**What are final methods of Object class ?**

Ans -> wait notify and notifyAll method

**What are the disadvantages of Hibernate ?**

1. Hibernate generates lot of SQL statements in runtime based on our mapping , so it is bit **slower than JDBC**.

If we use **JDBC**, then we directly write our queries, so **no extra effort** is required.

1. **Poor performance in Batch processing:**

It advisable to use pure JDBC for batch processing as Hibernate performance is not good in Batch processing.

1. **Not good for small project**

Small project will have less number of tables , introducing entire Hibernate framework will be **overhead**than useful.

4. **Lots of API to learn:**A lot of effort is required to learn Hibernate. So, not very easy to learn hibernate easily

**What will be the exception will occur if we add object of Dog in List<Animal>?**

No Issue it will work fine

**What is difference between Synchronized Hashmap and ConcurrentHashMap?**

Synchronized HashMap：

1. Each method is synchronized using an object level lock. So the get and put methods on synchMap acquire a lock.
2. Locking the entire collection is a performance overhead. While one thread holds on to the lock, no other thread can use the collection.

ConcurrentHashMap was introduced in JDK 5.

1. There is no locking at the object level, The locking is at a much finer granularity. For a ConcurrentHashMap, the locks may be at a hashmap bucket level.
2. The effect of lower level locking is that you can have concurrent readers and writers which is not possible for synchronized collections. This leads to much more scalability.
3. ConcurrentHashMap does not throw a ConcurrentModificationException if one thread tries to modify it while another is iterating over it.

**If we have configured Second level cache and data is changed from Database ? what output will be returned on calling get or load method?**

**What is Inverse keyword/Annotation In Hibernate?**

 The “**inverse**” keyword is always declare in **one-to-many** and **many-to-many**relationship (many-to-one doesn’t has inverse keyword), it means which side is responsible to take care of the relationship.

**What is difference between @JoinColumn and @MappedBy?**

**How different annotation of Spring interact to each other?**

**Can We write any other annotation instead of @Contoller?**

**What are different view resolver?**

Below are the important viewresolvers provided by spring framework:

1. **AbstractCachingViewResolver :** Abstract view resolver that caches views. Often views need preparation before they can be used; extending this view resolver provides caching.
2. **XmlViewResolver :**Implementation of ViewResolver that accepts a configuration file written in XML with the same DTD as Spring’s XML bean factories. The default configuration file is /WEB-INF/views.xml.
3. **ResourceBundleViewResolver :**Implementation of ViewResolver that uses bean definitions in a ResourceBundle, specified by the bundle base name. Typically you define the bundle in a properties file, located in the classpath. The default file name is views.properties.
4. **UrlBasedViewResolver :**Simple implementation of the ViewResolver interface that effects the direct resolution of logical view names to URLs, without an explicit mapping definition. This is appropriate if your logical names match the names of your view resources in a straightforward manner, without the need for arbitrary mappings.
5. **InternalResourceViewResolver :**Convenient subclass of UrlBasedViewResolver that supports InternalResourceView (in effect, Servlets and JSPs) and subclasses such as JstlView and TilesView. You can specify the view class for all views generated by this resolver by using setViewClass(..).
6. **VelocityViewResolver/FreeMarkerViewResolver :**Convenient subclass of UrlBasedViewResolver that supports VelocityView (in effect, Velocity templates) or FreeMarkerView ,respectively, and custom subclasses of them.
7. **ContentNegotiatingViewResolver :**Implementation of the ViewResolver interface that resolves a view based on the request file name or Accept header.

**What are the different Bean Scope of Spring?**

**1) singleton:**Returns a single bean instance per Spring IoC container.

**2) prototype:**Returns a new bean instance each time when requested.

**3) request:**Returns a single instance for every HTTP request call.

**4) session:**Returns a single instance for every HTTP session.

**5) global session:**global session scope is equal as session scope on portlet-based web applications.

If no bean scope is specified in bean configuration file, then it will be by default 'singleton'.

**What is dependency injection?**

* Spring helps in the creation of loosely coupled applications because of **Dependency Injection**.
* In Spring, objects define their associations (dependencies) and do not worry about how they will get those **dependencies**. It is the responsibility of Spring to provide the required dependencies for creating objects.

**For example**: Suppose we have an object Employee and it has a dependency on object Address. We would define a bean corresponding to Employee that will define its dependency on object Address.

When Spring tries to create an Employee object, it will see that Employee has a dependency on Address, so it will first create the Address object (dependent object) and then inject it into the Employee object.

