

웹 프로그래밍 / 김지원 교수

| 05

서블릿 기초 문법



□ 강의교안 이용 안내

- 본 강의교안의 저작권이 있습니다.
 - 이 자료는 강의 보조자료로 제공되는 것으로 무단으로 전제하거나 배포하는 것을 금합니다.
-



□ 서블릿 기초 문법

05

학습목표



서블릿 기초 문법
서블릿의 기본 문법과 메소드

□ 서블릿 기초 문법

학습내용

01

서블릿 기초 문법

□ 서블릿의 개념

□ 서블릿의 주요 클래스와 메소드

□ 서블릿의 요청방식

01 서블릿

■ 서블릿이란?

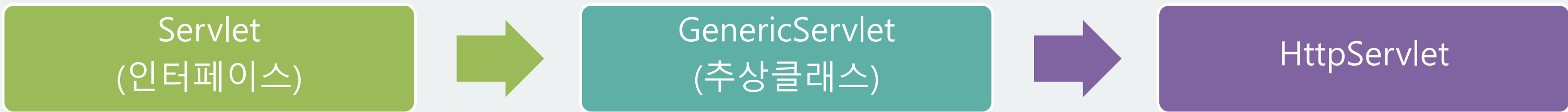
- JSP 이전에 동적인 웹페이지 콘텐츠를 생성하는 기술로 제공
- 서블릿의 자바의 많은 장점 사용가능
- HTML, CSS 및 자바스크립트 그리고 웹디자인과 연동하기에는 많은 단점
- 단점이 많지만 서블릿의 존재 이유
 - JSP에는 없는 서버 측 프로그램의 기능
 - 대규모 프로젝트에 사용되는 프레임워크의 기술 사용

01 서블릿

- 서블릿에서 제공되는 주요 클래스

- HttpServlet

- 서블릿을 만들기 위해 반드시 상속해야 할 필수 클래스구성



[표] HttpServlet에서 제공되는 주요 메소드

메소드	설명
void init()	서블릿의 객체가 생성될 때 호출되는 메소드
void destroy()	서블릿의 객체가 메모리에서 사라질 때 호출되는 메소드
void service(request, response)	서블릿의 요청이 있을 때 호출되는 메소드
void doGet(request, response)	html에서 form의 메소드가 get일 때 호출되는 메소드
void doPost(request, reponse)	html에서 form의 메소드가 post일 때 호출되는 메소드

02 서블릿의 주요 클래스의 메소드

- **HttpServletRequest**
 - 클라이언트가 데이터를 입력하거나 또는 정보에 대한 요청 값을 가지고 있는 클래스

[표] HttpServletRequest에서 제공되는 주요 메소드

메소드	설명
String getParameter(name)	name에 할당된 값을 반환하며 지정된 파라미터 값이 없으면 null값을 반환
String[] getParameterValues(name)	name의 모든 값을 String 배열로 반환
Enumeration getParameterNames()	요청에 사용된 모든 파라미터 이름을 java.util.Enumeration타입으로 반환
void setCharacterEncoding(env)	post방식으로 요청된 문자열의 character encoding을 설정

02 서블릿의 주요 클래스의 메소드

- **HttpServletResponse**
 - 클라이언트가 요청한 정보를 처리하고 다시 응답하기 위한 정보를 담고 있는 클래스

[표] HttpServletResponse에서 제공되는 주요 메소드

메소드	설명
void setHeader(name, value)	응답에 포함될 Header 설정
void setContentType(type)	출력되는 페이지의 contentType 설정
String getCharacterEncoding()	응답 페이지의 문자 인코딩 Type 반환
void sendRedirect(url)	지정된 URL로 요청 재전송

02 서블릿의 주요 클래스의 메소드

- **HttpSession**
 - 클라이언트가 세션을 정보를 저장하고 세션 기능을 유지 하지 위해서 제공되는 클래스

[표] HttpSession에서 제공되는 주요 메소드

메소드	설명
String getId()	해당 세션의 세션 ID반환
long getCreationTime()	세션의 생성된 시간 반환
long getLastAccessedTime()	클라이언트 요청이 마지막으로 시도된 시간 반환
void setMaxInactiveInterval(time)	세션을 유지할 시간을 초단위로 설정
int getMaxInactiveInterval()	setMaxInactiveInterval(time)로 지정된 값 반환(기본값 : 30분)
boolean is New()	클라이언트 세션 ID를 할당하지 않은 경우 true값 반환
void invalidate()	해당 세션 종료

03 서블릿의 라이프 사이클

■ 서블릿의 라이프 사이클

클라이언트 서블릿 요청 -> init(최초 한번 호출) ->
service, doGet, doPost(반복적 호출) -> destroy(마지막 한번 호출)

- init() : 서블릿이 처음으로 요청이 되어 객체가 생성 될 때 호출되는 메소드
- service() : 클라이언트가 요청이 있을 때 마다 호출되는 메소드
- destroy() : 서블릿 서비스의 종료 및 재시작 그리고 서블릿 코드가 수정이 될 때 호출되는 메소드
- doGet() : 클라이언트 요청 할 때 form의 method가 get 방식일 때 호출되는 메소드
- doPost() : 클라이언트 요청 할 때 form의 method가 post 방식일 때 호출되는 메소드

04 서블릿의 요청방식

- **get 방식**

- 일반적으로 검색이나 조건을 다른 페이지로 전송할 때 사용하는 방식
- url에 노출되는 특징이 있다

- **post 방식**

- 게시판에 글을 입력하거나 또는 회원가입을 하는 기능으로 다른 페이지로 전송할 때 사용하는 방식
- html 헤더 값에 같이 전송되어 url에 노출되지 않는다

05 서블릿의 세션

- 세션(session)
 - 클라이언트(브라우저)와 서버(톰캣)와 통신의 연속성을 유지하기 위해 사용 되는 기술



- ☐ 웹 프로그래밍
- ☐ 컴퓨터공학과 ☒ 김지원

수고하셨습니다

질문이 있으시다면 Cyber Compus의 QnA에 올려주세요!
감사합니다.

