**Core Java**

1. What is advantage of using Java language?
2. What is Abstraction? (Tell 2 sentence definition )
3. What is abstract class? What are the access modifiers we can use for abstract class methods and class level variables? Can a abstract class have constructor?
4. What is interface? What are the access modifiers we can use for interface methods and class level variables?
5. What is polymorphism? What is overloading ? what is overriding?
6. Class a has a method add (int a, int b) . Class b is created by extending class a and overriding add method with 3 int parameters instead of 2. Will class compile? What happens in this case?
7. Why is string immutable?
8. What happens when we give  string x =”aa”;  x=”bb; x=”cc”; x=”aa”;  - How many strings are created in string pool.
9. What is serialization?
10. How to serialize object’s state ?
11. Can we call a static method from non-static method? Can we call a non-static method from static method?
12. Can I override a static method?

If a subclass defines a static method with the same signature as a static method in the superclass, then the method in the subclass *hides* the one in the superclass.

1. Can we overload a static method? Yes
2. What is synchronization?
3. Explain wait , notify, notify all , join methods.
4. What are the different collection interfaces?
5. Array vs ArrayList vs Generic List Vs LinkedList
6. How is data stored in Hashmap – explain the hashing logic.
7. To store a object in hashmap what methods we have to override in a class? (gethashcode, equals)
8. What is marker interface? Give an example of marker interface.
9. What is use of transient keyword? What is use of volatile keyword?
10. Diff between hashmap vs hashtable vs treemap.
11. Diff between treeset vs sortedset vs hashset.
12. Checked vs unchecked exceptions
13. Does Java support multiple inheritance and why
14. What is abstraction?

process of **abstraction in Java** is used to hide certain details and only show the essential features of the object. In other words, it deals with the outside view of an object (interface).

1. How to achieve abstraction?
2. **To create immutable class in java, you have to do following steps.**

* Declare the class as final so it can't be extended.
* Make all fields private so that direct access is not allowed.
* Don't provide setter methods for variables.
* Make all mutable fields final so that it's value can be assigned only once.

1. What is polymorphism?

Polymorphism is the ability of an object to take on many forms. The most common use of polymorphism in OOP occurs when a parent class reference is used to refer to a child class object.

1. How is polymorphism implemented in java? Through overloading and overriding (also called as compile time and runtime polymorphism)
2. **To Howmap works in java**
3. What methods are required to be overridden by an object it needs to be added in hashmap? Is it mandatory to override?
4. Difference between comparator and Comparable?
5. How to create a thread in java? Which method of creating thread is better?
6. Which version of Java have you used? What are the new features in the version which you worked on? (java 1.7
7. What is synchronization ?
8. Can we synchronize run method?

If you synchronize the run method, there are two outcomes:  
1) Synchronize does not help, if the threads are created using different runnable instances.  
2) There is no multi-threading, if all threads are created using the same runnable. Its as good as directly calling the run method from the same thread of execution

1. Synchronized block vs synchronized method.

One significant difference between synchronized method and block is that, Synchronized block generally **reduce scope of lock**. As scope of lock is inversely proportional to performance, its always better to lock only critical section of code. One of the best example of using synchronized block is [double checked locking in Singleton pattern](http://javarevisited.blogspot.com/2012/07/why-enum-singleton-are-better-in-java.html) where instead of locking whole getInstance() method we only lock critical section of code which is used to create Singleton instance. This improves performance drastically because locking is only required one or two times.

Synchronized block provide **granular control over lock**, as you can use arbitrary any lock to provide mutual exclusion to critical section code. On the other hand synchronized method always lock either on current object represented by [this keyword](http://javarevisited.blogspot.com/2012/01/this-keyword-java-example-tutorial.html)  or class level lock, if its static synchronized method.

Synchronized block can throw throw [java.lang.NullPointerException](http://java67.blogspot.sg/2012/09/what-is-nullpointerexception-in-java.html) if expression provided to block as parameter evaluates to null, which is not the case with synchronized methods.

