01 - Spring MVC Introduction

Spring MVC (Model-View-Controller) is a part of Spring Framework that helps in building web applications by separating concerns into three main components:

1. **Model:** Represents the application's data and business logic.
2. **View:** Responsible for displaying data to the user.
3. **Controller:** Handles the user requests, processes them (by interacting with the model), and returns the appropriate view.

- Spring MVC is built around **DispatcherServlet**, which acts as the central controller that dispatches requests to different components based on configuration.
- We need an IDE like Eclipse, IntelliJ IDEA, or Spring Tool Suite (STS) for developing Spring MVC applications.
- For deploying the web application, we can use an external Apache Tomcat server.

  Link for Apache TomcatServer: https://tomcat.apache.org/download-10.cgi

👉 Advantages of Spring MVC:

➢ **Separation of Concerns:** The MVC architecture separates the logic of the application, making it more maintainable and scalable.

➢ **Annotation-Driven Configuration:** Spring MVC supports easy configuration through annotations (@Controller, @RequestMapping, @ModelAttribute), reducing the complexity of XML configuration.

➢ **Robust Request Handling:** It provides powerful request handling, including support for multiple HTTP methods (GET, POST, etc.)

- **Integration with Other Spring Features:** It seamlessly integrates with other modules of the Spring Framework, such as Spring Security, Spring Data, and Spring Boot, making it easy to build fully-featured web applications.
- **Flexible View Resolution:** Spring MVC allows the use of different view technologies (JSP, Thymeleaf, FreeMarker).
- **Form Handling and Validation:** It provides easy form handling and data binding capabilities. It also integrates with JSR-303/JSR-380 for validating user inputs via annotations like @Valid.