockerhub in order to answer questions on pre-available images.**

### **Q4. What is Docker container?**

This is a very important question so just make sure you don’t deviate from the topic and I will advise you to follow the below mentioned format:

Docker containers include the application and all of its dependencies, but share the kernel with other containers, running as isolated processes in user space on the host operating system. Docker containers are not tied to any specific infrastructure: they run on any computer, on any infrastructure, and in any cloud.  
Now explain how to create a Docker container, Docker containers can be created by either creating a Docker image and then running it or you can use Docker images that are present on the Dockerhub.

Docker containers are basically runtime instances of Docker images.

### **Q5 What is Docker hub?**

Answer to this question is pretty direct.

Docker hub is a cloud-based registry service which allows you to link to code repositories, build your images and test them, stores manually pushed images, and links to Docker cloud so you can deploy images to your hosts. It provides a centralized resource for container image discovery, distribution and change management, user and team collaboration, and workflow automation throughout the development pipeline.

[**Learn Docker With DevOps Now**](https://www.edureka.co/devops/)

### **Q6. How is Docker different from other container technologies?**

According to me, below, points should be there in your answer:

Docker containers are easy to deploy in a cloud. It can get more applications running on the same hardware than other technologies, it makes it easy for developers to quickly create, ready-to-run containerized applications and it makes managing and deploying applications much easier. You can even share containers with your applications.  
If you have some more points to add you can do that but make sure the above the above explanation is there in your answer.

### **Q7. What is Docker Swarm?**

You should start this answer by explaining Docker Swarn.

Docker Swarm is native clustering for Docker. It turns a pool of Docker hosts into a single, virtual Docker host. Docker Swarm serves the standard Docker API, any tool that already communicates with a Docker daemon can use Swarm to transparently scale to multiple hosts.  
I will also suggest you to include some supported tools:

* Dokku
* Docker Compose
* Docker Machine
* Jenkins

### **Q8. What is Dockerfile used for?**

This answer, according to me should begin by explaining the use of Dockerfile.

Docker can build images automatically by reading the instructions from a Dockerfile.

Now I will suggest you to give a small definition of Dockerfle.

A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image. Using docker build users can create an automated build that executes several command-line instructions in succession.

Now, the next set of Docker interview questions will test your experience with Docker.

### **Q9. Can I use json instead of yaml for my compose file in Docker?**

You can use json instead of yaml for your compose file, to use json file with compose, specify the filename to use for eg:  
**docker-compose -f docker-compose.json up**

### **Q10. Tell us how you have used Docker in your past position?**

Explain how you have used Docker to help rapid deployment. Explain how you have scripted Docker and used Docker with other tools like Puppet, Chef or Jenkins.

If you have no past practical experience in Docker and have past experience with other tools in a similar space, be honest and explain the same. In this case, it makes sense if you can compare other tools to Docker in terms of functionality.

### **Q11. How to create Docker container?**

I will suggest you to give a direct answer to this.

We can use Docker image to create Docker container by using the below command:

|  |  |
| --- | --- |
| 1 | docker run -t -i command name |

This command will create and start a container.

You should also add, If you want to check the list of all running container with the status on a host use the below command:

|  |  |
| --- | --- |
| 1 | docker ps -a |

### **Q12. How to stop and restart the Docker container?**

In order to stop the Docker container you can use the below command:

|  |  |
| --- | --- |
| 1 | docker stop container ID |

Now to restart the Docker container you can use:

|  |  |
| --- | --- |
| 1 | docker restart container ID |

### **Q13 How far do Docker containers scale?**

Large web deployments like Google and Twitter, and platform providers such as Heroku and dotCloud all run on container technology, at a scale of hundreds of thousands or even millions of containers running in parallel.

### **Q14. What platforms does Docker run on?**

### I will start this answer by saying Docker runs on only Linux and Cloud platforms and then I will mention the below vendors of Linux:

Ubuntu 12.04, 13.04 et al

* Fedora 19/20+
* RHEL 6.5+
* CentOS 6+
* Gentoo
* ArchLinux
* openSUSE 12.3+
* CRUX 3.0+

Cloud:

* Amazon EC2
* Google Compute Engine
* Microsoft Azure
* Rackspace

**Note that Docker does not run on Windows or Mac.**

### **Q15. Do I lose my data when the Docker container exits?**

You can answer this by saying, no I won’t lose my data when Docker container exits, any data that your application writes to disk gets preserved in its container until you explicitly delete the container. The file system for the container persists even after the container halts.

**SonarQube:**

**What is SonarQube  
A:**Sonar is a web based code quality analysis tool for Maven based Java projects. It covers a wide area of code quality check points which include: Architecture & Design, Complexity, Duplications, Coding Rules, Potential Bugs, Unit Test etc.  
  
**Q: Why use SonarQube ?  
A:**Sonar covers the 7 sections of code quality

* Architecture and Design
* Unit tests
* Duplicated code
* Potential bugs
* Complex code
* Coding standards
* Comments

**Q: What is role of database in SonarQube?  
A:** Sonar uses a Derby or H2 as default database. When running Sonar, it says that these databases may only be used for evaluation. We can change this default database and use our custom DB.  
  
**Q: How to create reports in SonarQube?  
A:** To create reports using SonarQube

mvn clean install

mvn sonar:sonar -Dsonar.issuesreport.html.enable=true

**SPRING BOOT :**

**Q: What is Spring Boot?**

**A:**Over the years spring has become more and more complex as new functionalities have been added. Just visit the page-<https://spring.io/projects> and we will see all the spring projects we can use in our application for different functionalities. If one has to start a new spring project we have to add build path or add maven dependencies, configure application server, add spring configuration . **So a lot of effort is required to start a new spring project as we have to currently do everything from scratch. Spring Boot is the solution to this problem**. Spring boot has been built on top of existing spring framework. Using spring boot we avoid all the boilerplate code and configurations that we had to do previously. Spring boot thus helps us use the existing Spring functionalities more robustly and with minimum efforts.  
[More details and miscellaneous examples](http://www.javainuse.com/spring/sprboot)   
  
**Q: What are advantages of Spring Boot ?  
A:** The advantages of Spring Boot are

* Reduce Developement, Testing time and efforts.
* Use of JavaConfig helps avoid usage of XML.
* Avoid lots of maven imports and the various version conflicts.
* Provide Opinionated Development approach.
* Quick start to development by providing defaults.
* No Separate Web Server Needed.Which means that you no longer have to boot up Tomcat, Glassfish, or anything else.
* Requires less configuration-Since there is no web.xml file. Simply add classes annotated with@Configuration and then you can add methods annotated with@Bean, and Spring will automagically load up the object and manage it like it always has. You can even add @Autowired to the bean method to have Spring autowire in dependencies needed for the bean.
* Environment Based Configuration-Using these properties, you can pass into the application which environment you are using with:-Dspring.profiles.active={enviornment}. Spring will then load up the subsequent application properties file at (application-{environment}.properties) after loading up the main application properties file.

**Q: Which build tool have you used to develop Spring Boot Application ?  
A:**Spring Boot application can be developed using Maven as well as Gradle.   
[Spring Boot application using Maven](http://www.javainuse.com/spring/SpringBoot_HelloWorld)   
[Spring Boot application using Gradle](http://www.javainuse.com/spring/SpringBoot_HelloWorld_gradle)   
  
**Q: What is JavaConfig?  
A:**Spring JavaConfig is a product of the Spring community that provides a pure-Java approach to configuring the Spring IoC Container. It thus helps avoid using XML configurations. The advantages of using JavaConfig are  
The advantages of JavaConfig are

* Object-oriented configuration. Because configurations are defined as classes in JavaConfig, users can take full advantage of object-oriented features in Java. One configuration class may subclass another, overriding its @Bean methods, etc.
* Reduced or eliminated XML configuration. The benefits of externalized configuration based on the principles of dependency injection have been proven. However, many developers would prefer not to switch back and forth between XML and Java. JavaConfig provides developers with a pure-Java approach to configuring the Spring container that is conceptually similar to XML configuration. It is technically possible to configure the container using only JavaConfig configuration classes, however in practice many have found it ideal to mix-and-match JavaConfig with XML.
* Type-safe and refactoring-friendly. JavaConfig provides a type-safe approach to configuring the Spring container. Thanks to Java 5.0's support for generics, it is now possible to retrieve beans by type rather than by name, free of any casting or string-based lookups.