In case of synchronized method, lock is acquired by thread when it enter method and released when it leaves method, either normally or by throwing Exception. On the other hand in case of synchronized block, thread acquires lock when they enter synchronized block and release when they leave synchronized block

1. What is class level lock vs object level lock?  (synchronized on static method will apply class level lock, synchronized on non-static method will apply object level lock)
2. What is abstraction?
3. How is abstraction supported in java?
4. Can I abstract class have a constructor?
5. How to call a constructor defined in abstract class?
6. What is constructor chaining?
7. What all access modifiers are allowed for a method in abstract class?
8. What all access modifiers are allowed for a class level variable in abstract class?
9. Can a method in abstract class be both abstract and static? Can it be both abstract and final? Can it be both static and final?
10. What access modifiers are allowed in interface for method and class level variables ?
11. What is difference between abstract class vs interface?
12. Which is better between abstract class and interface and why?
13. What is polymorphism?
14. How is polymorphism supported in java?
15. What is difference between overloading and overriding? (or) difference between compile time and runtime polymorphism?
16. Can we change the access modifier of a overridden method? If so what access modifier values it can take?
17. Can we overload a static method?
18. Can we override a static method?
19. Can the overridden method throw a different exception than the base class method?
20. Can we override a main method in class – public static void main (String args[] )? If we can override, will it get called?
21. What is serialization?
22. Can we control what data is serialized in a class?
23. Is serializable inherited?
24. What happens if a member class is present within main class and main class is implementing serializable.
25. What is difference between deep and shallow copy? How to create deep and shallow copy?
26. Why we have clonable interface when we already have a clone method in object class?
27. How to create a thread? Which method is better to create a thread?
28. What is difference between wait/Notify, join, yield, sleep methods.
29. What is synchronization?
30. Method vs block synchronization- which is better and why
31. What is difference between race condition and deadlock.
32. Can we make a run method itself synchronized? What happens.
33. What is difference between calling threadclass.start() vs Threadclass.run() method?
34. What are different types of memory used by java programs?
35. Where is local variables stored? Heap or stack memory?
36. Where are class level variables stored? Heap or stack?
37. What are functions calls stored? Heap or stack?
38. Where are thread variables stored? Heap or stack?
39. How to make a class /object immutable?
40. What is the logic for storing and retrieving objects from Hashmap? Explain the hashing logic used. What happens if 2 objects stored in hashmap has same key.
41. Is it mandatory to override gethashcode and equals method of a class to store that in hashmap? Why is it required?
42. What is difference between generic vs non-generic collections.
43. What collection types you have used – set, map , list, queue, stack, deque
44. Purpose of each collection type – set, list, map, queue, stack, deque.
45. What is difference between arraylist vs array, arraylist vs linkedlist, arraylist vs copyonwritearraylist vs linkedlist, vector vs arraylist
46. What is difference between hashmap vs hashtable, hashmap vs sortedhashmap vs identiyhashmap vs weakhashmap vs synchronizedhashmap vs Treemap
47. What is difference between priority queue vs concurrent queue implementations? What concurrent queue implementations are available?
48. What is difference between sortedset vs hashset vs treeset
49. What is failfast iterator and what is failsafe iterator?
50. What is difference between iterator vs listiterator vs enumaration?
51. What is difference between string, stringbuilder, stringbuffer? Which is better to use.
52. Explain the concept of string pool and how string instances are managed in memory.
53. What is difference between comparator vs comparable interface.
54. Observer and observable interfaces purpose and how to use.
55. Difference between runnable and callable (if you have worked with executor framework in Java 1.7)?
56. What is executor framework (java 1.7)
57. What is new in java 1.5, 1.6, 1.7 and 1.8 based on whichever versions you have worked on recently.
58. What is generics?
59. What is annotation?
60. What is static import?
61. Methods in object class

Queues :

1. 1.       What is a priority queue
2. 2.       What is priority in a priority queue
3. 3.       How is it implemented/ How does it work under the hood ?
4. 4.       Acknowledgment of messages

Threading :

1. 1.       What is the difference between sleep and wait
2. 2.       What is notify/notifyall
3. 3.       What is the issue with this code – issue with deadlock
4. 4.       When is lock released ?
5. 5.       What is Deadlock ?

Collection

1. 1.       What is ConcurrentHashMap
2. 2.       What is CopyOnArraylist ?
3. 3.       How do you convert Array into Arraylist ?
4. 4.       What is the difference between SingleLinked List and Doubly Linked List ?
5. 5.       What is the difference between ArrayList and LinkedList
6. 6.       How are they internally implemented
7. 7.       Which DS is better to use for insertion/deletion – ArrayList or LinkedList
8. 8.       Which DS is better to use for searching -- ArrayList or LinkedList
9. 9.       If I want to and and write lot of data which is better – Linked List or Array List ?
10. .       Can class file be decompiled ? What happens under the hood ?
11. 5.       Is Java Intepretable or Compilation language ? What is ByteCode ?
12. 6.       What is Concurrent Modification exception? When it happens ? He may give code snippet and ask you which exception will be thrown
13. 7.       What kind of reference we have in Java ? What is Weak reference and Phantom reference ?

