1)Question

DQueue

Priority Queue

Tree Set

LinkedHashSet

Hash Set

Linked List

Array List

Queue

Sorted Set

Set

List

Iterable

Collection

2)Question

Implementation of Map Interface

NavigableMap

TreeMap

LinkedHashMap

SortedMap

HashMap

HashTable

Map

3) Hash Set

The Hash set is executed with the help of a HashTable.

It does not authorise a heterogeneous object.

It permits a null object.

To compare two objects, we use the equals method.

It does not support any order

Tree Ser

The tree set is executed with the help of a tree structure.

It authorises a heterogeneous object.

It does not permit the null object

To compare two objects, we use the compare method.

TreeSet supports an object in sorted order.

9) Core (spring-core) is the core of the framework that power features such as Inversion of Control and dependency injection. Beans (spring-beans) provides Beanfactory, which is a sophisticated implementation of the factory pattern

10) Dependency Injection is a fundamental aspect of the Spring framework, through which the Spring container “injects” objects into other objects or “dependencies”. Simply put, this allows for loose coupling of components and moves the responsibility of managing components onto the container

It is achieved by the Spring-Core module is responsible for injecting dependencies through either Constructor or Setter methods.

11)By using @Autowired and @ComponentScan annotation

12)Yes we can have more than one configuration file in one project By the annotation that for configuration file @Configuration for Component file @Component for Bean it is @Bean.

13) StringBuilder :-StringBuilder is non-synchronized i.e. not thread safe. It means two threads can call the methods of StringBuilder simultaneously.

StringBuilder is more efficient than StringBuffer.

StringBuilder was introduced in Java 1.5

StringBuffer:- StringBuffer is synchronized i.e. thread safe. It means two threads can't call the methods of StringBuffer simultaneously.

StringBuffer is less efficient than StringBuilder.

StringBuffer was introduced in Java 1.0

14)BeanFactory:- This is the root interface for accessing a Spring bean container. It is the actual container that instantiates, configures, and manages a number of beans. These beans collaborate with one another and thus have dependencies between themselves. These dependencies are reflected in the configuration data used by the BeanFactory. This interface is implemented by the objects that hold a number of bean definitions, each uniquely identified by a String name. The most common implementation class used for this BeanFactory is **XmlBeanFactory** available in **org.springframework.beans.factory.xml package.**

**ApplicationContext:-** This interface is designed on top of the BeanFactory interface. The ApplicationContext interface is the advanced container that enhances BeanFactory functionality in a more framework-oriented style. While the BeanFactory provides basic functionality for managing and manipulating beans, often in a programmatic way, the ApplicationContext provides extra functionality like MessageSource, Access to resources, Event propagation to beans, Loading of multiple (hierarchical) contexts etc. There are so many implementation classes that can be used such as **ClassPathXmlApplicationContext**, **FileSystemXmlApplicationContext**, **AnnotationConfigWebApplicationContext** etc.

15)Spring AOP:- Aspect-Oriented Programming (AOP) is one of the key elements of the Spring Framework. AOP praises Object-Oriented Programming in such a way that it also provides modularity. But the key point of modularity is the aspect than the class. AOP breaks the program logic into separate parts called concerns.

IOC:- Spring IoC is the mechanism to achieve loose-coupling between Objects dependencies. To achieve loose coupling and dynamic binding of the objects at runtime, objects dependencies are injected by other assembler objects.

@Component:- @Component is a class-level annotation. It is used to denote a class as a Component. We can use @Component across the application to mark the beans as Spring's managed components. A component is responsible for some operations.

@Bean:- Spring @Bean Annotation is applied on a method to specify that it returns a bean to be managed by Spring context. Spring Bean annotation is usually declared in Configuration classes methods. In this case, bean methods may reference other @Bean methods in the same class by calling them directly.