[Added security configuration without xml using java config.](http://www.javainuse.com/spring/sprboot_sec)

**Q:How to reload my changes on Spring Boot without having to restart server?  
A:**This can be achieved using DEV Tools. With this dependency any changes you save, the embedded tomcat will restart. Spring Boot has a Developer tools (DevTools) module which helps to improve the productivity of developers. One of the key challenge for the Java developers is to auto deploy the file changes to server and auto restart the server. Developers can reload changes on Spring Boot without having to restart my server. This will eliminates the need for manually deploying the changes every time. Spring Boot doesn’t have this feature when it has released it’s first version. This was a most requested features for the developers. The module DevTools does exactly what is needed for the developers. This module will be disabled in the production environment. It also provides H2-database console for better testing the application. The following dependency is used

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<optional>true</optional>

</dependency>

The DevTool dependency usage for autorestart and H2 DB console is illustrated in this [example](http://www.javainuse.com/spring/SpringBootUsingPagination)  
  
**Q:What is Actuator in Spring Boot?  
A:**Spring boot actuator is one of the important feature in spring boot framework. Spring boot actuator helps you to access the current state of the running application in production environment. There are several metrics that has to be checked and monitored in the production environment. Even some external applications may be using those services to trigger the alert message to concerned person. Actuator module exposes set of REST endpoints that can be directly accessed as a HTTP URL to check the status.  
  
**Q:How to disable Actuator endpoint security in Spring Boot?  
A:** By default all sensitive HTTP endpoints are secured such that only users that have an ACTUATOR role may access them. Security is enforced using the standard HttpServletRequest.isUserInRole method.  
We can disable security using -   
**management.security.enabled=false**  
It is suggested to disable security only if the actuator endpoints are accessed behind firewall.  
  
**Q: How to run Spring boot application to custom port ?  
A:**In order to run a spring boot application on a custom port you can specify the port in application.properties.  
**server.port=8090**  
  
**Q: Have you written Test cases using Spring Boot ?  
A:**Spring Boot provides the @SpringBootTest for writing Unit Test Cases  
[Spring Boot Unit Test Simple Example](http://www.javainuse.com/spring/springboot_testcases)  
  
**Q: What is YAML ?  
A:**YAML is a human-readable data serialization language. It is commonly used for configuration files.  
Compared to properties file, YAML file is much more structured and less confusing in case we want to add complex properties in the configuration file. As can be seen YAML has hierarchical configuration data.   
[Use YAML properties in Spring Boot](http://www.javainuse.com/spring/bootyaml)

**Q: How to implement security for Spring boot application ?  
A:**For Implementing security for Spring Boot we use the **spring-boot-starter-security dependency** and have to add the Security config. It requires very little code. Config class will have to extend WebSecurityConfigurerAdapter and override its methods.  
[Spring Boot Security example and explanation](http://www.javainuse.com/spring/sprboot_sec)   
  
**Q: Have you integrated Spring Boot and ActiveMQ ?  
A:**For integrating Spring Boot and ActiveMQ we use the **spring-boot-starter-activemq dependency**  
It requires very little configuration and no boilerplate code.  
[Spring Boot ActiveMQ explanation](http://www.javainuse.com/spring/sprboot_activemq)   
  
**Q: Have you integrated Spring Boot and Apache Kafka ?  
A:**For integrating Spring Boot and Apache Kafka we use the **spring-kafka** dependency.  
[Spring Boot + Apache Kafka Example](http://www.javainuse.com/spring/spring-boot-apache-kafka-hello-world)   
  
**Q: How to implement Pagination and Sorting with Spring Boot ?  
A:**Using Spring Boot achieving pagination is very simple. Using the Spring Data-JPA this is achieved passing pageable org.springframework.data.domain.Pageable to the repository methods.  
[Spring Boot pagination explanation](http://www.javainuse.com/spring/SpringBootUsingPagination)  
  
**Q: What is Swagger ? Have you implemented it using Spring Boot ?  
A:**Swagger is widely used for visualizing APIs, and with Swagger UI it provides online sandbox for frontend developers. For the tutorial, we will use the Springfox implementation of the Swagger 2 specification. Swagger is a tool, a specification and a complete framework implementation for producing the visual representation of RESTful Web Services. It enables documentation to be updated at the same pace as the server. When properly defined via Swagger, a consumer can understand and interact with the remote service with a minimal amount of implementation logic. Thus Swagger removes the guesswork in calling the service.  
[Spring Boot + Swagger2](http://www.javainuse.com/spring/boot_swagger)   
  
**Q: What is Spring Profiles ? How do you implement it using Spring Boot ?  
A:**Spring Profiles allows users to register beans depending on the profile(dev, test, prod etc). So when the application is running in DEVELOPMENT only certain beans can be loaded and when in PRODUCTION certain other beans can be loaded. Suppose our requirement is that the Swagger documentation be enabled only for the QA environment and disabled for all others. This can be done using Profiles. Spring Boot makes using Profiles very easy.   
[Spring Boot + profiles](http://www.javainuse.com/spring/boot_swagger_profile)   
  
**Q: What is Spring Batch ? How do you implement it using Spring Boot ?  
A:**Spring Boot Batch provides reusable functions that are essential in processing large volumes of records, including logging/tracing, transaction management, job processing statistics, job restart, skip, and resource management. It also provides more advanced technical services and features that will enable extremely high-volume and high performance batch jobs though optimization and partitioning techniques.Simple as well as complex, high-volume batch jobs can leverage the framework in a highly scalable manner to process significant volumes of information.   
[Spring Boot + Batch](http://www.javainuse.com/spring/bootbatch)   
  
**Q: What is FreeMarker Template? How do you implement it using Spring Boot ?  
A:**FreeMarker is a Java-based Template Engine, originally focusing on dynamic web page generation with MVC software architecture. The major advantage of using Freemarker is the complete separation of the Presentation layer and the Business Layer. The Programmers can work on the application code while the designers can work on the html page design. Finally using freemarker these can then be combined to give the final output page.   
[Spring Boot + FreeMarker Example](http://www.javainuse.com/spring/spring-boot-freemarker-hello-world)   
  
**Q: How to implement Exception Handling using Spring Boot ?  
A:**Spring provides a very useful way to handle exceptions using ControllerAdvice.   
We will be implementing a ControlerAdvice class which will handle all exceptions thrown by the controller class.  
[Spring Boot Exception Handling](http://www.javainuse.com/spring/boot-exception-handling)  
  
**Q: What is caching? Have you used any caching framework with Spring Boot ?  
A:**A cache is an area of local memory that holds a copy of frequently accessed data that is otherwise expensive to get or compute. Have used Hazelcast for caching.  
[Spring Boot + Hazelcast Example](http://www.javainuse.com/spring/spring-boot-hazelcast)  
  
**Q: Have you exposed a SOAP webservice endpoint using Spring Boot?  
A:** Yes. Using Spring Boot exposed a web service to be consumed. Used Contract first approach to generate the classes from wsdl.   
[Spring Boot + SOAP Web Service Example](http://www.javainuse.com/spring/springbootsoapwebservice)   
  
**Q: How did you perform database operations using Spring Boot ?  
A:**[Spring Boot Tutorial-Spring Data JPA](http://www.javainuse.com/spring/SpringBoot_DataJPA)  
[Spring Boot JDBC Example](http://www.javainuse.com/spring/bootjdbc)   
  
