**2. Spring MVC (Model-View-Controller) 1.** **Spring MVC Overview:**

• **Theory:**

1. **Introduction to the MVC design pattern and how it is implemented in Spring:-**

**MVC (Model-View-Controller)** is a design pattern that separates an application into three components:  
1️.Model (M) → Manages business logic & data (e.g., database operations).  
2️.View (V) → Handles UI presentation (e.g., JSP, Thymeleaf).  
3️.Controller (C) → Controls request processing & response handling.

**How is MVC Implemented in Spring?**

Spring MVC follows this architecture:

1️.**Client Request** → Sent to a **Spring Controller**.  
2️.**Controller** → Processes the request & interacts with the **Model** (Service/DAO).  
3️.**Model** → Fetches data & returns it to the Controller.  
4️.**Controller** → Passes data to the **View**.  
5️.**View (JSP/Thymeleaf)** → Displays the response to the user.

**2.Explanation of core components: Controller, Model, and View:-**

**1. Controller (C) → Handles Requests**

* Acts as an intermediary between the Model and View.
* Handles HTTP requests, processes data, and returns a response.
* Uses @Controller or @RestController annotation.

**2. Model (M) → Manages Data & Business Logic**

* Represents the **data & business logic** of the application.
* Uses **Service** and **Repository** layers for database operations.
* Model is used to pass data to the View.

**3. View (V) → Displays Data**

* The **UI layer** that presents data to users.
* Can be JSP, Thymeleaf, or other templates.