**Day 12 Revisit Spring Web (MVC) – Spring AOP**

MVC – Model View Controller Design Pattern

Model (Data Layer)

View (Presentation Layer)

Controller (Business Layer – Controls Entire Application)

WRT Data Flow 🡪 User to Application

Application to User

Spring Web (Spring MVC) is a module for designing web based applications.

@Component – Helps to indicate a class as a bean and framework manages it’s lifecycle

@Entity – JPA Annotation which represents a Bean class is a reference to the DB Table

@Table (To provide custom name to the DB table)

@Id – To specify the Primary Key of the DB table in our Java Code

@GeneratedValue – Helps to Auto Generate the numerical ID property in the DB table

JPA – Java Persistence API (It’s a Specification to simplify DB operations)

Popular JPA Implementations

1. Hibernate (ORM Framework – Object Relational Mapping)
2. iBatis/MyBatis
3. EclipseLink

Spring AOP (Aspect Oriented Programming) - AspectJ

JointPoints

PointCut

Advice

@Aspect

**Day 13 Agenda Spring Boot & CRUD with Mongo DB**

* SPRING BOOT INTRODUCTION
  + Introduction to Spring Boot
  + Value Proposition of Spring Boot
  + High-level Spring Boot features
  + Creating a simple Boot application using Spring

Initializr web-site

* SPRING BOOT DEPENDENCIES, AUTO-CONFIGURATION AND RUNTIME
  + Dependency management using Spring Boot starters
  + How auto-configuration works
  + Configuration properties
  + Using CommandLineRunner
  + Using In-memory Database (h2)
* Introduction to MongoReposiotory
  + Performing CRUD Operations with MongoDB
* Testing the Endpoints using Postman
* API Documentation using Swagger

Web Service – Services based on web (Internet)

* Machine to Machine Communication using Http (TCP/IP/Http)
* Invoking a method based on URI and getting responses

URI – URL

URI – Uniform Resource Identifier (Endpoint) – api/v1/trainings

URL – Uniform Resource Locator – http://www.google.com?searchq=dfngdlfng

Path – Absolute Path & Relational(Relative) Path

Types of WebService

* SOAP – Simple Object Access Protocol
* REST – Representation State Transfer (It re-uses HTTP protocol)

|  |  |  |  |
| --- | --- | --- | --- |
| Sl No | Http Method | DB Operation | Example URI |
| 1 | Get() | ReadAll | Api/v1/trainings |
| 2 | Get(int id) | ReadById | Api/v1/trainings/{id} |
| 3 | Post(Object obj) | Create (Insert) | Api/v1/trainings |
| 4 | Put(Int id, Object updates) | Update | Api/v1/training/{id} |
| 5 | Delete(int id) | DeleteById | Api/v1/trainings/{id} |

SpringBoot – It’s a simplified way of creating Spring based Enterprise Application

<https://spring.io> -

Creating a SpringBoot based Application

1. Using Spring Initializr (<https://start.spring.io>)
2. Using Spring recommended IDE (STS/VS Code/IntelliJ)

Spring Boot Auto Configuration

@SpringBootApplication – Annotation

SpringBoot is called as Opinionated Framework.

Configurations

1. Application.properties /Application.yml
2. Annotations @Configuration
3. Pom.xml (jars)

Important Annotations

1. @SpringBootApplication [ @ComponentScan, @EnableAutoConfiguration & @SpringBootConfiguration]
2. @Autowired
3. @RestController
4. @Service
5. @RequestBody
6. @PathVariable
7. @RequestMapping
8. @GetMapping
9. @PostMapping
10. @PutMapping
11. @DeleteMapping
12. @Id
13. @Entity
14. @GeneratedValue

Testing API endpoints

1. Using Postman (HttpClient)
2. Using Curl
3. Using Swagger (API Documentation & Testing tool)

Download Postman

<https://www.postman.com/downloads/>

<https://chrome.google.com/webstore/detail/postman/fhbjgbiflinjbdggehcddcbncdddomop>

<https://www.mongodb.com/developer/code-examples/java/rest-apis-java-spring-boot/>

<https://spring.io/guides/gs/accessing-data-mongodb/> -- Using Command Line Runner

Using REST Repositories