**Q: How to upload a file using Spring ?  
A:**[Spring Boot + File Upload Example](http://www.javainuse.com/spring/bootupload)

**Q: How to implement interceptors with Spring Boot ?  
A:**[Using Spring MVC HandlerInterceptor with Spring Boot](http://www.javainuse.com/spring/bootInterceptor)   
  
**Q: How to use schedulers with Spring Boot ?  
A:**[Spring Boot Task Scheduler Example](http://www.javainuse.com/spring/bootTask)   
  
**Q: Which all starter maven dependencies have you used ?  
A:**Have used different starter dependencies like **spring-boot-starter-activemq dependency**, **spring-boot-starter-security dependency**, **spring-boot-starter-web dependency.**   
This helps in adding less number of dependencies and also in reducing version conficts.   
[Spring Boot Security example and explanation](http://www.javainuse.com/spring/sprboot_sec)   
  
**Q: What is CSRF attack? How to enable CSRF protection against it?  
A:**CSRF stands for Cross-Site Request Forgery. It is an attack that forces an end user to execute unwanted actions on a web application in which they are currently authenticated. CSRF attacks specifically target state-changing requests, not theft of data, since the attacker has no way to see the response to the forged request.   
[Spring Boot Security - Enabling CSRF Protection](http://www.javainuse.com/spring/boot_security_csrf)   
  
**Q: How to use Form Login Authentication using Spring Boot?  
A:**[Spring Boot Form Security Login Hello World Example](http://www.javainuse.com/spring/boot_form_security)   
  
**Q: What is OAuth2? How to implement it using Spring Boot?  
A:**[Spring Boot + OAuth2 implementation](http://www.javainuse.com/spring/spring-boot-oauth-introduction)   
  
**Q: What is GZIP?How to implement it using Spring Boot?  
A:** gzip is a file format and a software application used for file compression and decompression.  
[Spring Boot + GZIP Compression](http://www.javainuse.com/spring/boot-zip)  
  
**Q: Have you used any integration framework with Spring Boot?  
A:** Have integrated Apache Camel with Spring Boot. Made use of Apache Camel Spring Boot starter dependency.[Spring Boot +Apache Camel](http://www.javainuse.com/spring/bootcamel)  
  
**Q: What is Apache Freemarker? When to use it instead of JSP? How to integrate it with Spring Boot?  
A:**JSP is tailor made for Web pages, Freemarker template is a more generic templating language - it can be used to generate html, plain text, emails, etc.  
[Spring Boot + FreeMarker Example](http://www.javainuse.com/spring/spring-boot-freemarker-hello-world)  
  
**Q: When will you use WebSockets? How tto implement it using Spring Boot?  
A:**WebSocket is a computer communications protocol, providing full-duplex communication channels over a single TCP connection.

* **WebSocket are bi-directional -**Using WebSocket either client or server can initiate sending a message.
* **WebSocket are Full Duplex -**The client and server communication is independent of each other.
* **Single TCP connection -**The initial connection is using HTTP, then this connection gets upgraded to a socket based connection. This single connection is then used for all the future communication
* **Light -**The WebSocket message data exchange is much lighter compared to http.

[Spring Boot + WebSockets Example](http://www.javainuse.com/spring/boot-websocket)  
  
**Q: What is AOP? How to use it with Spring Boot?  
A:**During software development, functions that span multiple points of an application are called cross-cutting concerns. These cross-cutting concerns differ from the application’s main business logic. Hence ,separating these cross-cutting concerns from the business logic is where aspect-oriented programming (AOP) comes into picture.  
[Spring Boot + AOP Example](http://www.javainuse.com/spring/spring-boot-aop)   
  
**Q: What is Apache Kafka? How to integrate it with Spring Boot?  
A:** Apache Kafka is a distributed publish-subscribe messaging system. It is a scalable, fault-tolerant, publish-subscribe messaging system which enables us to build distributed applications. It is an Apache Top Level project. Kafka is suitable for both offline and online message consumption.  
[Spring Boot + Apache Kafka Example](http://www.javainuse.com/spring/spring-boot-apache-kafka-hello-world)   
  
**Q: How can we monitor all the Spring Boot Microservices?  
A:**Spring Boot provides actuator endpoints to monitor metrics of individual microservices. These endpoints are very helpful for getting information about applications like if they are up, if their components like database etc are working good. But a major drawback or difficulty about using actuator enpoints is that we have to individually hit the enpoints for applications to know their status or health. Imagine microservices involving 50 applications, the admin will have to hit the actuator endpoints of all 50 applications. To help us deal with this situation, we will be using open source project located at [https://github.com/codecentric/spring-boot-admin](http://www.javainuse.com/).   
Built on top of Spring Boot Actuator, it provides a web UI to enable us visualize the metrics of multiple applications.  
[Spring Boot Admin](http://www.javainuse.com/spring/boot-admin)   
  
**Q: Have you used any Spring Cloud Components with Spring Boot?  
A:** Have used Spring Cloud components like Netflix Eureka for Service Registration,Ribbon for Load Balancing.  
[Spring Boot + Cloud Components](http://www.javainuse.com/spring/springcloud)  
[Spring Cloud interview Questions](http://www.javainuse.com/spring/spring-cloud-interview-questions)

**Micro Services:**

**Q: What is Spring Boot?  
A:**Over the years spring has become more and more complex as new functionalities have been added. Just visit the page-<https://spring.io/projects> and we will see all the spring projects we can use in our application for different functionalities. If one has to start a new spring project we have to add build path or add maven dependencies, configure application server, add spring configuration . **So a lot of effort is required to start a new spring project as we have to currently do everything from scratch. Spring Boot is the solution to this problem**. Spring boot has been built on top of existing spring framework. Using spring boot we avoid all the boilerplate code and configurations that we had to do previously. Spring boot thus helps us use the existing Spring functionalities more robustly and with minimum efforts.  
[More details and miscellaneous examples](http://www.javainuse.com/spring/sprboot)   
[Spring Boot Interview](http://www.javainuse.com/spring/SpringBootInterviewQuestions)

**Q: What are Microservices ?  
A:** Microservices is a variant of the service-oriented architecture (SOA) architectural style that structures an application as a collection of loosely coupled services. In a microservices architecture, services should be fine-grained and the protocols should be lightweight. The benefit of decomposing an application into different smaller services is that it improves modularity and makes the application easier to understand, develop and test. It also parallelizes development by enabling small autonomous teams to develop, deploy and scale their respective services independently. It also allows the architecture of an individual service to emerge through continuous refactoring. Microservices-based architectures enable continuous delivery and deployment.  
  
**Q: What is Spring Cloud ?  
A:** Spring Cloud Stream App Starters are Spring Boot based Spring Integration applications that provide integration with external systems. Spring Cloud Task. A short-lived microservices framework to quickly build applications that perform finite amounts of data processing.  
[Spring Cloud](http://www.javainuse.com/spring/springcloud)   
  
**Q: What are the advantages of using Spring Cloud ?  
A:** When developing distributed microservices with Spring Boot we face the following issues-

* **Complexity associated with distributed systems-**  
  This overhead includes network issues, Latency overhead, Bandwidth issues, security issues.
* **Service Discovery-**  
  Service discovery tools manage how processes and services in a cluster can find and talk to one another. It involves a directory of services, registering services in that directory, and then being able to lookup and connect to services in that directory.
* **Redundancy-**  
  Redundancy issues in distributed systems.
* **Loadbalancing-**  
  Load balancing improves the distribution of workloads across multiple computing resources, such as computers, a computer cluster, network links, central processing units, or disk drives.
* **Performance issues-**  
  Performance issues due to various operational overheads.
* **Deployment complexities-**  
  Requirement of Devops skills.

