

# JSP 웹 프로그래밍

세션

장바구니 페이지 만들기

## 세션(session)

---

- ▶ 클라이언트와 웹 서버 간의 상태를 지속적으로 유지하는 방법
  - ▶ 예) 웹 쇼핑몰에서 장바구니나 주문 처리
  - ▶ 사용자 인증을 통해 특정 페이지를 사용할 수 있도록 권한 상태를 유지하는 것.
- ▶ 웹 서버에서만 접근이 가능하므로 보안 유지에 유리하며 데이터를 저장하는 데 한계가 없음



## 세션(session)

---

- ▶ 웹 서버에 존재하는 객체로 웹 브라우저마다 하나씩 존재
- ▶ 웹 서버의 서비스를 제공받는 사용자를 구분하는 단위가 됨
- ▶ 웹 브라우저를 닫기 전까지 웹 페이지를 이동하더라도 사용자의 정보가 웹 서버에 보관되어 있어 사용자 정보를 잃지 않음.



# 세션 생성

- ▶ `setAttribute( )` 메소드를 사용하여 세션의 속성을 설정
  - 계속 세션 상태를 유지할 수 있음.
- ▶ 동일한 세션의 속성 이름으로 세션을 생성하면 마지막에 설정한 것이 세션 속성 값이 됨.

```
void setAttribute(String name, Object value)
```

## ▶ 매개변수

- ▶ `name` - 세션 속성 이름
  - 세션에 저장된 특정 값을 찾아오기 위한 키.
- ▶ `value` - 세션의 속성 값
  - 속성 값은 `Object` 객체 타입만 가능하기 때문에 `int`, `double`, `char` 등의 기본 타입은 사용할 수 없음
- ▶ [`setAttribute()` 메소드 사용 예]

```
session.setAttribute("memberId", "admin");
```

# 세션 생성하기

session01.jsp

```
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <html>
3 <head>
4 <title>Session</title>
5 </head>
6 <body>
7 <form action="session01_process.jsp" method="POST">
8   <p>아이디 : <input type="text" name="id">
9   <p>비밀번호 : <input type="text" name="passwd">
10  <p><input type="submit" value="전송">
11 </form>
12 </body>
13 </html>
```

http://localhost:8090/JSPTest/chapter13/session01.jsp

아이디 : admin

비밀번호 : 1234

전송

# 세션 생성하기

session01\_process.jsp

```
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <html>
3 <head>
4 <title>Session</title>
5 </head>
6 <body>
7 <%
8     String userId = request.getParameter("id");
9     String userPwd = request.getParameter("passwd");
10    if(userId.equals("admin") && userPwd.equals("1234")){
11        session.setAttribute("userID", userId);
12        session.setAttribute("userPW", userPwd);
13        out.println("세션 설정이 성공했습니다<br>");
14        out.println(userId + "님 환영합니다.");
15    }else{
16        out.println("세션 설정이 실패했습니다.");
17    }
18    %>
19 </body>
20 </html>
```

http://localhost:8090/JSPTest/chapter13/session01\_process.jsp

세션 설정이 성공했습니다  
admin님 환영합니다.

# 세션 정보

---

## ▶ 단일 세션 정보 얻기

### ▶ `getAttribute( )` 메소드

- ▶ 세션에 저장된 세션 속성 이름에 대한 속성 값을 얻어 옴
- ▶ 반환 유형이 `Object` 형이므로 반드시 형 변환을 하여 사용
- ▶ 형식:

- `Object getAttribute(String name)`

- ▶ 매개변수 `name` - 세션에 저장된 세션 속성 이름
- ▶ 해당 속성 이름이 없는 경우 `null`을 반환

### ▶ `getAttribute( )` 메소드 사용 예

```
String id = (String)session.getAttribute("memberId");
```



# 세션에 저장된 속성 값 가져오기

```
session02.jsp
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <html>
3 <head>
4 <title>Session</title>
5 </head>
6 <body>
7 <%
8     String userId = (String) session.getAttribute("userID");
9     String userPwd = (String) session.getAttribute("userPW");
10
11     out.println("설정된 세션의 속성 값 [1] : " + userId + "<br>");
12     out.println("설정된 세션의 속성 값 [2] : " + userPwd);
13 %>
14 </body>
15 </html>
```

