



1. MVC기본구조및패턴

1. JSP와 서블릿 개념 소개:

- JSP는 화면단(View)을 담당하는 기술로, HTML과 유사하지만 Java 언어를 함께 사용할 수 있다는 점이 특징입니다. 주로 웹 페이지의 동적 콘텐츠를 생성하는 데 사용됩니다.
- 서블릿은 컨트롤러(Controller) 역할을 하며, 데이터와 화면단을 연결하는 논리적 코드를 작성하는 영역입니다.

2. 웹 어플리케이션 서버(WAS):

- 웹 서버는 정적 데이터(HTML, CSS, JavaScript)만을 처리하는 반면, 웹 어플리케이션 서버(WAS)는 동적 데이터(Java와 같은 백엔드 언어)를 처리할 수 있습니다.
- WAS는 웹 어플리케이션을 실행시키고 서버 환경을 제공하는 소프트웨어 프레임워크입니다.

3. MVC 패턴:

- MVC(Model-View-Controller) 패턴은 웹 어플리케이션 서버에서 사용하는 백엔드 아키텍처 패턴입니다.
- **Model:** 데이터 저장소에서 데이터를 가져오고, 데이터를 캡슐화하여 자바 빈즈(Java Beans)로 관리하는 역할을 합니다.
- **View:** JSP와 같이 화면을 구성하고, 사용자가 볼 수 있는 데이터를 출력하는 역할을 담당합니다.
- **Controller:** 서블릿과 같이 모델과 뷰 사이에서 논리적으로 연결하는 역할을 하며, 데이터를 화면에 전달하거나 사용자의 입력을 처리합니다.

4. MVC 패턴의 장점:

- 유지보수성이 뛰어납니다. 역할이 분리되어 있어 코드 수정 시 어디에서 문제가 발생했는지 찾기 쉽고, 문제 해결이 간편합니다.
- 코드의 확장성이 높아져, 프로젝트가 복잡해져도 구조적으로 관리가 용이합니다.

5. JSP와 서블릿의 활용도:

- JSP와 서블릿은 과거에 많이 사용되었지만, 최근에는 스프링과 같은 프레임워크가 더 많이 사용되고 있으며, 정부 사업 등 특수한 경우를 제외하면 사용 빈도가 줄어드는 추세입니다.

.