

1. 연습문제

A. request를 활용한 데이터 전송 예제

Request를 활용해 데이터의 이동 예제

```
<form action="Example1.jsp" method="post">
  <label for="id">ID : </label>
  <input type="text" id="id" name="id"/>
  <br />
  <label for="pw">PW : </label>
  <input type="password" pw="pw" name="pw"/>
  <br />
  <button type="submit">전송</button>
</form>
```

```
<%
  request.setCharacterEncoding("UTF-8");
  String id = request.getParameter("id");
  String pw = request.getParameter("pw");
%>
```

```
<h1>안녕하세요 <%=id%>님</h1>
<h1>패스워드 : <%=pw%></h1>
```

ID :

PW :

안녕하세요 김근형님

패스워드 : 1234

=>

B. request 의 값 관련 메서드

Request 값 관련 메서드

메소드명	설명
public String getParameter(String name)	요청 파라미터 name에 할당된 값을 리턴한다.
public Enumeration getParameterNames()	요청에 사용된 모든 파라미터 이름을 리턴한다.
public String[] getParameterValues(String name)	지정된 파라미터 이름에 대한 값들을 리턴한다. 같은 name 속성으로 지정된 값들은 이 메서드를 사용하여 읽어낼 수 있다.

C. 연습 문제

- i. 다음 아래의 버튼을 클릭하면 다음과 같은 페이지로 이동하는 웹을 만들어라

1. 첫 페이지

HOME

공지사항

BBS

- 1-1. home을 눌렀을 시

home

- 1-2. 공지사항을 눌렀을 시

공지사항

- 1-3. BBS를 눌렀을 시

BBS

ii. 아래 요청사항의 페이지 이동을 구현하라

아래 테이블을 완성하되 데이터들이 전부 자바의 변수에 저장되어 있다
가정하고 만들어라

페이지 이동

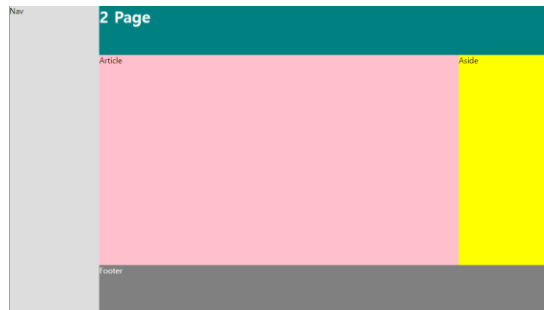
1페이지 ▾

이동

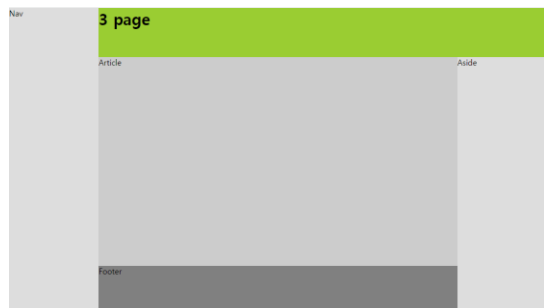
1페이지 이동 시



2페이지 이동 시



3페이지 이동 시



2. 연습문제

A. response와 pageContext에 대한 설명

Response 객체

리턴타입	메소드명	설명
없음	setHeader(String headerName, String headerValue)	응답에 포함될 헤더 정보에 headerName의 이름으로 headerValue 값을 설정해 추가한다.
없음	addCookie(Cookie cookie)	javax.servlet.http.Cookie 타입의 쿠키 객체를 응답 헤더에 추가한다.
없음	sendRedirect(String url)	지정된 URL로 요청을 재전송한다
없음	setContentType(String type)	응답 페이지의 contentType을 설정한다

pageContext 객체

리턴 타입	메소드 명	설명
ServletRequest	getRequest()	클라이언트의 요청 정보를 담고 있는 객체 (request)를 리턴한다
ServletResponse	getResponse()	요청에 대한 응답 객체(response)를 리턴한다
JspWriter	getOut()	응답 출력 스트림을 리턴한다(out)
Object	getPage()	서블릿 인스턴스 객체를 리턴한다(page)
ServletConfig	getServletConfig()	서블릿의 초기 설정 정보를 담고 있는 객체를 리턴한다(config)
ServletContext	getServletContext()	서블릿의 실행 환경 정보를 담고 있는 객체를 리턴한다(application)