← → □ 📌 ▼ http://localhost:8090/JSPTest/chapter13/session02.jsp

설정된 세션의 속성 값 [1] : admin  
설정된 세션의 속성 값 [2] : 1234



# 세션 정보

---

## ▶ 다중 세션 정보 얻기

```
Enumeration getAttributeNames();
```

## ▶ getAttributeNames()메소드 사용 예

```
Enumeration enum = session.getAttributeNames();

while(enum.hasMoreElements()) {
    String name = enum.nextElement().toString();
    String value = session.getAttribute(name).toString();
}
```



# 세션 정보

- ▶ 세션에 저장된 모든 세션 속성 이름과 속성 값 가져와 출력하기

session03.jsp

```
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <%@ page import="java.util.Enumeration"%>
3 <html>
4 <head>
5 <title>Session</title>
6 </head>
7 <body>
8 <%
9     String name;
10    String value;
11    Enumeration<String> en = session.getAttributeNames();
12    int i = 0;
13    while(en.hasMoreElements()){
14        i++;
15        name = en.nextElement();
16        value=session.getAttribute(name).toString();
17        out.println("설정된 세션의 속성 이름[" + i + "] : " + name + "<br>");
18        out.println("설정된 세션의 속성 값[" + i + "] : " + value + "<br>");
19    }
20    %>
21 </body>
22 </html>
```

http://localhost:8090/JSPTest/Chapter13/session03.jsp

설정된 세션의 속성 이름[1] : userPW  
설정된 세션의 속성 값[1] : 1234  
설정된 세션의 속성 이름[2] : userID  
설정된 세션의 속성 값[2] : admin

## 단일 세션 삭제

---

- ▶ removeAttribute( ) 메소드를 사용해 세션에 저장된 하나의 세션 속성 이름을 삭제

```
void removeAttribute(String name)
```

- ▶ removeAttribute( ) 메소드 사용 예

```
session.removeAttribute("memberId");
```



# 단일 세션 삭제

## ▶ 세션에 저장된 세션 속성 삭제하기

```
session04.jsp
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <%@ page import="java.util.Enumeration"%>
3 <html>
4 <head>
5 <title>Session</title>
6 </head>
7 <body>
8 <h4>----- 세션을 삭제하기 전 -----</h4>
9 <%
10     String userId = (String) session.getAttribute("userID");
11     String userPwd = (String) session.getAttribute("userPW");
12     out.println("설정된 세션 이름 userID : " + userId + "<br>");
13     out.println("설정된 세션 값 userPW : " + userPwd + "<br>");
14
15     session.removeAttribute("userID");
16 %>
17 <h4>----- 세션을 삭제한 후 -----</h4>
18 <%
19     userId = (String) session.getAttribute("userID");
20     userPwd = (String) session.getAttribute("userPW");
21
22     out.println("설정된 세션 이름 userID : " + userId + "<br>");
23     out.println("설정된 세션 값 userPW : " + userPwd + "<br>");
24 %>
25 </body>
26 </html>
```

http://localhost:8090/JSPTe

----- 세션을 삭제하기 전 -----

설정된 세션 이름 userID : admin  
설정된 세션 값 userPW : 1234

----- 세션을 삭제한 후 -----

설정된 세션 이름 userID : null  
설정된 세션 값 userPW : 1234

# 단일 세션 삭제

## ▶ 세션에 저장된 세션 속성 삭제하기2

```
session05.jsp
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <%@ page import="java.util.Enumeration"%>
3 <html>
4 <head>
5 <title>Session</title>
6 </head>
7 <body>
8 <h4>----- 세션을 삭제하기 전 -----</h4>
9 <%
10     String name;
11     String value;
12
13     Enumeration<String> en = session.getAttributeNames();
14     int i = 0;
15
16     while (en.hasMoreElements()) {
17         i++;
18         name = en.nextElement();
19         value = session.getAttribute(name).toString();
20         out.println("설정된 세션 이름 [ " + i + " ] : " + name + "<br>");
21         out.println("설정된 세션 값 [ " + i + " ] : " + value + "<br>");
22     }
23
24     session.removeAttribute("userID");
25 %>
```

http://localhost:8090/JSPTest

----- 세션을 삭제하기 전 -----

설정된 세션 이름 [ 1 ] : userPW  
설정된 세션 값 [ 1 ] : 1234  
설정된 세션 이름 [ 2 ] : userID  
설정된 세션 값 [ 2 ] : admin