**Q: How will you monitor multiple microservices for various indicators like health?  
A:** Spring Boot provides actuator endpoints to monitor metrics of individual microservices. These endpoints are very helpful for getting information about applications like if they are up, if their components like database etc are working good. But a major drawback or difficulty about using actuator enpoints is that we have to individually hit the enpoints for applications to know their status or health. Imagine microservices involving 50 applications, the admin will have to hit the actuator endpoints of all 50 applications. To help us deal with this situation, we will be using open source project located at <https://github.com/codecentric/spring-boot-admin>.   
Built on top of Spring Boot Actuator, it provides a web UI to enable us visualize the metrics of multiple applications.  
[Spring Boot Admin Example](http://www.javainuse.com/spring/boot-admin)  
  
**Q: What does one mean by Service Registration and Discovery ? How is it implemented in Spring Cloud  
A:** When we start a project, we usally have all the configurations in the properties file. As more and more services are developed and deployed, adding and modifying these properties become more complex. Some services might go down, while some the location might change. This manual changing of properties may create issues.  
Eureka Service Registration and Discovery helps in such scenarios. As all services are registered to the Eureka server and lookup done by calling the Eureka Server, any change in service locations need not be handled and is taken care of  
[Microservice Registration and Discovery with Spring cloud using Netflix Eureka.](http://www.javainuse.com/spring/spring_eurekaregister)   
  
**Q: What does one mean by Load Balancing ? How is it implemented in Spring Cloud  
A:** In computing, load balancing improves the distribution of workloads across multiple computing resources, such as computers, a computer cluster, network links, central processing units, or disk drives. Load balancing aims to optimize resource use, maximize throughput, minimize response time, and avoid overload of any single resource. Using multiple components with load balancing instead of a single component may increase reliability and availability through redundancy. Load balancing usually involves dedicated software or hardware, such as a multilayer switch or a Domain Name System server process.  
In SpringCloud this can be implemented using Netflix Ribbon.  
[Spring Cloud- Netflix Eureka + Ribbon Simple Example](http://www.javainuse.com/spring/spring_ribbon)   
  
**Q: How to achieve server side load balancing using Spring Cloud?  
A:** Server side load balancingcan be achieved using Netflix Zuul.   
Zuul is a JVM based router and server side load balancer by Netflix.   
It provides a single entry to our system, which allows a browser, mobile app, or other user interface to consume services from multiple hosts without managing cross-origin resource sharing (CORS) and authentication for each one. We can integrate Zuul with other Netflix projects like Hystrix for fault tolerance and Eureka for service discovery, or use it to manage routing rules, filters, and load balancing across your system.  
[Spring Cloud- Netflix Zuul Example](http://www.javainuse.com/spring/spring-cloud-netflix-zuul-tutorial)   
  
**Q: In which business scenario to use Netflix Hystrix ?  
A:** **Hystrix is a latency and fault tolerance library designed to isolate points of access to remote systems, services and 3rd party libraries, stop cascading failure and enable resilience in complex distributed systems where failure is inevitable.**  
Usually for systems developed using Microservices architecture, there are many microservices involved. These microservices collaborate with each other.  
Consider the following microservices-

Suppose if the microservice 9 in the above diagram failed, then using the traditional approach we will propagate an exception. But this will still cause the whole system to crash anyways.  
This problem gets more complex as the number of microservices increase. The number of microservices can be as high as 1000. This is where hystrix comes into picture-  
We will be using two features of Hystrix-

* Fallback method
* Circuit Breaker

[Spring Cloud- Netflix Eureka + Ribbon + Hystrix Simple Example](http://www.javainuse.com/spring/spring_hystrix)   
  
**Q: What is Spring Cloud Bus? Need for it?  
A:** Consider the scenario that we have multiple applications reading the properties using the Spring Cloud Config and the Spring Cloud Config in turn reads these properties from GIT.  
Consider the below example where multiple employee producer modules are getting the property for Eureka Registration from Employee Config Module.

What will happen if suppose the eureka registration property in GIT changes to point to another Eureka server. In such a scenario we will have to restart the services to get the updated properties. There is another way of using Actuator Endpoint **/refresh**. But we will have to individually call this url for each of the modules. For example if Employee Producer1 is deployed on port 8080 then call **http://localhost:8080/refresh**. Similarly for Employee Producer2 **http://localhost:8081/refresh** and so on. This is again cumbersome. This is where Spring Cloud Bus comes into picture.

The Spring Cloud Bus provides feature to refresh configurations across multiple instances. So in above example if we refresh for Employee Producer1, then it will automatically refresh for all other required modules. This is particularly useful if we are having multiple microservice up and running. This is achieved by connecting all microservices to a single message broker. Whenever an instance is refreshed, this event is subscribed to all the microservices listening to this broker and they also get refreshed. The refresh to any single instance can be made by using the endpoint **/bus/refresh**  
[Spring Cloud Tutorial - Publish Events Using Spring Cloud Bus](http://www.javainuse.com/spring/cloud-stream-bus)   
  
**Q: What is Spring Cloud Data Flow? Need for it?  
A:** Spring Cloud Data Flow is a toolkit to build real-time data integration and data processing pipelines by establishing message flows between Spring Boot applications that could be deployed on top of different runtimes.

Long lived applications require Stream Applications while Short lived applications require Task Applications.  
In this example we make use of Stream Applications. Previously we had already developed Spring Cloud Stream applications to understand the concept of [Spring Cloud Stream Source](http://www.javainuse.com/spring/cloud-stream-rabbitmq-1) and [Spring Cloud Sink](http://www.javainuse.com/spring/cloud-stream-rabbitmq-2) and their benefit.

Pipelines consist of Spring Boot apps, built using the Spring Cloud Stream or Spring Cloud Task microservice frameworks. SCDF can be accessed using the REST API exposed by it or the web UI console.  
We can make use of metrics, health checks, and the remote management of each microservice application Also we can scale stream and batch pipelines without interrupting data flows. With SCDF we build data pipelines for use cases like data ingestion, real-time analytics, and data import and export. SCDF is composed of the following Spring Projects-

**Spring Cloud:**

**Q: What is Spring Cloud?  
A:** Spring Cloud Stream App Starters are Spring Boot based Spring Integration applications that provide integration with external systems. Spring Cloud Task. A short-lived microservices framework to quickly build applications that perform finite amounts of data processing.  
[Spring Cloud](http://www.javainuse.com/spring/springcloud)   
  
**Q: What are the advantages of using Spring Cloud?  
A:** When developing distributed microservices with Spring Boot we face the following issues-

* **Complexity associated with distributed systems-**  
  This overhead includes network issues, Latency overhead, Bandwidth issues, security issues.
* **Service Discovery-**  
  Service discovery tools manage how processes and services in a cluster can find and talk to one another. It involves a directory of services, registering services in that directory, and then being able to lookup and connect to services in that directory.
* **Redundancy-**  
  Redundancy issues in distributed systems.
* **Loadbalancing-**  
  Load balancing improves the distribution of workloads across multiple computing resources, such as computers, a computer cluster, network links, central processing units, or disk drives.
* **Performance issues-**  
  Performance issues due to various operational overheads.
* **Deployment complexities-**  
  Requirement of Devops skills.