pageContext 객체

리턴 타입	메소드 명	설명
HttpSession	getSession()	클라이언트의 세션 정보를 담고 있는 객체를 리턴한다(session)
없음	forward(String url)	현재 페이지의 요청과 응답에 관한 제어권을 URL로 지정된 주소로 영구적으로 넘긴다. forward 페이지의 요청 처리가 종료되면 응답도 종료된다.
없음	include(String url)	현재 페이지의 요청과 응답에 관한 제어권을 URL로 지정된 주소로 임시로 넘긴다. include 된 페이지의 처리가 끝나면 제어권은 다시 원래대로 돌아온다. 따라서 include로 지정된 페이지의 내용을 원래 페이지에 삽입하는 효과를 가진다.

redirect와 forward의 구현 예제

- Redirect

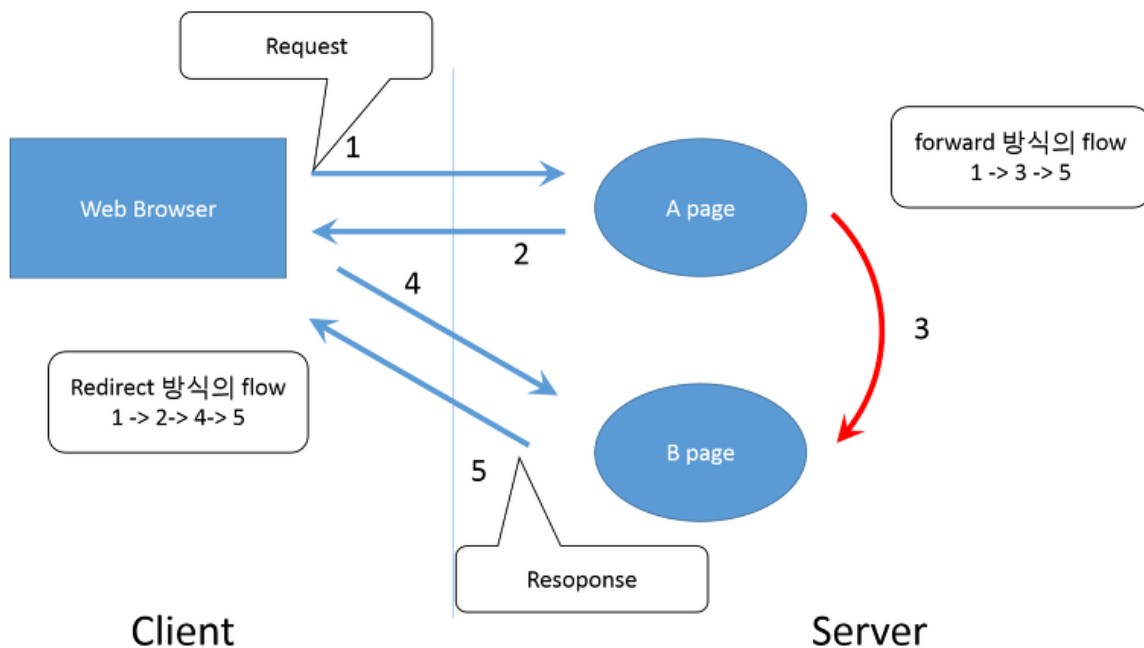
```
<%  
response.sendRedirect("responseTest2.jsp");  
%>
```

```
<h1>  
현재 페이지는 responseTest1.jsp에 의해<br>  
리다이렉트 된 페이지입니다.  
</h1>
```

- forward

```
<%  
pageContext.forward("pageContextTest2.jsp");  
%>
```

B. Redirect 와 forward 의 차이점



액션 태그의 forward

▶ forward 액션

```
<jsp:forward page="이동할 페이지"/>
```

- ▶ `pageContext` 내장 객체의 `forward()` 메소드가 태그로 구현된 기능.
- ▶ forward 액션은 현재 페이지의 요청과 응답에 관한 처리권을 page 속성에 지정된 이동할 페이지로 영구적으로 넘기는 기능을 한다.
- ▶ `request`와 `response` 객체가 그대로 넘어가며 따라서 사용자가 request에서 지정한 속성들은 포워딩 된 페이지에서도 그대로 사용이 가능하다.

