

2025년 상반기 K-디지털 트레이닝

EL, JSTL

[KB] IT's Your Life



🗸 ex05라는 이름으로 Dynamic Web 프로젝트를 만드세요.

☑ login_form.jsp를 다음과 같이 정의하세요.

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<body>
<form action="login" method="get">
  <fieldset>
    <legend>로그인 폼</legend>
    <l
       <
         <label for="userid">0101CI</label>
         <input type="text" id="userid" name="userid">
       <
         <a href="passwd">비밀번호</a>label>
         <input type="password" id="passwd" name="passwd">
       -로그인 폼
       <|i>
         <input type="submit" value="전송">
       • 아이디
    </fieldset>
                                                            • 비밀번호
</form>
                                                               전송
</body>
</html>
```

다음 설정으로 LoginServlet을 작성하세요.

- o 패키지: org.scoula.ex05
- o url 맵핑: /login
- userid, passwd 파라미터를 추출하여 request 스코프에 저장
- o login.jsp로 포워딩

LoginServlet.java

```
package org.scoula.ex05;
@WebServlet("/login")
public class LoginServlet extends HttpServlet {
  protected void doGet(HttpServletRequest req, HttpServletResponse res) throws IOException, ServletException {
     String userid = req.getParameter("userid");
     String passwd = req.getParameter("passwd");
     req.setAttribute("userid", userid);
     req.setAttribute("passwd", passwd);
     req.getRequestDispatcher("login.jsp").forward(req, res);
```

♡ 다음과 같이 login.jsp를 작성하여 Scope에 저장된 데이터를 출력한다.

EL 실습

사용자 아이디: hong 사용자 비밀번호: 1234

- 💿 org.scoula.ex05.doman 패키지에 Member 클래스를 추가하세요.
 - ㅇ 필드
 - String name
 - String userid
 - 생성자
 - 기본 생성자, 전체 매개변수 생성자
 - o Getter/Setter 메서드

Member.java

```
package org.scoula.ex05.domain;
                                                                    this.name = name;
public class Member {
  private String name;
                                                                          public String getUserid() {
  private String userid;
                                                                    return userid;
  public Member() {
                                                                          public void setUserid(String userid) {
                                                                    this.userid = userid;
  public Member(String name, String userid) {
     this.name = name;
     this.userid = userid;
  public String getName() {
     return name;
           public void setName(String name) {
```

다음과 같이 ScopeServlet을 작성하세요.

```
@WebServlet("/scope")
public class ScopeServlet extends HttpServlet {
  ServletContext sc;
  @Override
  public void init(ServletConfig config) throws ServletException {
     sc = config.getServletContext();
  protected void doGet(HttpServletRequest reg, HttpServletResponse res) throws IOException, ServletException {
     sc.setAttribute("scopeName", "applicationScope 값"); // Application Scope
     HttpSession session = reg.getSession(); // Session Scope
     session.setAttribute("scopeName", "sessionScope 값");
     reg.setAttribute("scopeName", "requestScope 값"); // Request Scope
     Member member = new Member("홍길동", "hong");
     req.setAttribute("member", member);
     req.getRequestDispatcher("scope.jsp").forward(req, res);
```

- ☑ scope.jsp를 작성하여 Scope에 저장된 데이터를 출력하세요.
 - 요청 url:/scope

scope 데이터 보기

pageScope의 속성값은 :

requestScope의 속성값은 : requestScope 값 sessionScope의 속성값은 : sessionScope 값

applicationScope의 속성값은 : applicationScope 값

scopeName 자동 찾기: requestScope 값

member: 홍길동(hong)

scope.jsp

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
  <title>Title</title>
</head>
<body>
  <h1>scope 데이터 보기 </h1>
  pageScope의 속성값은: ${pageScope.scopeName}<br>
  requestScope의 속성값은: ${requestScope.scopeName}<br/>br>
  sessionScope의 속성값은: ${sessionScope.scopeName}<br>
  applicationScope의 속성값은: ${applicationScope.scopeName}<br>
  scopeName 자동 찾기: ${scopeName} <br>
  member: ${member.name}(${member.userid})<br>
</body>
</html>
</html>
```