**Q: What does one mean by Service Registration and Discovery? How is it implemented in Spring Cloud?  
A:** When we start a project, we usally have all the configurations in the properties file. As more and more services are developed and deployed, adding and modifying these properties become more complex. Some services might go down, while some the location might change. This manual changing of properties may create issues.  
Eureka Service Registration and Discovery helps in such scenarios. As all services are registered to the Eureka server and lookup done by calling the Eureka Server, any change in service locations need not be handled and is taken care of  
[Microservice Registration and Discovery with Spring cloud using Netflix Eureka.](http://www.javainuse.com/spring/spring_eurekaregister)   
  
**Q: What does one mean by Load Balancing ? How is it implemented in Spring Cloud?  
A:** In computing, load balancing improves the distribution of workloads across multiple computing resources, such as computers, a computer cluster, network links, central processing units, or disk drives. Load balancing aims to optimize resource use, maximize throughput, minimize response time, and avoid overload of any single resource. Using multiple components with load balancing instead of a single component may increase reliability and availability through redundancy. Load balancing usually involves dedicated software or hardware, such as a multilayer switch or a Domain Name System server process.  
In SpringCloud this can be implemented using Netflix Ribbon.  
[Spring Cloud- Netflix Eureka + Ribbon Simple Example](http://www.javainuse.com/spring/spring_ribbon)

**Q: What is Hystrix? How does it implement Fault Tolerance?  
A:** In computing, load balancing improves the distribution of workloads across multiple computing resources, such as computers, a computer cluster, network links, central processing units, or disk drives. Load balancing aims to optimize resource use, maximize throughput, minimize response time, and avoid overload of any single resource. Using multiple components with load balancing instead of a single component may increase reliability and availability through redundancy. Load balancing usually involves dedicated software or hardware, such as a multilayer switch or a Domain Name System server process.  
In SpringCloud this can be implemented using Netflix Ribbon.  
[Spring Cloud- Netflix Eureka + Ribbon Simple Example](http://www.javainuse.com/spring/spring_ribbon)   
  
**Q: What is Hystrix? How does it implement Fault Tolerance?  
A:** **Hystrix is a latency and fault tolerance library designed to isolate points of access to remote systems, services and 3rd party libraries, stop cascading failure and enable resilience in complex distributed systems where failure is inevitable.**  
Usually for systems developed using Microservices architecture, there are many microservices involved. These microservices collaborate with each other.  
Consider the following microservices-

Suppose if the microservice 9 in the above diagram failed, then using the traditional approach we will propagate an exception. But this will still cause the whole system to crash anyways.  
This problem gets more complex as the number of microservices increase. The number of microservices can be as high as 1000. This is where hystrix comes into picture  
We will be the Fallback method feature of Hystrix for this scenario. We have two services employee-consumer consuming the service exposed by the employee-producer.  
The simplified diagram is as below-

Now suppose due to some reason the employee-producer exposed service throws an exception. In this case using Hystrix we define a fallback method. This fallback method should have the same return type as the exposed service. In case of exception in the exposed service the fallback method will return some value. [Spring Cloud- Netflix Hystrix Fallback method Simple Example](http://www.javainuse.com/spring/spring_hystrix)

**Q: What is Hystrix Circuit Breaker? Need for it?  
A:** Due to some reason the employee-producer exposed service throws an exception. In this case using Hystrix we defined a fallback method. In case of exception in the exposed service the fallback method returned some default value.

If the exceptions keep on occuring in the firstPage method() then the Hystrix circuit will break and the employee consumer will skip the firtsPage method all together and directly call the fallback method.

The purpose of circuit breaker is to give time to the first page method or other methods that the firstpage method might be calling and is causing the exception to recover. It might happen that on less load the issue causing the exceptions have better chance of recovering

[Spring Cloud- Circuit Breaker using Netflix Hystrix Simple Example](http://www.javainuse.com/spring/spring_hystrix_circuitbreaker)   
  
**Q: What is Netflix Feign? What are its advantages?  
A:** Feign is a java to http client binder inspired by Retrofit, JAXRS-2.0, and WebSocket. Feign's first goal was reducing the complexity of binding Denominator uniformly to http apis regardless of restfulness. Previous examples in the employee-consumer we consumed the REST services exposed by the employee-producer using **REST Template**

But we had to write a lot of code to perform following-

* For Load balancing using Ribbon.
* Getting the Service instance and then the Base URL.
* Make use of the REST Template for consuming service.

The previous code was as below

@Controller

public class ConsumerControllerClient {

**@Autowired**

**private LoadBalancerClient loadBalancer;**

**public void getEmployee() throws RestClientException, IOException {**

**ServiceInstance serviceInstance=loadBalancer.choose("employee-producer");**

**System.out.println(serviceInstance.getUri());**

**String baseUrl=serviceInstance.getUri().toString();**

baseUrl=baseUrl+"/employee";

RestTemplate restTemplate = new RestTemplate();

ResponseEntity<String> response=null;

try{

response=restTemplate.exchange(baseUrl,

HttpMethod.GET, getHeaders(),String.class);

}catch (Exception ex)

{

System.out.println(ex);

}

System.out.println(response.getBody());

}

The previous code, there are chances of exceptions like NullPointer and is not optimal. We will see how the call is made much easier and cleaner using Netflix Feign. If the Netflix Ribbon dependency is also in the classpath, then Feign also takes care of load balancing by default.  
[Spring Cloud- Netflix Feign Simple Example](http://www.javainuse.com/spring/spring-cloud-netflix-feign-tutorial)   
  
**Q: What is Spring Cloud Bus? Need for it?  
A:** Consider the scenario that we have multiple applications reading the properties using the Spring Cloud Config and the Spring Cloud Config in turn reads these properties from GIT.  
Consider the below example where multiple employee producer modules are getting the property for Eureka Registration from Employee Config Module.

What will happen if suppose the eureka registration property in GIT changes to point to another Eureka server. In such a scenario we will have to restart the services to get the updated properties. There is another way of using Actuator Endpoint **/refresh**. But we will have to individually call this url for each of the modules. For example if Employee Producer1 is deployed on port 8080 then call **http://localhost:8080/refresh**. Similarly for Employee Producer2 **http://localhost:8081/refresh** and so on. This is again cumbersome. This is where Spring Cloud Bus comes into picture.

The Spring Cloud Bus provides feature to refresh configurations across multiple instances. So in above example if we refresh for Employee Producer1, then it will automatically refresh for all other required modules. This is particularly useful if we are having multiple microservice up and running. This is achieved by connecting all microservices to a single message broker. Whenever an instance is refreshed, this event is subscribed to all the microservices listening to this broker and they also get refreshed. The refresh to any single instance can be made by using the endpoint **/bus/refresh**

**Apache Kafka:**

## **1. What is Kafka?**

Wikipedia defines Kafka as “an open-source message broker project developed by the [Apache Software Foundation](https://en.wikipedia.org/wiki/Apache_Software_Foundation) written in Scala, where the design is heavily influenced by transaction logs”. It is essentially a distributed publish-subscribe messaging system.

## **2. List the various components in Kafka.**

The four major components of Kafka are:

* Topic – a stream of messages belonging to the same type
* Producer – that can publish messages to a topic
* Brokers – a set of servers where the publishes messages are stored
* Consumer – that subscribes to various topics and pulls data from the brokers.

## **3. Explain the role of the offset.**

Messages contained in the partitions are assigned a unique ID number that is called the offset. The role of the offset is to uniquely identify every message within the partition.

## **4. What is a Consumer Group?**

Consumer Groups is a concept exclusive to Kafka.  Every Kafka consumer group consists of one or more consumers that jointly consume a set of subscribed topics.

## **5. What is the role of the ZooKeeper?**

Kafka uses Zookeeper to store offsets of messages consumed for a specific topic and partition by a specific Consumer Group.

## **6. Is it possible to use Kafka without ZooKeeper?**

No, it is not possible to bypass Zookeeper and connect directly to the Kafka server. If, for some reason, ZooKeeper is down, you cannot service any client request.

## **7. Explain the concept of Leader and Follower.**

Every partition in Kafka has one server which plays the role of a Leader, and none or more servers that act as Followers. The Leader performs the task of all read and write requests for the partition, while the role of the Followers is to passively replicate the leader. In the event of the Leader failing, one of the Followers will take on the role of the Leader. This ensures load balancing of the server.

## **8. What roles do Replicas and the ISR play?**

Replicas are essentially a list of nodes that replicate the log for a particular partition irrespective of whether they play the role of the Leader. On the other hand, ISR stands for In-Sync Replicas. It is essentially a set of message replicas that are synced to the leaders.

## **9. Why are Replications critical in Kafka?**

Replication ensures that published messages are not lost and can be consumed in the event of any machine error, program error or frequent software upgrades.