----- 세션을 삭제한 후 -----

설정된 세션 이름 [ 1 ] : userPW  
설정된 세션 값 [ 1 ] : 1234

# 단일 세션 삭제

---

```
26 <h4>----- 세션을 삭제한 후 -----</h4>
27 <%
28     en = session.getAttributeNames();
29
30     i = 0;
31     while (en.hasMoreElements()) {
32         i++;
33         name = en.nextElement().toString();
34         value = session.getAttribute(name).toString();
35         out.println("설정된 세션 이름 [ " + i + " ] : " + name + "<br>");
36         out.println("설정된 세션 값 [ " + i + " ] : " + value + "<br>");
37     }
38 %>
39 </body>
40 </html>
```



# 다중 세션 삭제

---

- ▶ invalidate( ) 메소드를 사용해 세션에 저장된 모든 세션 속성 이름을 삭제

```
void invalidate()
```

- ▶ invalidate( ) 메소드 사용 예

```
session.invalidate()
```



# 다중 세션 삭제하기

## ▶ 세션에 저장된 모든 세션 속성 삭제하기

```
session06.jsp
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <%@ page import="java.util.Enumeration"%>
3 <html>
4 <head>
5 <title>Session</title>
6 </head>
7 <body>
8   <h4>----- 세션을 삭제하기 전 -----</h4>
9   <%
10     String userId = (String) session.getAttribute("userID");
11     String userPwd = (String) session.getAttribute("userPW");
12
13     out.println("설정된 세션 이름 userID : " + userId + "<br>");
14     out.println("설정된 세션 값 userPW : " + userPwd + "<br>");
15
16     if (request.isRequestedSessionIdValid() == true) {
17       out.print("세션이 유효합니다.");
18     } else {
19       out.print("세션이 유효하지 않습니다.");
20     }
21
22     session.invalidate();
23   %>
```

http://localhost:8090/JSI

----- 세션을 삭제하기 전 -----

설정된 세션 이름 userID : admin  
설정된 세션 값 userPW : 1234  
세션이 유효합니다.

----- 세션을 삭제한 후 -----

세션이 유효하지 않습니다.



# 다중 세션 삭제하기

---

```
24 <h4>----- 세션을 삭제한 후 -----</h4>
25 <%
26     if (request.isRequestedSessionIdValid() == true) {
27         out.print("세션이 유효합니다.");
28     }else {
29         out.print("세션이 유효하지 않습니다.");
30     }
31     %>
32 </body>
33 </html>
```



# 세션 유효 시간 설정

---

## ▶ 세션 유효 시간

- ▶ 세션을 유지하기 위한 세션의 일정 시간
- ▶ 웹 브라우저에 마지막 접근한 시간부터 일정 시간 이내에 다시 웹 브라우저에 접근하지 않으면 자동으로 세션이 종료
- ▶ 세션 유효 시간을 설정하기 위해 session 내장 객체의 `setMaxInactiveInterval()` 메소드를 사용

```
void setMaxInactiveInterval(int interval)
```

- ▶ `setMaxInactiveInterval()` 메소드 사용 예

```
session.setMaxInactiveInterval(60 * 60);
```



# 세션 유효 시간 설정

## ▶ 세션 유효 시간을 가져와 출력하기

session07.jsp

```
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <html>
3 <head>
4 <title>Session</title>
5 </head>
6 <body>
7 <h4>----- 세션 유효 시간 변경 전 -----</h4>
8 <%
9     int time = session.getMaxInactiveInterval() / 60;
10    out.println("세션 유효 시간 : " + time + "분<br>");
11 %>
12 <h4>----- 세션 유효 시간 변경 후 -----</h4>
13 <%
14     session.setMaxInactiveInterval(60 * 60);
15     time = session.getMaxInactiveInterval() / 60;
16     out.println("세션 유효 시간 : " + time + "분<br>");
17 %>
18 </body>
19 </html>
```

http://localhost:8090/JSPTe

----- 세션 유효 시간 변경 전 -----  
세션 유효 시간 : 30분  
----- 세션 유효 시간 변경 후 -----  
세션 유효 시간 : 60분

