



# JSP 프로그래밍

## 2차시

- Servlet 이란?
- JSP 이란?



## □ Servlet

- 자바를 사용하여 웹 페이지를 동적으로 생성하는 서버 측 프로그램을 서블릿이라고 함
- .java확장자로 자바를 이용하여 웹 어플리케이션을 만들기 위해 제공되는 기술
- 클라이언트 요청에 동적으로 작동하고, 응답은 html을 이용
- MVC패턴에서 Controller로 이용됨



## □ Servlet의 장점

- Java를 사용해서 동적 웹 어플리케이션을 구현할 수 있다.
- 플랫폼에 독립적이다.
- 서버에 독립적이다.
- 프로세스 기반이 아닌 스레드 기반이기 때문에 더 효과적으로 멀티 태스킹을 지원한다. (프로세스 대비 최소 5배 빠름)



## □ Servlet 단점

- 화면에 표현될 HTML코드를 프로그램적으로 작성해야 한다.
- 서비스하기 전에 반드시 컴파일을 해야한다.



## □ Servlet 활용 시 좋은 점

- 콘텐츠와 비즈니스 로직을 분리할 수 있다.
- 컨트롤러와 뷰 역할 분담으로 퍼블리셔(프론트엔드)와 협업이 가능하다.
- 유지 보수가 용이하다.
- 기능 확장이 용이하다.



## □ Java Servlet API

- 서블릿은 응답/요청 모델에서 작동되도록 설계되어있다.
- 응답/요청 모델에서 클라이언트는 서버에게 요청 메시지를 보내고, 서버는 이에 대한 응답으로 메시지를 돌려보낸다.
- GET 요청과 POST 요청 등 HTTP Method에 따라 작동이 다르다.



- HTML 페이지의 form 태그를 이용하여 http method를 호출 할 수 있다.
- get 요청은 servlet의 doGet으로 처리하고
  - 가장 많이 사용되며 사이트 방문은 모두 get으로 사용한다.
- post 요청은 servlet의 doPost으로 처리한다.
  - 웹서버에 데이터를 전송할 때 사용된다.



- method에는 방식을 기술하여
- action에는 서블릿이나 요청을 받을 페이지를 기술하여 서버로 요청을 보낸다.





## □ HttpServletRequest 객체

- 요청에 대한 정보를 담고 있는 객체
- `getHeader()`: 요청 정보의 헤더정보를 반환한다.
- `getMethod()`: 요청 정보의 HTTP Method를 반환한다.
- `getParameter()`: 요청 시 정보의 이름으로 요청 값을 반환한다.
- `setCharacterEncoding()`: 요청정보의 인코딩을 설정한다.



## □ HttpServletResponse 객체

- 응답에 대한 정보를 담고 있는 객체
- `setContentType()`: 클라이언트로 보내는 콘텐츠 타입을 설정
- `getWriter()`: `PrintWriter` 객체를 반환하여 클라이언트에게 전달할 텍스트 데이터를 작성한다.
- `setStatus()`: 응답에 대한 상태코드를 지정한다.



## □ JSP(Java Server Page)

- HTML 페이지 내에 자바 코드를 삽입하여 웹 서버에서 동적으로 웹 페이지를 생성하여 클라이언트에 전달하는 언어
- .jsp 확장자이며 html코드와 java코드가 동시에 존재
- 클라이언트 요청에 동적으로 작동하여 응답은 html을 이용
- jsp는 서블릿으로 변환되어 실행
- MVC패턴에서 View로 이용됨



## □ Servlet과 비교

- Servlet은 Java소스에 HTML코드가 삽입
- JSP는 HTML코드에 Java코드가 삽입
- Sevlet class는 컴파일과 web.xml(Java config)에 등록하는 과정이 필요하지만 JSP는 필요 없다.
- Servlet은 JSP로 대체될 수 있고, JSP는 Servlet으로 대체될 수 있다. (보완의 관계)



## □ JSP 장점

- Java 코드에 HTML코드가 삽입되는 문제를 해결
- 퍼블리셔와 협업이 가능하다.
- 유지보수가 용이하다.
- 간단한 로직을 구현할 때는 JSP가 더 간편하다.



## □ JSP 단점

- 복잡한 로직을 구현할 때 HTML 중심의 코드가 이해하기 어렵게 만들 수 있다.
- 프로그래밍 언어를 모르는 사람이 실수로 코드를 건드릴 수 있다.
- Core적인 부분의 로직의 유출을 막기 위해서 Java단에서 로직을 작성하는게 보안에 더 유리하다.