## **10. If a Replica stays out of the ISR for a long time, what does it signify?**

It means that the Follower is unable to fetch data as fast as data accumulated by the Leader.

## **11. What is the process for starting a Kafka server?**

Since Kafka uses ZooKeeper, it is essential to initialize the ZooKeeper server, and then fire up the Kafka server.

* To start the ZooKeeper server: > bin/zookeeper-server-start.sh config/zookeeper.properties
* Next, to start the Kafka server: > bin/kafka-server-start.sh config/server.properties

## **12. How do you define a Partitioning Key?**

Within the Producer, the role of a Partitioning Key is to indicate the destination partition of the message. By default, a hashing-based Partitioner is used to determine the partition ID given the key. Alternatively, users can also use customized Partitions.

## **13. In the Producer, when does QueueFullException occur?**

QueueFullException typically occurs when the Producer attempts to send messages at a pace that the Broker cannot handle. Since the Producer doesn’t block, users will need to add enough brokers to collaboratively handle the increased load.

## **14. Explain the role of the Kafka Producer API.**

The role of Kafka’s Producer API is to wrap the two producers – kafka.producer.SyncProducer and the kafka.producer.async.AsyncProducer. The goal is to expose all the producer functionality through a single API to the client.

## **15. What is the main difference between Kafka and Flume?**

Even though both are used for real-time processing, Kafka is scalable and ensures message durability.

These are some of the frequently asked Apache Kafka interview questions with answers. You can brush up on your knowledge of Apache Kafka with [these](https://www.edureka.co/blog/category/apache-kafka/)blogs.

**. Compare Kafka & Flume**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Kafka** | **Flume** |
| Data flow | Pull | Push |
| Hadoop Integration | Loose | Tight |
| Functionality | Publish-subscribe model messaging system | System for data collection, aggregation & movement |

**2. Which are the elements of Kafka?**

The most important elements of Kafka:

* Topic – It is the bunch of similar kind of messages
* Producer – using this one can issue communications to the topic
* Consumer – it endures to a variety of topics and takes data from brokers.
* Brokers – this is the place where the issued messages are stored

Get a detailed understanding of Kafka in this comprehensive [Kafka Tutorial](https://intellipaat.com/tutorial/big-data-and-hadoop-tutorial/the-ultimate-goal-of-this-tutorial/)

**3. What role ZooKeeper plays in a cluster of Kafka?**

Kafka is an open source system and also a distributed system is built to use Zookeeper. The basic responsibility of Zookeeper is to build coordination between different nodes in a cluster. Since Zookeeper works as periodically commit offset so that if any node fails, it will be used to recover from previously committed to offset.

The ZooKeeper is also responsible for configuration management, leader detection, detecting if any node leaves or joins the cluster, synchronization, etc.

Bottom of Form

**4. What is Kafka?**

Kafka is a message divider project coded in Scala. Kafka is originally developed by LinkedIn and developed as an open sourced in early 2011. The purpose of the project is to achieve the best stand for conducting the real-time statistics nourishment.

Learn “[What is Kafka](https://intellipaat.com/blog/what-is-apache-kafka/)?” in this insightful blog.

**5. Why do you think the replications are dangerous in Kafka?**

Duplication assures that issued messages which are available are absorbed in the case of any appliance mistake, plan fault or recurrent software promotions.

**6. What major role a Kafka Producer API plays?**

It is responsible for covering the two producers- kafka.producer.SyncProducer and the kafka.producer.async.AsyncProducer. The main aim is to disclose all the producer performance through a single API to the clients.

**7. Distinguish between the Kafka and Flume?**

Flume’s major use-case is to gulp down the data into Hadoop. The Flume is incorporated with the [Hadoop’s monitoring system](https://intellipaat.com/tutorial/big-data-and-hadoop-tutorial/the-hadoop-module-high-level-architecture/), file formats, file system and utilities such as Morphlines. Flume’s design of sinks, sources and channels mean that with the aid of Flume one can shift data among other systems lithely, but the main feature is its Hadoop integration.

The Flume is the best option used when you have non-relational data sources if you have a long file to stream into the Hadoop.Kafka’s major use-case is a distributed publish- subscribe messaging system. Kafka is not developed specifically for Hadoop and using Kafka to read and write data to Hadoop is considerably trickier than it is in Flume.

Kafka can be used when you particularly need a highly reliable and scalable enterprise messaging system to connect many multiple systems like Hadoop.

***Find out***[***in this blog***](https://intellipaat.com/blog/processing-json-data-in-real-time-streaming-using-storm-kafka/)***how Kafka is used to process real time JSON Data***

**8. Describe partitioning key?**

Its role is to specify the target divider of the memo, within the producer. Usually, a hash-oriented divider concludes the divider ID according to the given factors. Consumers also use the tailored Partitions.

**9. Inside the manufacturer, when does the QueueFullException emerge?**

QueueFullException naturally happens when the manufacturer tries to propel communications at a speed which Broker can’t grip. Consumers need to insert sufficient brokers to collectively grip the amplified load since the Producer doesn’t block.

**10. Can Kafka be utilized without Zookeeper?**

It is impossible to use [Kafka without Zookeeper](https://intellipaat.com/tutorial/kafka-tutorials/kafka-configuration/) because it is not feasible to go around Zookeeper and attach in a straight line to the server. If the Zookeeper is down for a number of causes, then we will not be able to serve any customer demand.

**11. Elaborate Kafka architecture.**

A cluster contains multiple brokers since it is a distributed system. Topic in the system will get divided into multiple partitions and each broker store one or more of those partitions so that multiple producers and consumers can publish and retrieve messages at the same time.

**12. How to start a Kafka server?**

Given that Kafka exercises Zookeeper, we have to start the Zookeeper’s server.  
Learn more in this [Zookeeper Tutorial now](https://intellipaat.com/tutorial/hbase-tutorial/installation/).

One can use the convince script packaged with Kafka to get a crude but effective single node Zookeeper instance> bin/zookeeper-server-start.shconfig/zookeeper.propertiesNow the Kafka server can start> bin/Kafka-server-start.shconfig/server.properties

**13. What are consumers or users?**

Kafka provides single consumer abstractions that discover both queuing and publish-subscribe Consumer Group. They tag themselves with a user group and every communication available on a topic is distributed to one user case within every promising user group. User instances are in disconnected process. We can determine the messaging model of the consumer based on the consumer groups.

* If all consumer instances have the same consumer set, then this works like a conventional queue adjusting load over the consumers.
* If all customer instances have dissimilar consumer groups, then this works like a publish-subscribe and all messages are transmitted to all the consumers.

**14. Describe an Offset?**

The messages in the partitions will be given a sequential ID number known as an offset, the offset will be used to identify each message in the partition uniquely. With the aid of Zookeeper Kafka stores the offsets of messages consumed for a specific topic and partition by this consumer group.

**15. What do you know about partitioning key?**

A partition key can be precise to point to the aimed division of a communication, in Kafka producer. Usually, a hash-oriented divider concludes the division id with the input and people uses modified divisions also.

**16. Why is Kafka technology significant to use?**

Kafka being distributed publish-subscribe system has the advantages as below.Fast: Kafka comprises of a broker and a single broker can serve thousands of clients by handling megabytes of reads and writes per second.Scalable: facts are partitioned and streamlined over a cluster of machines to enable large informationDurable: Messages are persistent and is replicated in the cluster to prevent record loss Distributed by Design: It provides fault tolerance guarantees and robust.

**Jenkins:**

## **Jenkins Interview Questions**

Just commit changes to the SCR (Source Code Repository) and Jenkins can automate the rest of the process for you with the help of plugins. So that makes it a very important DevOps tool. There is a high possibility that you encounter many Jenkins questions if you go for a DevOps job interview. Below are the most frequently asked Jenkins interview questions. I have collected these questions after doing a lot of research and after discussing with some DevOps experts who are directly involved in the hiring process. Curious to know more about Jenkins [***check out this Jenkins blog series***](https://www.edureka.co/blog/what-is-jenkins/).

