## CoreJava Interview Questions

1. What are the usage of Java packages ?  
Ans - It helps in following ,

* It helps in program organization, i.e keeping related files together
* It helps in resolving name conflicts.

Q. What is difference between NoClassDefFoundException and ClassNotFoundException ?  
Ans - NoClassDefFoundException is raised when a class is referenced by "new" operator. ClassNotFoundException is raised when class is referenced dynamically. One of the example of dynamic class loading is Class.forName(String className)

Q. One important tips - How does in eclipse you see a folder type icon "Referenced Libraries" ?  
Ans - When we add jars to the Java project created in the eclipse / sts then it starts appearing as "Referenced Library" .

Q. What is "Inversion of Control" ?   
Ans - Its basically approach of outsourcing of construction and management of objects.

One of best practices of software engineering is "Coding to interface".

Q. How do you visualize Spring container ?  
Ans - On the high level we can think of this like in below notes / tips ,

* You can think of that there is an application "MyApp" and and its demanding of objects, llike let say its saying that give me a "Coach" object. Its demanding this object from say Spring container or say "object factory". Factory / Spring container creates object with the help of configuration present in the spring configuration file.

Q. What is primary function of Spring Container ?  
Ans - Below are the primary functions of Spring container ,

* Create and manage objects (Inversion of control)
* Inject object's dependencies(Dependency Injection)

Q. What are different ways of configuring Spring container ?  
Ans - Below are the different ways to configure Spring container,

* XML Configuration file
* Java annotations
* Java source code

Q. How will you summarise "Spring Development Process" ?  
Ans - Below are the high level steps or say to-do list ,

* Configure you Spring beans
* Create a Spring container
* Retrieve beans from Spring Container

Q. By what name Spring container is known generally ?  
Ans - Spring container is generally known as ApplicationContext. Below is a sample code,  
 ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");

Q. Once spring container is built, how do you get bean from a Spring container ?  
Ans - Below is the example of getting bean from Spring container,  
 Coach myCoach = context.getBean("myCoach", Coach.class);  
Q. What is a Spring bean ?  
Ans - A Spring bean is simply a Java object. When java objects is created by Spring container, then Spring refers to them as Spring beans. Spring beans are created from normal Java classes, just like Java objects.

Q. Why do we specify the Coach interface in the getBean() method ?  
Ans - Below is the code snippet of above problem statement, Below is the sample Java code,  
Coach theCoach = context.getBean("myCoach", Coach.class);

When we pass the interface to the method, behind the scenes Spring will cast the object for you.

Behaves the same as getBean(String), but provides a measure of type safety by throwing a BeanNotOfRequiredTypeException if the bean is not of the required type. This means that ClassCastException can't be thrown on casting the result correctly, as can happen with getBean(String).

Q. What are different type of injections in the Spring ?  
Ans - There are 2 types of most common dependency injection. They are,

* Constructor Injection
* Setter Injection

Note : - We just completed some very basic sample Spring code using "Dependency Injection". We used "Constructor Injection" technique for "Dependency Injection".