Restful web Services

1.       What are the various kinds of methods used in restful web services – GET, PUT, POST etc

2.       When is each one of them used

3.       If I need to insert a record into db which restful ws method will I use

4.       Please explain the URL to be used to hit a restful ws

5.       Please explain the input / output format ?

**Struts 2.x**

1. What is Value Stack?
2. How to access session, application, request objects in action class? Or Explain all aware interfaces?
3. What is an interceptor? How to create a custom interceptor?
4. What is interceptor stack? How to create/configure an interceptor stack? Does struts provide a default interceptor stack?
5. How to configure default interceptor and default interceptor stack for all action classes?
6. How to handle validations in action class? What interceptors are needed for validation to work and purpose /role played by each of those interceptors to make validation work.
7. How to read resource files in action class and view layer to display messages or labels?
8. What is OGNL? How is it useful?
9. How is resource file searched in struts 2 framework? Explain the order in which it will look for a resource file .

**Struts 1.x**

1. What is dispatch action ? what is look up dispatch action? What is forward action? Difference between all 3.
2. What is dynaction form?
3. How to copy values from data transfer object to a form bean ? Do we have a utility?
4. How to handle validations in form bean and action class? What interface to implement in form bean for validation.
5. How to ensure validator framework as well as custom validation in business layer are both executed in action class?
6. How to handle duplicate form submission?
7. How to set the locale for a struts application if we are providing a option to select language during login and where we have to write this code.
8. How to add errors in action class and diplay error messages in UI?
9. How is resource file searched by struts framework?
10. If we have multiple resource files, how to specify which resource file to use in UI for displaying label or error message?

Spring MVC

1. Spring bean life cycle

<http://javabeginnerstutorial.com/spring-framework-tutorial/java-spring-bean-lifecycle/>

1. Bean factory vs applicationcontext

1) BeanFactory doesn't provide support for internationalization i.e. i18n but ApplicationContext provides support for it.

2) Another difference between BeanFactory vs ApplicationContext is ability to publish event to beans that are registered as listener.

3) One of the popular implementation of BeanFactory interface is XMLBeanFactory while one of the popular implementation of ApplicationContext interface is ClassPathXmlApplicationContext. On [Java web application](http://javarevisited.blogspot.sg/2012/08/what-is-jsessionid-in-j2ee-web.html) we use WebApplicationContext  which extends ApplicationContext interface and adds getServletContext method.

4) If you are using auto wiring and using BeanFactory than you need to register AutoWiredBeanPostProcessor using API which you can configure in XML if you are using  ApplicationContext. In summary BeanFactory is OK for testing and non [production](http://javarevisited.blogspot.in/2011/09/how-to-write-production-quality-code.html) use but ApplicationContext is more feature rich container implementation and should be favored over BeanFactory

1. Types of applicationContext and when to use which implementation
2. BeanPostProcessor
3. Interceptor HandlerInterceptorAdapter – How to create custom interceptors and methods in this interface.
4. Dependency Injection - Inversion of Control
5. How to set dependencies in spring.
6. Inherirtance in spring.
7. What is prototype? What is singleton?
8. Authorization and authentication in spring MVC.
9. Handling of stale data loading in application.
10. Validation in spring? (Validator and validatorUtils class usage has to be explained. )
11. How to handle duplicate form submissions in spring MVC?

Spring mvc provides flash attributes in order to  resolve issue of duplicate form submission. you can put attributes into flashmap that you want to get into success view page & get that values into success page. (See: <http://viralpatel.net/blogs/spring-mvc-flash-attribute-example/>)