This Jenkins Interview Questions blog is a part of parent blog [***DevOps Interview Questions***](https://www.edureka.co/blog/interview-questions/top-devops-interview-questions-2016/). It includes all the DevOps Stages.

First question in this Jenkins Interview Questions blog has to be:

## **DevOps Interview Questions and Answers | DevOps Training | Edureka**

### **Q1. What is Jenkins?**

My suggestion is to start this answer by giving a definition of Jenkins.

Jenkins is an open source automation tool written in Java with plugins built for Continuous Integration purpose. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. It also allows you to continuously deliver your software by integrating with a large number of testing and deployment technologies.

Once you have defined Jenkins give an example, you can refer the below mentioned use case:

* First, a developer commits the code to the source code repository. Meanwhile, the Jenkins server checks the repository at regular intervals for changes.
* Soon after a commit occurs, the Jenkins server detects the changes that have occurred in the source code repository. Jenkins will pull those changes and will start preparing a new build.
* If the build fails, then the concerned team will be notified.
* If built is successful, then Jenkins deploys the built in the test server.
* After testing, Jenkins generates a feedback and then notifies the developers about the build and test results.
* It will continue to check the  source code repository for changes made in the source code and the whole process keeps on repeating.

Interviewer now knows what is Jenkins but why we use it, there are many other CI tools as well, so why Jenkins?, the next question in this Jenkins interview questions will deal with that answer.

### **Q2. What are the benefits of using Jenkins?**

I will suggest you to include the following benefits of Jenkins, if you can recall any other benefit apart from the below mentioned points you can include that as well.

* At integration stage, build failures are cached.
* For each change in the source code an automatic build report notification is generated.
* To notify developers about build report success or failure, it is integrated with LDAP mail server.
* Achieves continuous integration agile development and test driven development.
* With simple steps, maven release project is automated.
* Easy tracking of bugs at early stage in development environment than production.

Interviewer: Okay Jenkins looks like a really cool tool, but what are the requirements for using Jenkins?

### **Q3. What are the pre-requisites for using Jenkins?**

Answer to this is pretty straightforward To use Jenkins you require:

* A source code repository which is accessible, for instance, a Git repository.
* A working build script, e.g., a Maven script, checked into the repository.

Remember, you have mentioned Plugins in your previous answer, so next question in this Jenkins interview questions blog will be regarding Plugins.

### **Q4. Mention some of the useful plugins in Jenkins?**

Below I have mentioned some important Plugins:

* Maven 2 project
* Git
* Amazon EC2
* HTML publisher
* Copy artifact
* Join
* Green Balls

These Plugins I feel are the most useful plugins, if you want to include any other Plugin that is not mentioned above, you can add that as well, but make sure you first mention the above stated plugins and then add your own.

### **Q5. Mention what are the commands you can use to start Jenkins manually?**

For this answer I will suggest you to go with the below mentioned flow:

To start Jenkins manually open Console/Command line, then go to your Jenkins installation directory. Over there you can use the below commands:

To start Jenkins: **jenkins.exe start**  
To stop Jenkins: **jenkins.exe stop**  
To restart Jenkins: **jenkins.exe restart**

### **Q6. Explain how you can set up Jenkins job?**

My approach to this answer will be to first mention how to create Jenkins job.

Go to Jenkins top page, select “New Job”, then choose “Build a free-style software project”.

Now you can tell the elements of this freestyle job:

* 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..

### **Q7. Explain how to create a backup and copy files in Jenkins?**

Answer to this question is really direct.

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. You can also copy a job directory to clone or replicate a job or rename the directory.

### **Q8. How will you secure Jenkins?**

The way I secure Jenkins is mentioned below, if you have any other way to do it than mention that:

* Ensure global security is on.
* Ensure that Jenkins is integrated with my company’s user directory with appropriate plugin.
* Ensure that matrix/Project matrix is enabled to fine tune access.
* Automate the process of setting rights/privileges in Jenkins with custom version controlled script.
* Limit physical access to Jenkins data/folders.
* Periodically run security audits on same.

I hope you have enjoyed the above set of Jenkins interview questions, the next set of questions will be more challenging, so be prepared.

### **Q9 Explain how you can deploy a custom build of a core plugin?**

Below are the steps to deploy a custom build of a core plugin:

* Stop Jenkins.
* Copy the custom HPI to **$Jenkins\_Home/plugins**.
* Delete the previously expanded plugin directory.
* Make an empty file called **<plugin>.hpi.pinned**.
* Start Jenkins.

### **Q10. What is the relation between Hudson and Jenkins?**

You can just say Hudson was the earlier name and version of current Jenkins. After some issue, the project name was changed from Hudson to Jenkins.

### **Q11. What you do when you see a broken build for your project in Jenkins?**

There can be multiple answers to this question I will approach this task in the following way:

I will open the console output for the broken build and try to see if any file changes were missed. If I am unable to find the issue that way, then I will clean and update my local workspace to replicate the problem on my local and try to solve it.

If you do it in a different way then just mention that in your answer.

### **Q12. Explain how you can move or copy Jenkins from one server to another?**

I will approach this task by copying the jobs directory from the old server to the new one. There are multiple ways to do that, I have mentioned it below:

You can:

* Move a job from one installation of Jenkins to another by simply copying the corresponding job directory.
* Make a copy of an existing job by making a clone of a job directory by a different name.
* Rename an existing job by renaming a directory. Note that if you change a job name you will need to change any other job that tries to call the renamed job.

### **Q13. What are the various ways in which build can be scheduled in Jenkins?**

You can schedule a build in Jenkins in the following ways:

* By source code management commits
* After completion of other builds
* Can be scheduled to run at specified time ( crons )
* Manual Build Requests

### **Q14. What is the difference between Maven, Ant and Jenkins?**

Maven and Ant are Build Technologies whereas Jenkins is a continuous integration tool.

### **Q15. Which SCM tools Jenkins supports?**

Below are Source code management tools supported by Jenkins:

* AccuRev
* CVS,
* Subversion,
* Git,
* Mercurial,
* Perforce,
* Clearcase
* RTC

Now, the next set of Jenkins interview questions will test your experience with Jenkins.

### **Q16. What are the two components Jenkins is mainly integrated with?**

According to me Jenkins is mainly integrated with the following:

* Version Control system like GIT,SVN.
* Build tools like Apache Maven.

### **1) What is Jenkins?**

Jenkins is an open source continuous integration tool written in Java. It keeps a track on version control system and to initiate and monitor a build system if changes occur.

### **2) What is the difference between Maven, Ant and Jenkins?**

The most basic difference is:

Maven and Ant are Build Technologies whereas Jenkins is a continuous integration tool.

### **3) Which SCM tools does Jenkins support?**

Jenkins supports the following SCM tools:

* AccuRev
* CVS
* Subversion
* Git
* Mercurial
* Perforce
* Clearcase
* RTC

### **4) What is continuous integration in Jenkins?**

In software development, multiple developers or teams work on different segments of same web application so you have to perform integration test by integrating all modules. In order to do that an automated process for each piece of code is performed on daily bases so that all your codes get tested. This process is known as continuous integration.

### **5) What is the relation between Hudson and Jenkins?**

Hudson was the earlier name and version of current Jenkins. After some issue, the project name was changed from Hudson to Jenkins.

### **6) What is the requirement for using Jenkins?**

For using Jenkins, you have to need a source code repository which is accessible. For example, a Git repository and a working build script, e.g., a Maven script, checked into the repository.

### **7) What are the advantages of Jenkins?**

Advantage of Jenkins includes:

* Bugs tracking are easy at early stage in development environment.
* Provides a large numbers of plugin support.
* Iterative improvement to the code.
* Build failures are cached at integration stage.
* For each code commit changes an automatic build report notification generates.
* To notify developers about build report success or failure, it is integrated with LDAP mail server.
* Achieves continuous integration agile development and test driven development.
* With simple steps, maven release project is automated.