# 세션 유효 시간 설정

## ▶ 세션 아이디와 웹 사이트에서 유지한 시간 출력하기

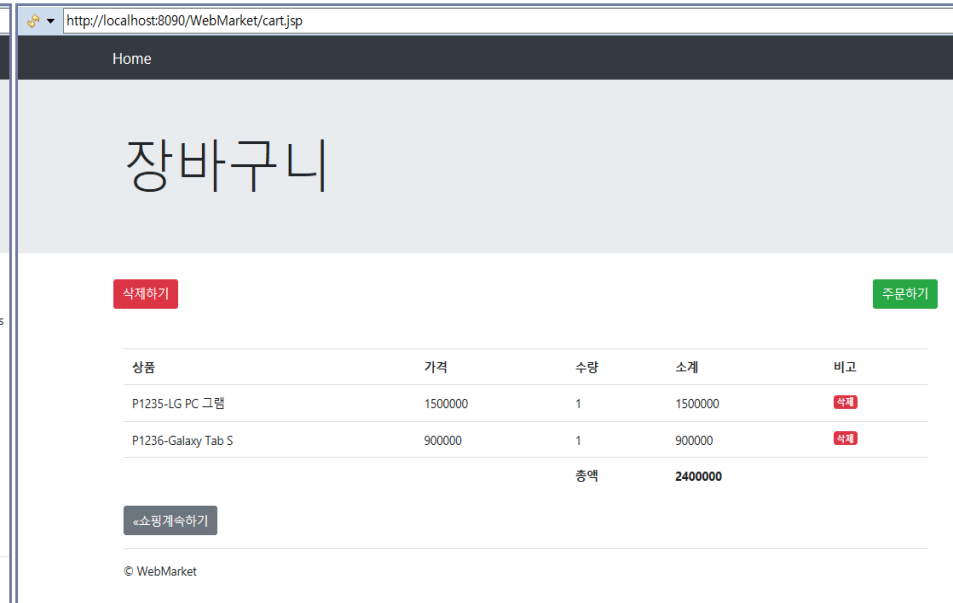
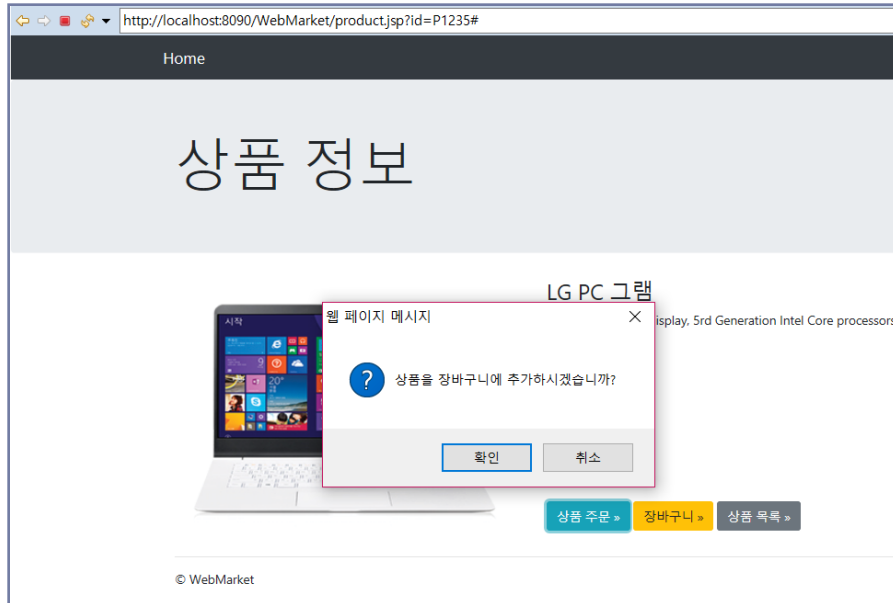
session08.jsp

```
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <html>
3 <head>
4 <title>Session</title>
5 </head>
6 <body>
7   <%
8     String sessinId = session.getId();
9     long lastTime = session.getLastAccessedTime();
10    long startTime = session.getCreationTime();
11    long usedTime = (lastTime - startTime) / 60000;
12
13    out.println("세션 아이디 : " + sessinId + "<br>");
14    out.println("요청 시작 시간 : " + startTime + "<br>");
15    out.println("요청 마지막 시간 : " + lastTime + "<br>");
16    out.println("웹 사이트에서 경과 시간 : " + usedTime + "<br>");
17  %>
18 </body>
19 </html>
```

http://localhost:8090/JSPTest/Chapter13/session08.jsp

세션 아이디 : A984A2CE311C41B6BF58F6B0C2F860DA  
요청 시작 시간 : 1550060422567  
요청 마지막 시간 : 1550062900563  
웹 사이트에서 경과 시간 : 41

# 장바구니 페이지 만들기



# 세션을 이용해 장바구니 페이지 만들기

## ▶ 상품 클래스에 멤버 변수 추가하기

Product.java

```
1 package dto;
2 import java.io.Serializable;
3 public class Product implements Serializable {
4     private static final long serialVersionUID
5     private String productId;
6     private String pname;
7     private Integer unitPrice;
8     private String description;
9     private String manufacturer;
10    private String category;
11    private long unitsInStock;
12    private String condition;
13    private String filename;
14    private int quantity;
15 }
```