* Inversion of Control (**IOC**) and Dependency Injection (**DI**) are used interchangeably. IOC is achieved through DI. DI is the process of providing the dependencies and IOC is the end result of DI. (**Note:** DI is not the only way to achieve IOC. There are [other ways](https://en.wikipedia.org/wiki/Inversion_of_control#Implementation_techniques) as well.)
* By DI, the responsibility of creating objects is shifted from our application code to the Spring container; this phenomenon is called IOC.
* Dependency Injection can be done by setter injection or constructor injection.

**What happens internally in case of using @Autowire annotation?**

automatically based on the name it will identify the bean and inject based on its type .

Autoeiring is the way of DI .A creaed bean wiil be autowired at field level,method level and constructor level but the best way of autowiring is Constructor autowirin because at the time of object initialization object is autowired.

**What is wildcard?**

Can we use any other keyword instead of extends in case of WildCard?

What is the use of Ribbon and Zuul Proxy ? How Hystrix comes in picture?

How you secure your Rest Webservice in your application?

**Can We throw Error object using Throws Clause in catch block?**

You can throw any type of Throwable object using the keyword throw . It interrupts the method. Anything after the throw statement would not be executed, unless the thrown exception is handled. The exception object is not returnedfrom the method, it is thrown from the method.

**What is difference between war and jar and ear?**

**Jar : Java archive file :** It contains a group of .class file

**War : Web archive file :** It represents a web application and in a single war file it contains all the html, css , js ,xml, jsp, servlet etc all the web component required for a web application. If we maintain a war file project delivery, deployment and transportation will become easy.

**EAR: Enterprise archive file :** It represents enterprise application. In web application we can use only web related technology like Servlet and Jsp but in enterprise application anything related to java j2ee ,we can use like ejb Jms etc

**What is default logging mechanism of Spring boot?**

Logback mechanism

**How we change the server in Spring Boot?**

For changing the embedded server, we will user its spring-boot-starter-undertow.

## Adding Dependencies

<dependency>

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

<artifactId>spring-boot-starter-undertow</artifactId>

</dependency>

spring-boot-starter-web comes with Embedded Tomcat. We need to exclude this dependency.

<dependency>

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

<artifactId>spring-boot-starter-web</artifactId>

<exclusions>

<exclusion>

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

<artifactId>spring-boot-starter-tomcat</artifactId>

</exclusion>

</exclusions>

</dependency>

What is difference between Callable and Runnable?

What is difference between KeySet and Hashset?

What are Rest API? What are different methods of Restful Api?

What is difference between Put and POST?

When Singleton Design Pattern fails?

What is double locking mechanism?

What is the use of intern() method?

What is Connection pooling? Which API do you use for connection pooling in your application and How?

What is difference between @ Bean and @Component and @Service?

What is Transaction management in JPA or Spring?

Different annotations in Spring and its explanation?

What is the way to call default and static method call in Java 8?

What is difference between @ Component and @ Service?

What is difference between Synchronized Hashmap and ConcurrentHashMap?

Which protocol do you use in your project?

New features of Java 8?

What is lambda expression?

What is default and static implementation in interface?

Why we have this static interface ? it could be also possible in any class?

If we are implementing default method in our class. Which method will be called our implemented method or default method implementation in Interface?

Interface A{

Default test(){}

} Class B implents A{

Test(){

}

}

Now Q.1 -> Can we implement test method?

Q.2 -> If we call test() method which method will be called?

Interface A {

Default test(){ }

}

Interface B{

Default test(){ }

}

Class C implements A, B{

We are calling c.test(); /// What it will give CTE/Exception?

}

What are different ways of locking in Java when we use synchronization?

Q-> There is a table called Employee. I have employee object I want to ensure that at any given point of time at a time one request for one employee should be updated. So suppose for the same employee I have three threads. I want to ensure for single request should be updated on sequential manner and for multiple request employee should be updated on parallel manner. How you will design the system?

To update a single employee object it should work sequential and for multiple employee that should go parallel. what level locking should we used? and how you will manage?

What is wait and notify method?

How long this thread will wait on calling wait () method? Is there any time limit?

What is difference between notify and notifyAll?

Which thread will get chance to continue its execution on after calling t1.notify() if t2, t3, and t4 are in waiting state?

Can we predict that which thread will get access next?

What is volatile keyword in java?

What are different type of Collection in Java?

How you decide which collection is used?

LinkedList vs ArrayList?

What is the complexity of each collection?

When we use Vector?

How will you implement your own List?