▶ forward 액션

- ▶ forward 태그를 사용하여 이동할 페이지에 추가적으로 파라미터를 넘겨줄 필요가 있을 때에는 다음처럼 forward 태그의 하위 태그인 `<jsp:param/>` 태그를 사용할 수 있다.

```
<jsp:forward page="이동할 페이지">  
  <jsp:param name="파라미터 이름1" value="파라미터 값1"/>  
  <jsp:param name="파라미터 이름2" value="파라미터 값2"/>  
</jsp:forward>
```

- ▶ `<jsp:param />` 태그를 사용하여 지정한 파라미터는 Get 방식으로 전송할 때처럼 이동할 페이지의 주소 뒤에 직접 붙여서 전송하는 것도 가능하다.

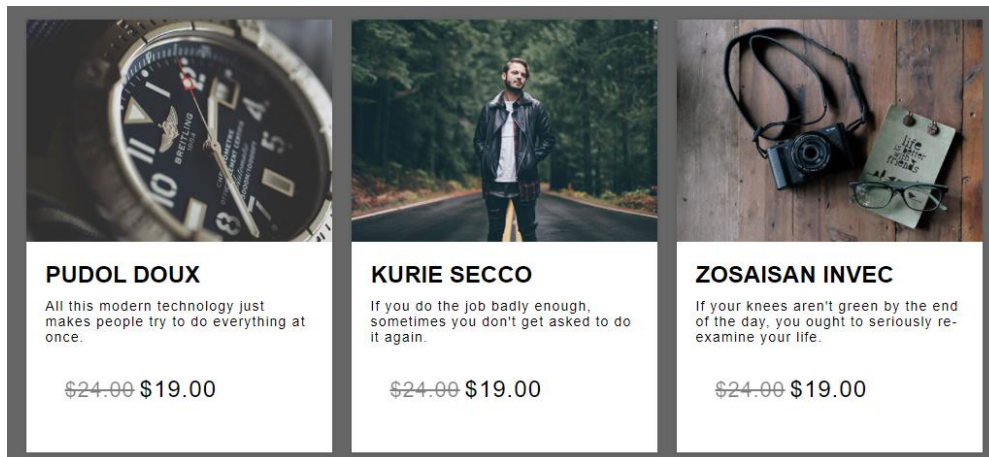
```
<jsp:forward page="forward.jsp?id=kimgh&password=abc" />
```

C. 연습 문제

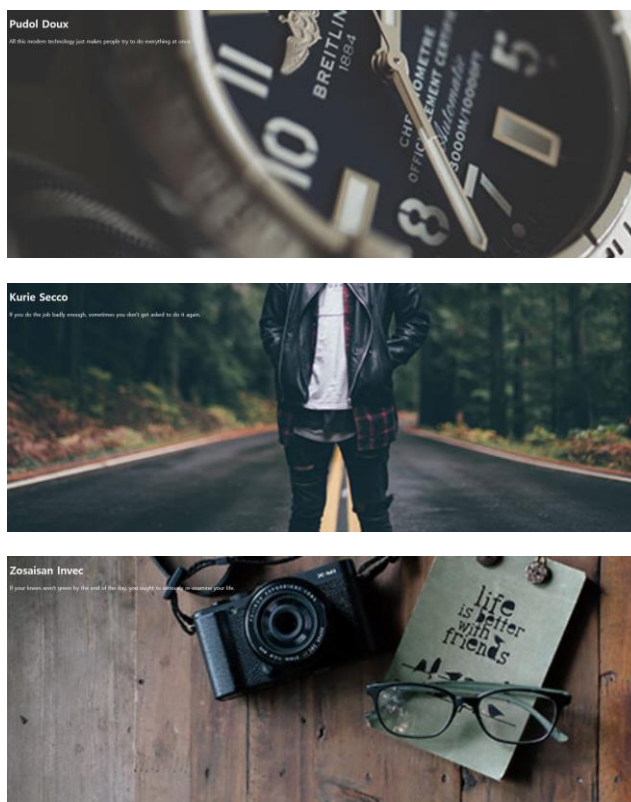
i. Redirect 페이지로 만드는 상품 링크 페이지

include와 forward를 사용하여 다음과 같이 만들어라

1. 첫 페이지



2. 각 페이지 클릭 시



※ resource

<https://s3-us-west-2.amazonaws.com/s.cdpn.io/331810/sample85.jpg>

<https://s3-us-west-2.amazonaws.com/s.cdpn.io/331810/sample96.jpg>

<https://s3-us-west-2.amazonaws.com/s.cdpn.io/331810/sample92.jpg>