### **8) How to make sure that your project builds doesn?t break in Jenkins?**

You must follow these steps to make sure that your project builds doesn?t break in Jenkins:

* First, perform successful clean install on your local machine with all unit tests.
* Check all your code changes.
* Synchronize with repository to make sure that all required config and POM changes and any difference is checked into the repository.

### **9) How can you move or copy Jenkins from one server to another?**

Follow these steps to move or copy Jenkins from one server to another:

* First, copy the related job directory and slide a job from one installation of Jenkins to another.
* Make a copy of an already existing job by making clone of a job directory by a different name.
* Renaming an existing job by rename a directory.

### **10) Which commands can be used to start Jenkins manually?**

You can use any one of the following commands to start Jenkins manually:

* (Jenkins\_url)/restart: Forces a restart without waiting for builds to complete.
* (Jenkin\_url)/safeRestart: Allows all running builds to complete.

### **11) What are the most useful plugins in Jenkins?**

Some most useful plugins in Jenkins:

* Maven 2 project
* Amazon EC2
* HTML publisher
* Copy artifact
* Join
* Green Balls

### **12) How to create a backup and copy files in Jenkins?**

If you want to create a back-up of your Jenkins setup, just copy the directory that saves all the setting, build artifacts and logs of Jenkins in its home directory. You can also copy a job directory to clone or replicate a job or rename the directory.

### **13) How can you clone a Git repository via Jenkins?**

If you want to clone a Git repository via Jenkins, you have to enter the e-mail and user name for your Jenkins system. Switch into your job directory and execute the "git config" command for that.

### **14) How can you setup Jenkins jobs?**

Follow these steps:

* Select new item from the menu.
* After that enter a name for the job and select free-style job.
* Then click OK to create new job in Jenkins.
* The next page enables you to configure your job.

### **15) What are the two components Jenkins is mainly integrated with?**

Jenkins is integrated with these two components:

* Version Control system like GIT,SVN
* And build tools like Apache Maven.

**Nexus:**

**Can I upgrade from 2.X to 3.X?**

* Yes :)

**Will Nexus Repository Manager be unusable while I upgrade?**

* No! Your 2.X instance will stay up while the upgrade runs, and can be tuned if need be while the Upgrade process runs.

**What happens if the Upgrade fails?**

* We’ve engineered the Upgrade path in such a way that your existing instance of 2.X will be unaffected and will go on serving up Nexus goodness just like it was before.
* The contents of your 2.x instance are not modified, making a rollback trivial.
* Information will be logged that you can provide to us which we can use to provide you with the world class support you’ve come to expect from Sonatype.

**What prerequisites will there be?**

* Your 2.X system will need to be upgraded to the newest version of Nexus Repository Manager 2.X before you can run the Upgrade
* We will have a lot of great documentation available to help you through the process. Make sure to read this as we will have lots of tips on how to run the process including tips on tuning the process.

**How long will it take?**

* This will be variable based on how many repos you have, size of the repos, underlying storage, etc…
  + You can run the Upgrade in different configurations, which can be blazingly fast depending on if your hardware supports it.

**When will Upgrade be available?**

* You will be able to Upgrade from 2.X to 3.X starting with Repository Manager 3.1

**Will I be able to Upgrade everything?**

* Formats such as YUM or P2 will not be upgradeable in 3.1, those will come at a later date. You will be able to Upgrade all features currently available in 3.1, check [here](https://support.sonatype.com/hc/en-us/articles/222426568-Nexus-Repository-Manager-Feature-Compatibility-Matrix) for more information on what features will be available.
* Some feature names have changed between 2.X and 3.X

**Will windows upgrades within Nexus 3.x have a real upgrade path via an installer?**

* Yes, via the installer you should be able to upgrade by installing to the same directory.

**Will there be an official Nexus 3 container? Migration supported from existing 2.x to 3.x container?**

* A Docker image is available here: <https://hub.docker.com/r/sonatype/nexus3/> and once 3.1 is out you could run the upgrade between the containers.

**Can Nexus 2 and 3 be run at the same time?**

* Yes, but will need different ports.

**Does nexus 3.0 fully support NPM?**

* NXRM 3.0 supports the majority of npm commands, major differences between v2 and v3 are the additions of scoped and login commands.

**With support for maven, docker, npm, and bower should this be done on a single server instance of Nexus or would you suggest separating them?**

* Single server is fine. We see no reason why you would need to separate them.

**How will staging be different in Nexus3?**

* Too early to say much, but given the use of Orient we have abilities to store metadata much easier.

**Why the addition of blobs? What are the benefits?**

* Additional metadata is available to us by using Orient in tandem with the Blobstore, and some more ideas for the future will benefit from this.

**Blobstore could use the ability (in UI or Rest) to move artifacts or groups of items from one repo to another. We won't be able to do this on the file system anymore right?**

* Correct, we do not support file system moves. We have plans to address the use cases you described, as well, although it may play out a bit differently given our additional capabilities with Orient behind the scenes.

**Does Nexus deduplicate the data it stores?  I.E. upload java 8 JDK to multiple places but it is only stored once.**

* At current time it does not but this is an intended future capability.

**Docker image support esp OpenShift?**

* The current image is built on centos, and is available here: <https://hub.docker.com/r/sonatype/nexus3/>, you can modify the Dockerfile available here if you want to run a specific flavor: <https://github.com/sonatype/docker-nexus3/blob/master/Dockerfile>. I believe the image will run in OpenShift, if you experience anything to the opposite, let us know!

**Is the IQ Server plugin available for IntelliJ now?**

* Yes, it is available.

**Does Nexus 3 support manual uploads when nexus is disconnected from internet?**

* Yes it supports this.

**When dealing with a "raw" type, would you use the Nexus API to do an automated artifact upload? Or use something like maven deploy-file?**

* You may use curl to add items, or do a Maven site deploy such as was done in the past with Site, and in the future the Nexus UI.
* 3.x Roadmap

**Prior versions of Nexus Manager UI have been lacking capability of manually mass-deleting artifacts. You could only delete one artifact at a time. Does the new Nexus 3 UI support mass-selecting and mass-deleting multiple artifacts?**

* Not as of yet. There have been thoughts around this and how to handle it in a way in which allows you the best experience, more to come.

**Nexus 2 was mainly centered with maven repos. We used to heavily rely on the Artifact upload GUI to upload custom artifacts. We did not see this functionality in Nexus 3.x. Any plan to include it?**

* We are currently looking at this and will likely be adding support for uploading via the UI Maven2 components in NXRM 3.1, with other formats coming later.

**Will YUM and DEBIAN repositories be supported on windows hosted instances?**

* YUM is going to improve quite a bit with NXRM 3.X as we are removing some of the obstacles. I can’t confirm yet but it may be able to run on Windows as a result in the future.

**I understand that Nexus 3 is a big re-write, how much compatibility has been retained for plug-ins, APIs, User Tokens?**

* Internal APIs have changed quite a bit, plugins written against NXRM 2.x will likely need to be rewritten as a result.

**Can you please comment on ETA when can we expect the re-envisioned Maven repos?**

* Maven2 repos are currently in NXRM 3.0, and the format itself will stay the same. Over the course of 3.1 and 3.2 we are adding more features such as a replacement for repo targets, group routing.

**Is Nexus Pro 3 out as GA?**

* No, the first bits of Pro will be coming in NXRM 3.1, with more features coming in each forthcoming release.

**Will there be functionality added to the Nexus GUI to allow for a "mass" upload (versus the individual component that is available today)?**

* Nothing planned as of yet, but it’s good feedback.

**When will nexus 3.X final version be released?**

* Software is never finished, so maybe never. We are here to serve our customers and the community needs will ultimately change over time. However, I believe your question is when will NXRM 3.x with all the features from NXRM 2.x be released? If that’s your query, feel free to ping me with the features you are looking for so I can better answer this.