# 장바구니 페이지 만들기

---

- ▶ 추가된 멤버 변수의 setter/setter() 메소드 작성하기
  - ▶ -Product.java

```
98 public int getQuantity() {  
99     return quantity;  
100 }  
101  
102 public void setQuantity(int quantity) {  
103     this.quantity = quantity;  
104 }  
105  
106 } //class Product
```

# 장바구니 페이지 만들기

## ▶ 상품 상세 정보 페이지 수정하기

```
product.jsp ✖
1 <%@ page contentType="text/html; charset=utf-8"%>
2 <%@ page import="dto.Product"%>
3 <%@ page import="dao.ProductRepository"%>
4 <%@ page errorPage="exceptionNoProductId.jsp" %>
5 <html>
6 <head>
7 <link rel="stylesheet" href="../resources/css/bootstrap.min.css" />
8 <title>상품 상세 정보</title>
9 <script type="text/javascript">
10     function addToCart(){
11         if(confirm("상품을 장바구니에 추가하시겠습니까?")){
12             document.addForm.submit();
13         }else{
14             document.addForm.reset();
15         }
16     }
17 </script>
18 </head>
```



# 상품 상세 정보 페이지 수정하기

## ▶ 상품 상세 정보 페이지 수정하기(product.jsp)

```
36<div class="col-md-6">
37    <h3><%=product.getPname()%></h3>
38    <p><%=product.getDescription()%>
39    <p><b>상품 코드 : </b><span class="badge badge-danger"> <%=product.getProductId()%><
40    <p><b>제조사</b> : <%=product.getManufacturer()%>
41    <p><b>분류</b> : <%=product.getCategory()%>
42    <p><b>재고 수</b> : <%=product.getUnitsInStock()%>
43    <h4><%=product.getUnitPrice()%>원</h4>
44    <p><form name="addForm" action="./addCart.jsp?id=<%=product.getProductId()%>"
45        method="post">
46        <a href="#" class="btn btn-info" onclick="addToCart()"> 상품 주문 &raquo;</a>
47        <a href="./cart.jsp" class="btn btn-warning"> 장바구니 &raquo;</a>
48        <a href="./products.jsp" class="btn btn-secondary"> 상품 목록 &raquo;</a>
49    </form>
50</div>
51</div>
```

# 장바구니 페이지 만들기

## ▶ 장바구니에 등록하는 페이지 작성하기:

```
addCart.jsp
1 <%@ page contentType="text/html; charset=UTF-8"%>
2 <%@ page import="java.util.ArrayList"%>
3 <%@ page import="dto.Product"%>
4 <%@ page import="dao.ProductRepository"%>
5
6 <%
7     String id = request.getParameter("id");
8     if(id==null || id.trim().equals("")){
9         response.sendRedirect("products.jsp");
10        return;
11    }
12
13    ProductRepository dao = ProductRepository.getInstance();
14    Product product = dao.getProductById(id);
15    if(product==null){
16        response.sendRedirect("exceptionNoProductId.jsp");
17    }
18 %>
```

# 장바구니 페이지 만들기

---

```
19 ArrayList<Product> goodsList = dao.getAllProducts();
20 Product goods = new Product();
21 for(int i=0; i<goodsList.size();i++){
22     goods = goodsList.get(i);
23     if(goods.getProductid().equals(id)){
24         break;
25     }
26 }
27
28 ArrayList<Product> list
29     = (ArrayList<Product>)session.getAttribute("cartlist");
30 if(list==null){
31     list = new ArrayList<Product>();
32     session.setAttribute("cartlist", list);
33 }
```



## 장바구니 페이지 만들기

```
35     int cnt=0;
36     Product goodsQnt = new Product();
37     for(int i=0; i<list.size();i++){
38         goodsQnt = list.get(i);
39         if(goodsQnt.getProductId().equals(id)){
40             cnt++;
41             int orderQuantity = goodsQnt.getQuantity() + 1;
42             goodsQnt.setQuantity(orderQuantity);
43         }
44     }
45
46     if(cnt==0){
47         goods.setQuantity(1);
48         list.add(goods);
49     }
50     response.sendRedirect("product.jsp?id=" + id);
51 %>
```

