### SPRING

* Explain Spring:
  1. Inversion of Control: Implementation of technique to facilitate loose coupling, where objects are given their dependencies rather than explicitly creating them.
  2. Container: Manages life-cycle and configuration of application objects.
  3. Framework: Provides horizontal services, leaving domain coding to developer.
* Aspect Orientated: Facilitates abstraction of system services from application business logic.
* Prototype vs singleton
* depends-on (how would you enforce an order of initialization
* How does Spring differ from Struts (Asked to people having an exp in struts). How dependency injection works in Struts?
* One singleton bean uses another prototype bean and if singleton bean is requested again and again, then how many objects of both will be created.
* ApplicationContextAware
* Annotations - @Autowired, @RequestMapping, @Controller
* Good Understanding of Core, MVC , Batch, Security, Rest, SOAP Other Spring Technologies
* Annotation configuration in spring
* Configuration for spring MVC
* Singleton pattern vs. Spring singleton type
* Filters in MVC spring application
* AOP internal implementation
* MVC application structure
* In Spring Data how can you design class to read the data from file system rather than reading from any database?
* What are the benefits of using spring boot
* How security is implemented in Spring (oauth2, SAML and custom token)
* At which later, transaction is managed ?
* How to handle the full transaction b/w two methods of DAO class if each method is having seperate transaction.

### Hibernate

* Difference between get and load
* What is SessionFactory and Session in Hibernate
* Hibernate second level cache
* How do you maintain caching in your project
* How to invalidate , refresh your cache and why you will refresh ?