

2025년 상반기 K-디지털 트레이닝

요청 포워딩, 리다이렉트 (심화1)

[KB] IT's Your Life



- ♡ 다음 설정으로 RequestServlet을 작성하세요.
 - o 패키지: org.scoula.ex04
 - o URL 맵핑: /request
 - request 스코프에 다음 속성을 설정함
 - 속성 명: 속성 값
 - username: 홍길동
 - useraddress: 서울
 - response.jsp로 포워딩함

RequestServlet.java

```
@WebServlet("/request")
public class RequestServlet extends HttpServlet {
          protected void doGet(HttpServletRequest req,
                                                                          HttpServletResponse resp)
                                                     throws ServletException, IOException {
                     //속성 설정
                     req.setAttribute("username", "홍길동");
                     req.setAttribute("useraddress", "서울");
                     //forward
                     RequestDispatcher dis = req.getRequestDispatcher("/response.jsp");
                     dis.forward(req, resp);
```

- ▽ 다음을 처리하는 response.jsp를 작성하세요.
 - o request 스코프에 저장된 username, useraddress 속성을 출력함
 - /request 요청으로 결과를 확인함

username 값: 홍길동

useraddress 값: 서울

response.jsp

- 다음과 같은 설정으로 RequestRedirectServlet 클래스를 작성하세요.
 - o 패키지: org.scoula.ex04
 - o URL 맵핑: /request_redirect
 - request 스코프에 다음 속성을 설정함
 - 속성 명: 속성 값
 - username: 홍길동
 - useraddress: 서울
 - response_redirect로 리다이렉트 함

RequestRedirectServlet.java

- 다음과 같은 설정으로 ResponseRedirectServlet 클래스를 작성하세요.
 - o 패키지: org.scoula.ex04
 - o URL 맵핑: /response_redirect
 - request 스코프에 다음 속성을 추출함
 - username, useraddress
 - 추출한 속성을 다시 request 스코프에 저장함
 - o redirect_response.jsp로 포워딩함

ResponseRedirectServlet.java

```
@WebServlet("/response redirect")
public class ResponseRedirectServlet extends HttpServlet {
           protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws IOException, ServletException {
                     String username = (String) req.getAttribute("username");
                     String useraddress = (String) req.getAttribute("useraddress");
                     //속성 설정
                     req.setAttribute("username", username);
                     req.setAttribute("useraddress", useraddress);
                     //forward
                     RequestDispatcher dis = req.getRequestDispatcher("/redirect_response.jsp");
                     dis.forward(req, resp);
```

- 포워딩 된 redirect_response.jsp를 작성하세요.
 - o request 스코프에 있는 username과 useraddress를 출력함

- ☑ /request_redirect로 요청했을 때 예상 결과와 비교하고, 이렇게 나온 이유를 설명하세요.
 - 브라우저 주소창의 최종 url을 확인함

username 값: null

useraddress 값: null

redirect_response.jsp



2025년 상반기 K-디지털 트레이닝

EL, JSTL (심화1)

[KB] IT's Your Life



💟 다음 설정으로 JstlServlet을 작성하세요.

- o 패키지: org.scoula.ex05
- o 맵핑 url: /jstl_ex
- o request 속성
 - "role": "ADMIN"
 - "members": ArrayList<Member>
 - for 루프를 이용하여 10개의 Member 데이터 생성해서 members 리스트에 추가
- /jstl_ex.jsp로 포워딩

JstlServlet.java

```
@WebServlet("/jstl_ex")
public class JstlServlet extends HttpServlet {
  protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws IOException, ServletException {
     List<Member> members = new ArrayList<>();
     for (int i = 0; i < 10; i++) {
        Member member = new Member("홍길동_" + i, "hong_" + i);
        members.add(member);
     req.setAttribute("members", members);
     req.setAttribute("role", "ADMIN");
     req.getRequestDispatcher("jstl_ex.jsp")
       .forward(req, resp);
```

- ♡ 다음을 처리하는 jstl_ex.jsp를 작성하세요.
 - role 속성값 검사
 - ADMIN이면 "관리자" 출력
 - ADMIN이 아니면 "일반 회원" 출력
 - o members 목록을 이용하여 로 출력

JSTL 테스트

관리자

0 홍길동_0 hong_0

1 홍길동_1 hong_1

2 홍길동_2 hong_2

3 홍길동_3 hong_3

4 홍길동_4 hong_4

5 홍길동_5 hong_5

6 홍길동_6 hong_6

7 홍길동_7 hong_7

8 홍길동_8 hong_8

9 홍길동_9 hong_9

jstl_ex.jsp

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
  <title>Title</title>
</head>
<body>
  <h1>JSTL 테스트</h1>
  <c:if test="${role == 'ADMIN'}">관리자</c:if>
  <c:if test="${role != 'ADMIN'}">일반회원</c:if>
  <c:forEach var="member" items="${members}" varStatus="state">
       ${state.index}
         ${member.name}
         ${member.userid}
       </c:forEach>
  <br>
</body>
</html>
```