1. What is an interceptor? How to implement?
2. How to handle exceptions in spring?
3. How to get servletcontext and servletconfig object in spring?
4. Spring request flow
5. What is aspect? What is advice? What is pointcut? What is JoinPoint
6. What are the different advice types in spring?
7. Spring AOP vs Aspect AOP
8. What are the different application context implementation available?
9. Different scopes of bean
10. Spring bean life cycle
11. What is ContextLoaderListener?
12. Handler mapping types?
    1. BeanNameUrlHandlerMapping
    2. SimpleUrlHandlerMapping
    3. ControllerClassNameHandlerMapping
    4. CommonsPathMapHandlerMapping
    5. DefaultAnnotationHandlerMapping
    6. RequestMappingHandlerMapping
13. What is view resolver? What are the view resolver types available?
    1. UrlBasedViewResolver
    2. InternalResourceViewResolver
    3. ResourceBundleViewResolver
    4. BeanNameViewResolver
    5. XmlViewResolver
14. What are the different controllers available in spring and purpose of each?
    1. Controller
    2. AbstractCommandController (methods in this controller)
    3. SimpleFormController (methods in this controller)
    4. WizardFormController (methods in this controller)
    5. MultiActionController (methods in this controller)
15. What is Look up method injection and method replacement in Spring

**Servlets**

1. Explain servlet life cycle method.
2. What is difference between generic and http servlet? Tell the class/interface they implement/extend.
3. What is difference between servlet config and servlet context.
4. How to forward request between a servlet and another other resource?
5. What is difference between forward and include in request dispatcher?
6. What are the difference servlet listener classes? Explain example scenarios when we can use each of these servlet listners.

**JSP**

1. What are the implicit objects in JSP (Application, session etc)
2. What are the different page attributes? Explain their use.
3. What are the JSP elements ? ( directive, scriptlets, actions)
4. What are the different JSP directives?
5. Include action vs include directive.
6. How to create an exception page and print exception details?
7. How to create a tag library? What classes to extend for different types of tags?

**Design Patterns**

1. What is singleton design pattern?
2. What is difference between static and Singleton?
3. What design pattern is implemented for pagination?
4. What is abstract factory pattern and factory pattern? Differences.
5. Give examples of 3 design patterns in creational and problem that it solves?
6. Give examples of 3 design patterns in structural and problem that it solves?
7. Give examples of 3 design patterns in behavioural and problem that it solves?
8. Give examples of 3 core j2ee design patterns?

**General**

1. What best practices have you implemented in project?
2. How do you analyze performance issues?
3. How do you handle code review? What all do you check?
4. What is SOLID principles?
5. Do you understand OO metrics?
   1. Depth of inheritance
   2. Cyclometric complexity
   3. Number of operations per class
   4. Number of parameters per method.
   5. Coupling between objects.
   6. Weighted methods per class.
   7. Comment ratio
6. What are the technologies and Architecture used in last project?

Ans : Spring,Hibernate,Java

1. What are the UI framework used in your project?

Ans : Jsp,HTML

1. Queries idea?

Ans: No

1. Javascript idea?

Ans : Basic

1. Having idea about Ajax Framework ?

Ans : Basic

1. What are the DB servers used?

Ans : Only SQL server

1. What about EJB and types?

Ans : Yes used, pojo class is used and hibernate for persistent.

**Basic Hibernate Questions:**

1. Load and get diff?

Ans : No Idea

1. How will you save the data into the DB ?

Ans : Save or Update method used

1. Any Session proxy used?

Ans : No

1. What are the core interface available in hibernate?

Ans : Not remember

1. Do you have idea about HQL?

Ans : Have some idea

1. What are the things we could configure in Hibernate.conf.xml

Ans : Db Connection,Cache

1. What cache used in your project ?

Ans : Secondary

1. How we can reattach the detached object?

Ans : No Idea

**Spring Basic Questions**

1. Which version of Spring used?

Ans : Spring 3

1. What kind of component used in your project?

Ans : MVC only

1. Any idea about front controller?

Ans : No

1. How we can transfer data to DB?

Ans : BO file(View-Controller-Business logic-Model)

1. Which method is used call controller?

Ans : Get method

1. Which methodology or service used in your project for object distribution?

Ans: Singleton

1. What kind of Injection used?

Ans : Dependency Injection used in my project and other types are Constructor and Interface.

1. Have you used Bean wiring in your project? Any idea?

Ans: No idea

1. Bean Scope Types?

Ans: Singleton in our project and other type is Prototype.

1. What is the difference between Bean factory and Application Context?

Ans: Bean factory will have set of Bean pool

1. How we can resolve view in spring?

Ans :Based on Bean ID

1. What parameter will be user for view resolver?

Ans : Not remember.