# 장바구니 페이지 만들기

## ▶ 장바구니 페이지 작성하기(cart.jsp)

```
cart.jsp ✕
1  <%@ page contentType="text/html; charset=UTF-8"%>
2  <%@ page import="java.util.ArrayList"%>
3  <%@ page import="dto.Product"%>
4  <%@ page import="dao.ProductRepository"%>
5  <html>
6  <head>
7  <link rel="stylesheet"
8      href="./resources/css/bootstrap.min.css" />
9  <%
10     String cartId = session.getId();
11 %>
12 <title>장바구니</title>
13 </head>
14 <body>
15     <jsp:include page="menu.jsp" />
16     <div class="jumbotron">
17         <div class="container">
18             <h1 class="display-3">장바구니</h1>
19         </div>
20     </div>
```

# 장바구니 페이지 만들기

```
21< <div class="container">
22    <div class="row">
23        <table width="100%">
24            <tr>
25                <td align="left"><a href="./deleteCart.jsp?cartId=<%=cartId%>"
26                    class="btn btn-danger">삭제하기</a></td>
27                <td align="right"><a href="#" class="btn btn-success">주문하기</a></td>
28            </tr>
29        </table>
30    </div>
31    <div style="padding-top: 50px">
32        <table class="table table-hover">
33            <tr>
34                <th>상품</th>
35                <th>가격</th>
36                <th>수량</th>
37                <th>소계</th>
38                <th>비고</th>
39            </tr>
40            <%
41                int sum=0, total;
42                ArrayList<Product> cartList
43                    = (ArrayList<Product>)session.getAttribute("cartlist");
44                if(cartList==null){
45                    cartList = new ArrayList<Product>();
46                }
```

# 장바구니 페이지 만들기

---

```
47         for(int i=0; i<cartList.size();i++){
48             Product product = cartList.get(i);
49             total = product.getUnitPrice()*product.getQuantity();
50             sum += total;
51         }
52     <tr>
53         <td><%=product.getProductId()%>-<%=product.getPname() %></td>
54         <td><%=product.getUnitPrice()%></td>
55         <td><%=product.getQuantity()%></td>
56         <td><%=total%></td>
57         <td><a href="./removeCart.jsp?id=<%=product.getProductId()%>"
58             class="badge badge-danger">삭제</a></td>
59     </tr>
60     <%} %>
61 </table>
```



# 장바구니 페이지 만들기

```
61<tr>
62    <th></th>
63    <th></th>
64    <th>총액</th>
65    <th><%=sum%></th>
66    <th></th>
67</tr>
68</table>
69    <a href="./products.jsp" class="btn btn-secondary">&laquo;쇼핑계속하기</a>
70</div>
71<hr>
72</div>
73<jsp:include page="footer.jsp" />
74</body>
75</html>
```



# 장바구니 페이지 만들기

- ▶ 장바구니에 등록된 개별 상품 삭제 페이지 작성하기:

```
removeCart.jsp ✖
1 <%@ page contentType="text/html; charset=UTF-8"%>
2 <%@ page import="java.util.ArrayList"%>
3 <%@ page import="dto.Product"%>
4 <%@ page import="dao.ProductRepository"%>
5
6 <%
7     String id = request.getParameter("id");
8     if(id==null || id.trim().equals("")){
9         response.sendRedirect("products.jsp");
10        return;
11    }
12
13    ProductRepository dao = ProductRepository.getInstance();
14    Product product = dao.getProductById(id);
15    if(product==null){
16        response.sendRedirect("exceptionNoProductId.jsp");
17    }
18 }
```

# 장바구니 페이지 만들기

---

```
19 ArrayList<Product> cartList
20     = (ArrayList<Product>)session.getAttribute("cartlist");
21
22 Product goodsQnt = new Product();
23 for(int i=0; i<cartList.size();i++){
24     goodsQnt = cartList.get(i);
25     if(goodsQnt.getProductId().equals(id)){
26         cartList.remove(goodsQnt);
27     }
28 }
29
30 response.sendRedirect("cart.jsp");
31 %>
```



# 장바구니 페이지 만들기

- ▶ 장바구니에 등록된 전체 상품 삭제 페이지 작성하기:

```
deleteCart.jsp ✕
1  <%@ page contentType="text/html; charset=UTF-8"%>
2
3  <%
4      String id = request.getParameter("cartId");
5      if(id==null || id.trim().equals("")){
6          response.sendRedirect("cart.jsp");
7          return;
8      }
9      session.invalidate();
10
11     response.sendRedirect("cart.jsp");
12 %>
```