What is Map?

What is difference between HashMap and HashTable?

If I have multithreaded system which map should we use for taking care of Performance and Security both?

How Hashmap internally works?

If I want to use my custom class object as key ? what are the guidelines we have to follow?

How the collections are able to sort these element?

I have a collection of employee/We want to sort it. How it will be sorted. If I don’t define anything basis will it give nay error?

I have an array of size 10. I have two stack object s1 and s2. I am performing pop, push operation on both object but the element should be stored in one array only?

I want to build an approval management system. Where Request comes to Manager if manager rejects it. it revert back to requester. If Manager approves request goes to Senior Manger. If senior manager approves then it goes to higher level. I Want to design a system. How will you design it.ie what will be the classes and how the all will be related?

In the above case if you are creating the same methods in all level of class. Then why are you creating these methods at different level.

What are types of Joining.

What is subquery?

For extracting data which one is better approch .using subquery or using Joining?

What are aggregate function ?

Write a query to fetch sum of salary department wise?

If we want to fetch the second highest paid department.

How we create Immutable?

In two phase locking we use synchronized at class level locking or object level locking?

How we make the class so that no one can clone my singleton object?

Which class will have the clone method?

Clone method is in object class or not?

If we don’t override equals method what will be the issue?

Have you used Criteria query?

If we have any AND OR condition .how we use Criteria?

What is Restrictions in Hibernate?

In which case we don’t use Hibernate?

Types of Autowiring?

Beanfactory vs ApplicationContext?

Union vs UnionAll?

Query 1 🡪 1 2 3

Query 2 -> 1 2 3 4

On using Union and UnionAll what will be the output?

Triggers?

Difference between @Mock @InjectMock?

Anything you have implemented in your project recently? // Hystrix or Swagger

What is synchronization?

What is deadlock? How we prevent the lock in this case?

What are new features with Java 7?

Executor ,Callable, Future?

What is the use of yield and join method?

What is monitor in thread?

Can you brief ur current project you are working for and the challenges you have?

How you get the request from downstream application?

How your downstream application sends the request picked by you? Like SELA picks that request one by one or any multithreading involved to process?

How you connect your upstream application either its sequential or parallel processing?

Is result coming from one upstream application dependent on others?

What happens if you are sending the request to upstream or sending the response to down stream and it is taking a longer time .then how do you handle it? Do you just truncate or is there any timeout? Or you wait to return response from all the services, means how do you handle it?

If you a controller ABC and you are calling different services s1, s2, s3, s4.This service call is any API driven or Rest call? --- its a rest API call

If 4 different request has been sent to s1, s2 s3 and s4 so s1, s2 s3 has returned the response and suppose s4 service is slow or hang so your request is stuck. i.e no any negative response nor any exception? What is your approach to handle this? // **Hystrix**

If any exception occurs then Hystrix but if it is taking too long time due to stickness of any reason ?

then what is your approach?

**What are the different ways to iterate a HashMap?**

If you're only interested in the keys, you can iterate through the [keySet()](http://java.sun.com/j2se/1.5.0/docs/api/java/util/Map.html#keySet%28%29) of the map:

Map<String, Object> map = ...;

for (String key : map.keySet()) {

// ...

}

If you only need the values, use [values()](http://java.sun.com/j2se/1.5.0/docs/api/java/util/Map.html#values%28%29):

for (Object value : map.values()) {

// ...

}

Finally, if you want both the key and value, use [entrySet()](http://java.sun.com/j2se/1.5.0/docs/api/java/util/Map.html#entrySet%28%29):

for (Map.Entry<String, Object> entry : map.entrySet()) {

String key = entry.getKey();

Object value = entry.getValue();

// ...

}

What Is Entry Interface implementation?

Problem statement :

Questions:

1. You have a Java web server in production that got stuck: stopped writing logs and doesn’t answer to http requests. How would you investigate?
2. Logs say OutOfMemoryError - how would you investigate?
3. Customer clicks on a button and gets NullPointerException, but same case works correctly in development environment. There are no log statements in code that help analyse the problem. How would you investigate and what step would you take to ease debugging such problems in future?
4. You are investigating an issue and written a JUnit test - it fails as expected. So you start debugger and step over several times, but the issue is not reproduced. Why can this happen?
5. Your production Java web server typically handles http request in <50ms, but now you see times raised to about 1 second. How would you investigate?
6. You have multithreaded app which looks properly synchronized, but sometimes it seems that when particular field is read it returns old value. You suspect some synchronization issue, how would you investigate?