**Data Base Queries**

1. What kind of queries used in your project and what you know?

Ans : I know simple query and Cursor, functions but used only simple queries in my project.

1. Are you used index?

Ans : No

**Java Basic Questions**

1. When will you use hashcode and how?

* No idea

1. Do you have idea about out of memory error?

Ans : No

1. What is immutable in java?

Ans : Strings are basically immutable in java

1. What is the diff between New String and String literal? Where it will be stored?

Ans: Literals stored in heap and New operator will store in Stack.

1. What is the diff b/w Checked and unchecked exception?

Ans : Checked exceptions are identified compile time itself ex : Symbol Cannot be resolved.

Unchecked exceptions will be identified at runtime Ex : ArrayOutOFBoundsException, FileNotFound

1. Throw and Throws difference?

Ans :Throw will lose the control but Throws will have the control

1. If a class doesn’t inherit any class then super key word will point to?

Ans : No idea

1. Diff b/w Sql.date object and Java.util.Date?

Ans :No idea

1. Diff b/w iterator and list iterator?

Ans : Iterator only forward direction but list can be both (Forward,Revers)

1. Diff b/w Linked hashmap and Hashmap?

Ans : Linked HM will be stored in insertion order but HashMap stored as random

1. Diff b/w string builder and Buffer?

Ans : Buffer is synchronized but builder not.

1. What kind of build used in your project?

Ans : Ant used in our project, Maven build no idea

1. What are the technologies and Architecture used in last project?

Ans : Spring,Hibernate,Java

1. Have you worked on any oracle project ?

Ans : No

1. What are the UI framework used in your project?

Ans : Jsp,HTML

1. JQueries idea?

Ans: No

1. Having idea about Ajax Framework ?

Ans : Basic

**Basic Hibernate Questions:**

1. Load and get diff?

Ans : Load wont throw null pointer exception when object not fount but get will throw exception.

If we sure that the object available then we can go with get.

1. What are the core interface available in hibernate?

Ans : Session,SessionFactory

1. Do you have idea about HQL?

Ans : Have some idea

1. What are the things we could configure in Hibernate.conf.xml

Ans : Db Connection,Cache

1. What cache used in your project ?

Ans : Secondary

1. How cache is working in hibernate?

Ans : No Idea

**Spring Basic Questions**

1. How hibernate template will be called?

Ans : Controller,Businesslogic,DAO

1. What kind of component used in your project?

Ans : MVC only

1. How business class is configured in Spring

Ans : Application Context by Bean Id, Class tag

1. Any idea about front controller?

Ans : Dispatch Servlet used as front controller.

1. How we can transfer data to DB?

Ans : BO file(View-Controller-Business logic-Model)

1. Which method is used call controller?

Ans : Get method

1. What methodology used for object configuration?

Ans : Dependency injection

1. Which component transfer data to Db ?

Ans : DAO

1. Bean Wiring any idea?

Ans : No

1. What kind of tags are used in jsp?

Ans : C tag

What is the use of this tag?

Ans : No idea.

**Data Base Queries**

1. What kind of queries used in your project and what you know?

Ans : I know simple query and Cursor, functions but used only simple queries in my project.

1. Are you used index?

Ans : No

1. Do u have idea about sub queries ?

Ans : Yes, Query within a query.

**Java Basic Questions**

1. When will you use hashcode and how?

* No idea

1. Do you have idea about out of memory error?

Ans : When more object are created then this error will come, but no idea how we can resolve it.

1. What is immutable in java?

Ans : Strings are basically immutable in java

1. What is the diff b/w Checked and unchecked exception?

Ans : Checked exceptions are identified compile time itself ex : Symbol Cannot be resolved.

Unchecked exceptions will be identified at runtime Ex : ArrayOutOFBoundsException, FileNotFound

1. Throw and Throws difference?

Ans :Throw one level but Throws can be multiple level.

1. Diff b/w Sql.date object and Java.util.Date?

Ans :System date format, DB date format

1. Diff b/w iterator and list iterator?

Ans : Iterator only forward direction but list can be both (Forward,Revers)

1. Diff b/w Linked list and ArrayList?

Ans : Linked list will be stored in insertion order but Arraylist stored as random

1. Diff b/w string builder and Buffer?

Ans : Buffer is synchronized but builder not.

1. What kind of build used in your project?

Ans : Ant used in our project, Maven build no idea

1. Can be abstract and final keyword at a time?

Ans : No