**Q1. what is the @component and @Controller.**

**ANS-** -The @**Component** is **used** as a meta-annotation here so that it **can** be picked-up using **component**-scanning. **You can** also create your own @**Component** based annotations by simply creating your own annotation and putting @**Component** on it

-@**Controller** is used in Spring MVC to define **controller**, which are first Spring bean and then **controller**.

**Q2. @RequestMapping,@Required,@Qualifier,@Autowired@Temporal,@Entity @RequestBody @RestController @Query @PathVariable Annotation, explain?**

**ANS-** The **@RequestMapping** annotation can be applied to class-level and/or method-level in a controller.

The class-level annotation maps a specific request path or pattern onto a controller. You can then apply additional method-level annotations to make mappings more specific to handler methods.

The **@Required** annotation applies to bean property setter methods and it indicates that the affected bean property must be populated in XML configuration file at configuration time.

the **@Qualifier** annotation along with **@Autowired** to remove the confusion by specifying which exact bean will be wired.

The **@Autowired** annotation provides more fine-grained control over where and how autowiring should be accomplished. The @Autowired annotation can be used to autowire bean on the setter method just like @Required annotation, constructor, a property or methods with arbitrary names and/or multiple arguments.

**@Temporal annotation** in JPA implementation can only be used with the fields and property get methods.

**@Entity -**Entities in JPA are nothing but POJOs representing data that can be persisted to the database. An entity represents a table stored in a database. Every instance of an entity represents a row in the table.

The **@RequestBody** annotation allows us to retrieve the request's body. We can then return it as a String or deserialize it into a Plain Old Java Object (POJO).

**@RestController** is the combination of @Controller and @ResponseBody. Flow of request in a @Controller class without using a @ResponseBody annotation: @RestController returns an object as response instead of view. Show activity on this post.

@**PathVariable** has only one element 'value' that is used to define URI template variable name. @PathVariable 'value' element is optional.

**Q3. What are the Stereotype annotations and define MVC flow.**

**ANS-** The stereotype annotations in spring are @Component, @Service, @Repository and @Controller. This annotation is used in classes to indicate a Spring component. The @Component annotation marks the Java class as a bean or says component so that the component-scanning mechanism of Spring can add to the application context.

**Q4.- Diffrence between application.properties & YML fiile.**

**ANS**- 1.).properties file : It store data in sequential format..yml file : It store data in hierarchical format. 2.).properties file : It supports only key-value pair basically string values..yml file : It supports key-value pair as well as map, list & scalar type values.

**Q5. What are the scopes in Spring Framework?**

**ANS**- In this quick tutorial, we'll learn about the different types of bean scopes in the Spring framework. The scope of a bean defines the life cycle and visibility of that bean in the contexts we use it. The latest version of the Spring framework defines 6 types of scopes.

**Q6. What is cascading and what are different types of cascading?**

**ANS-**

Cascading Style Sheet(CSS) is used to set the style in web pages that contain HTML elements. It sets the background color, font-size, font-family, color. etc property of elements on a web page.   
There are three types of CSS which are given below: 

* Inline CSS
* Internal or Embedded CSS
* External CSS

**Q7.what is the ioc container and autowiring**

**ANS-**

The Spring container is at the core of the Spring Framework. The container will create the objects, wire them together, configure them, and manage their complete life cycle from creation till destruction. The Spring container uses DI to manage the components that make up an application.

**Q8.What is difference between hibernate and jpa? and What is Hibernate cache?.**

**ANS**-

A JPA (Java Persistence API) is a specification of Java which is used to access, manage, and persist data between Java object and relational database. It is considered as a standard approach for Object Relational Mapping.

A Hibernate is a Java framework which is used to store the Java objects in the relational database system. It is an open-source, lightweight, ORM (Object Relational Mapping) tool.

**Q10.what is difference between Named Queries and Criteria Queries?**

**ANS-**

**Named queries are more optimal** (they are parsed/prepared once). Criteria queries are dynamic, (they are not precompiled, although some JPA providers such as EclipseLink maintain a criteria prepare cache). I would use criteria only for dynamic queries.