Assignment Questions 10

Q1.What is the Spring MVC framework?

Spring MVC is a Java framework used to build web applications. It follows the Model-View-Controller (MVC) design pattern. Spring MVC is request-driven and is designed around a central Servlet that dispatches requests to controllers and offers other functionality that facilitates the development of web applications. Spring MVC provides an elegant solution to use MVC in the Spring framework by the help of DispatcherServlet.

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

Spring MVC is a lightweight framework that is highly productive and can boost your development. It is fully secure, which is why most of the online banking web applications are developed using Spring MVC. Spring Security is a great API for enterprise-grade security implementation. Spring MVC is based on the Model-View-Controller (MVC) design pattern. It provides clear disseverment of roles such as controller, validator, command object, form object, model object, DispatcherServlet, handler mapping, view resolver, etc. Each role can be consummated by a specialized object.

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

Spring MVC is designed around the front controller pattern where a central Servlet, the DispatcherServlet, provides a shared algorithm for request processing. DispatcherServlet is a class that receives the incoming request and maps it to the right resource such as controllers, models, and views. It manages the entire request handling process. By default, DispatcherServlet will look for a name dispatcher-servlet.xml to load the Spring MVC configuration.

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

A View Resolver pattern is used to map view names to actual views. It is responsible for resolving views from the view name. The view resolver pattern is significant in Spring MVC because it allows you to use different types of views such as JSP, PDF, Excel, etc. without changing the controller code.

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

In Spring MVC, @RequestParam is used to bind a request parameter to a method parameter. It is typically used to extract query parameters or form data from an HTTP request . On the other hand, @PathVariable is used to bind a URI template variable to a method parameter. The key difference between @RequestParam and @PathVariable is that @RequestParam is used for accessing the values of the query parameters where as @PathVariable is used for accessing the values from the URI template.

Q6.What is the Model in Spring MVC?

In Spring MVC, the model works as a container that contains the data of the application. It can be in any form such as objects, strings, information from the database, etc. The Model interface is required to be placed in the controller part of the application. It defines a placeholder for model attributes and is primarily designed for adding attributes to the model. It is also used to transfer data between the view and controller of the Spring MVC application.

Q7.What is the role of @ModelAttribute annotation?

In Spring MVC, @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. It is used for preparing the model data and defining the command object that would be bound with the HTTP request data. The annotation works only if the class is a Controller class.

Q8.What is the significance of @Repository annotation?

In Spring, the @Repository annotation is used to indicate that the class provides the mechanism for storage, retrieval, search, update and delete operation on objects. It is a specialization of the @Component annotation and is used to encapsulate storage, retrieval, and search behavior which emulates a collection of objects. Spring Repository is very close to DAO pattern where DAO classes are responsible for providing CRUD operations on database tables.

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

REST stands for REpresentational State Transfer. It is an architectural style that provides standards between computer systems on a web service. RESTful web services are applications that follow the REST architecture. RESTful web services locate the resource by using the URL and perform actions based on the transport protocol.

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

The main differences between RESTful web services and SOAP web services are:

- SOAP is a protocol, while REST is an architectural approach.
- SOAP requires more bandwidth and resource than REST, which requires less bandwidth and resource.
- SOAP defines its own security, while REST inherits security measures from the underlying transport.
- SOAP permits XML data format only, while REST can use any data format.
- SOAP reads cannot be cached, while REST reads can be cached.