Assignment Questions 10

Q1. What is the Spring MVC framework?

The Spring web MVC framework provides model-view-controller architecture and ready to use components that are used to develop flexible and loosely coupled web applications. The MVC pattern helps in separating the different aspects of the application like input logic, business logic and UI logic, while providing a loose coupling between all these elements. Get Started with Spring MVC

Q2. What are the benefits of Spring MVC framework over other MVC frameworks?

Advantages of Spring MVC Framework

- The container is used for the development and deployment of applications and uses a lightweight servlet.
- It enables rapid and parallel development.
- Development of the application becomes fast.
- Easy for multiple developers to work together.
- Easier to Update the application.
- It is Easier to Debug because we have multiple levels in the application.

Q3. What is DispatcherServlet in Spring MVC? In other words, can you explain the Spring MVC architecture?

The *DispatcherServlet* acts as the front controller for Spring-based web applications. It provides a mechanism for request processing where actual work is performed by configurable and delegated components (usually controllers). It is inherited from *HttpServlet* and is typically configured in the web.xml file.

Q4. What is a View Resolver pattern and explain its significance in Spring MVC?

View Resolver pattern is a J2EE pattern that enables a web application to dynamically select its view technology. For example, HTML, JSP, Tapestry, JSF, XSLT etc. In this pattern, View resolver holds mapping of different views, controller return name of the view, which is then delegated to the View Resolver for selecting an appropriate view. Spring MVC framework supplies inbuilt view resolver for selecting views.

Q5. What are the differences between @RequestParam and @PathVariable annotations?

@RequestParam and @PathVariable annotations are used for accessing the values from the request.

The primary difference between @RequestParam and @PathVariable is that @RequestParam used for accessing the values of the query parameters where as @PathVariable used for accessing the values from the URI template.

Q6. What is the Model in Spring MVC?

A model contains the data of the application. A data can be a single object or a collection of objects.

Q7. What is the role of @ModelAttribute annotation?

@ModelAttribute is an annotation that binds a method parameter or method return value to a named model attribute, and then exposes it to a web view.
In this tutorial, we'll demonstrate the usability and functionality of this annotation through a common concept, a form submitted from a company's employee.

Q8. What is the significance of @Repository annotation?

@Repository Annotation is a specialization of **@Component** annotation which is used to indicate that the class provides the mechanism for storage, retrieval, update, delete and search operation on objects.

Q9. What does REST stand for? and what is RESTful web services?

Representational State Transfer (REST) is a software architecture that imposes conditions on how an API should work. REST was initially created as a guideline to manage communication on a complex network like the internet.

RESTful web services are services that follow REST architecture. REST stands for Representational State Transfer and uses HTTP protocol (web protocol) for implementation. These services are lightweight, provide maintainability, scalability, support communication among multiple applications that are developed using different programming languages. They provide means of

accessing resources present at server required for the client via the web browser by means of request headers, request body, response body, status codes, etc

Q10. What is differences between RESTful web services and SOAP web services?

SOAP and REST are two internet data exchange mechanisms. For example, imagine that your internal accounts system shares data with your customer's accounting system to automate invoicing tasks. The two applications share data by using an API that defines communication rules. SOAP and REST are two different approaches to API design. The SOAP approach is highly structured and uses XML data format. REST is more flexible and allows applications to exchange data in multiple formats.