**로고학습일지**

**kt ds University 자바 기반의 데이터 사이언티스트 양성과정**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 학습일시 | 2018. 08. 13 (월) | 장소 | kt ds University B관 201호 | **시 간** | 09:00~18:00 |
| 학습범위 |  | | | | |
| 작 성 자 | 신예리 | | | **강 사** | 장민창 강사 |

|  |  |
| --- | --- |
| 학습안건 | Servlet |

|  |  |
| --- | --- |
| 학습내용 | 내용 |
| Servlet 서블릿 🡪 이제 실제로 거의 사용되지 않는다.  🡪 메인 메소드가 필요하지 않다.  <HTML>  https://www.w3schools.com > REFERENCES > HTML Element Reference > HTML Tags 보고 공부하기  <Tomcat>  C:\Users\Admin\Desktop\새 폴더\SW\apache-tomcat-8.0.53 경로 복사  STS > View [Servers] > 기존에 있는 거 삭제 > 링크 클릭 > Apache - Tomcat v8.0 Server 선택  > Tomcat installation directory : C:\Users\Admin\Desktop\새 폴더\SW\apache-tomcat-8.0.53  > JRE : jdk-10.0.1  ※프로젝트 오른쪽 클릭 > Project Close ( Servers는 지우면 안대)  웹 프로젝트 생성 :  New > Dynamic Web Project > 이름 설정 > Dynamic web module version : 2.5 > next 클릭 > src 폴더 삭제  > Add Folder : src/main/java | src/main/resources | src/test/java | src/test/resources > next 클릭  > Content directory : src/main/webapp > Generate web.xml deployment descriptor 반드시 체크! > Finish  -생성된 폴더에 지구 모양 : Web Application  -src > main > webapp > WEB-INF > lib & web.xml  webapp부터 web.xml : content (jsp, css, js, img) & **content path** 🡺 webapp은 브라우저에서 접근 가능  WEB-INF : 설장 🡺 브라우저에서 접근 불가능  -src/main/java : java code 🡪 Servlet / Service, Dao  -src/main/resources : 자원들 (Servlet에서는 거의 사용되지 않아. 그냥 존재만 하는 폴더)  -src/test/java : src/main/java의 테스트 코드 🡺 jUnit  -src/test/resources : src/test/java를 위한 자원들 (거의 사용되지 않음)  Servlet 파일 생성  src/main/java 아래 \_\_\_\_패키지 안에 Servlet 만들기 > 이름 > URL mappings : /index  (브라우저에 http://localhost:8080/Hello-Web/index 입력하면 IndexServlet 실행됨) > Finish  web.xml : 프로젝트에 대한 전반적인 설정 (절대 지우면 안되욧!!!!!)  > Source > <servlet>  <servlet>  <description></description>  <display-name>IndexServlet</display-name>  <servlet-name>IndexServlet</servlet-name> : Servlet 이름  <servlet-class>com.tkds.index.web.IndexServlet</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>IndexServlet</servlet-name> : Servlet 이어지는  <url-pattern>/index</url-pattern>  </servlet-mapping>  Hello-Web과 Tomcat 8.0서버 연결  View [Servers] > Tomcat v8.0 Server at localhost 오른쪽 클릭 > Add and Remove > Configured로 Add후 Finish  ["http-nio-8080"] : http 포트 번호 8080  Server start후 http://localhost:8080/Hello-Web/index 입력하면 브라우저로 들어가기 가능  JSP 파일 생성  src > main > webapp > WEB-INF > view 폴더 만들어 그 안에 jsp 파일 만들기  >context path 부분은 브라우저에서 접근이 불가능 하지만 servlet을 사용하여 가능하게 만듬  -Servlet의 doGet에서 처리  protected void doGet(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException {  // JSP를 보여주는 역할  // Context path : src/main/webapp  String path = "/WEB-INF/view/index/index.jsp"; // JSP 경로  RequestDispatcher rd = request.getRequestDispatcher(path);    // Browser에 표시  rd.forward(request, response);  }  -index.jsp  <%@ page language="java" contentType="text/html; charset=UTF-8"pageEncoding="UTF-8"%> <!-- Directive  <%@ page...%> : 페이지 Directive (이 페이지에 대한) -->  <!DOCTYPE html> <!-- HTML5를 쓰겠다.  html\_\_\_\_ 뒤에 뭔가 적혀있다면 HTML4를 쓴다는 의미 -->  <html> <!-- <...> : 태그 (늘 한쌍)  <html> : html 여는 태그 -->  <head> <!-- 설정과 관련된 부분 -->  <meta charset="UTF-8"> <!-- Single Line Tag : 닫는 태그가 없다. (태그 사이에 적는 것이 없다)  <.../> 이렇게 적어주는 것이 정석  UTF-8 : 다국어 => 다국어를 쓰기 위한 인코딩 -->    <title>Insert title here</title> <!-- 브라우저 위(?)에 보이는 부분 (탭 이름...) -->  </head>  <body> <!-- 우리가 실제로 보는 content -->  신예리  </body>  </html> <!-- html 닫는 태그 -->  \*만들어 보기!  http://localhost:8080/Hello-Web/me  URL : /me  JSP Path : /WEB-INF/view/me/me.jsp  1. 이름 : 신예리  2. 전화번호 : 010-1234-5678  http : Hyper Text Transfer Protocol (Hyper Text 전송 규약)  HTML <--> HTML : Hyper Link (서로 다른 문서들이 연결)  > http://localhost:8080/Hello-Web/me 입력했을 때  -/me , /index  Client (Browser) Server (Servlet)  |<-------------------------------->| 1. TCP/IP Connect  | |  |---------------------------------->| 2. Request : URL => http://localhost:8080/Hello-Web/me  | |  |<----------------------------------| 3. Response : JSP > HTML  | |  |<-------------------------------->| 4. TCP/IP Close  🡪 2-3번 사이의 처리 1. URL Mapping  2. web.xml > url-pattern  3. Servlet find  4. JSP Parsing  JSP : 박스로 처리해 블럭을 통해 줄바꿈 처리함 (<br/> 사용하지 말것)  >Block  -Line Block : 처음부터 끝까지 채워지는 (body)  -Inline Block : 컨텐츠의 내용만큼만 채워지는  \* Line 블럭도 Inline처럼 만들 수 있다.    > <div> </div>  <p> </p> 를 사용해서 줄바꿈 처리  -내용에 스페이스를 주었을 때 너무 길면 줄바꿈이 일어나지만 스페이스가 없다면 스크롤 바가 생기고 줄바꿈이 일어나지 않는다  > 블럭 안에 블럭 넣기 가능 Wrapper Box  > Table을 쓰지 않아 -> div 많이 사용 : 반응형 웹에 적합하지 않아서 |

|  |  |
| --- | --- |
| 특이사항 | <https://www.w3schools.com> |