```
Ajax 란 무엇인가?
 1
 2
    1. Ajax 방식
 3
      1)Asynchronous JavaScript and XML : 기존의 기술들을 모아서 비동기적으로 Page를 Rendering하는 기술
 4
      2)기존 방식
 5
         -Web Browser가 Web Server에 Request한다.
         -Web Server는 JSP/PHP/ASP등의 Server측 application을 사용해서 사용자의 request을 처리한 뒤, 처리
 6
         결과를 HTML로 생성해서 Web Browser에 Response한다.
 7
         -Web Browser는 Response로 받은 HTML을 분석한 뒤 그 내용을 화면에 다시 Rendering한다.
 8
 9
      3) Ajax 방식
         -http://map.google.com
10
11
         -기존에는 Drag를 구현하기 위해 ActiveX나 Web Broser Plug-in을 사용했었다.
12
         -http://www.naver.com
13
         -제시어 기능
14
         -몇가지 특징들
15
           a. 사용자에게 즉각적인 반응과 풍부한 UI 경험을 제공한다.
16
           b. ActiveX와 같이 특정한 Web Browser에서만 제공하는 기능에 의존하지 않는다.
17
           c. 전체적인 Page Loading보다 Page의 일부분만 Rendering하기 때문에 Server의 부하가 줄어든다.
18
           d. 전체가 새로고침(Refresh)되지 않는다.
19
20
         [noajax.jsp]
21
         <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
22
         <!DOCTYPE html>
23
         <html lang="en">
24
         <head>
25
           <meta charset="UTF-8">
26
           <title>No Ajax : Submit</title>
27
           <script>
28
              window.addEventListener("load", setup, false);
29
              function setup(){
                document.getElementById("btnCalc").addEventListener("click", Click, false);
30
31
32
              function Click(){
33
                if(document.getElementById("num1").value != "" &&
                     document.getElementById("num2").value != ""){
34
                   document.forms[0].action="noajax.jsp";
35
36
                   document.forms[0].method="post";
37
                   document.forms[0].submit();
38
                }
39
40
           </script>
41
         </head>
42
         <body>
43
44
              <input type="text" id="num1" name="num1" size="10" value="${param.num1}" /> X
              <input type="text" id="num2" name="num2" size="10" value="${param.num2}" /> =
45
              <input type="text" id="answer" size="10" value="${param.num1 * param.num2}"</pre>
46
              />  
47
              <input type="submit" value="Calculate" id="btnCalc" />
48
           </form>
49
         </body>
50
         </html>
51
52
         [usingajax.jsp]
53
54
           int num1 = Integer.parseInt(request.getParameter("num1"));
55
           int num2 = Integer.parseInt(request.getParameter("num2"));
56
           out.println(num1 * num2);
57
         %>
58
59
         [usingajax.html]
         <!DOCTYPE html>
60
61
         <html>
         <head>
62
           <meta charset="UTF-8">
63
64
           <title>Insert title here</title>
           <script>
65
```

```
66
               window.addEventListener("load", setup, false);
 67
               function setup(){
                 document.getElementById("btnCalc").addEventListener("click", Click, false);
 68
 69
 70
               function Click(){
 71
                 var xmlObj = new XMLHttpRequest();
                 var param = "num1=" + document.getElementById("num1").value +
 72
 73
                                  "&num2=" + document.getElementById("num2").value;
 74
                 xmlObj.onreadystatechange = function(){
 75
                    if(xmlObj.readyState == 4 && xmlObj.status == 200){
 76
                      document.getElementById("answer").value = xmlObj.responseText;
 77
 78
                 }
 79
                 xmlObj.open("get", "usingajax.jsp?" + param, true);
 80
                 xmlObj.send(null);
 81
               }
 82
             </script>
 83
          </head>
 84
          <body>
             <input type="text" id="num1" size="10" /> X
 85
             <input type="text" id="num2" size="10" /> =
 86
 87
            <input type="text" id="answer" size="10" />&nbsp;&nbsp;
 88
             <input type="button" value="Calculate" id="btnCalc" />
 89
          </body>
 90
          </html>
 91
 92
 93
     2. Ajax 의 주요 구성 요소
 94
       1)XMLHttpRequest
 95
          -Web Server와 통신을 담당한다.
 96
          -사용자의 요청을 Web Server(WAS)에 전송하고, Web Server(WAS)로부터 받은 결과를 Web Broser에 전달한다.
 97
 98
       2)DOM
 99
          -문서의 구조를 나타낸다.
100
          -Form 등의 정보나 화면 구성을 조작할 때 사용된다.
101
102
       3)CSS
103
          -글자색, 배경색, 위치, 투명도 등 UI와 관련된 부분을 담당한다.
104
105
       4)JavaScript
106
          -사용자가 mouse를 drag하거나 button을 click하면, XHR객체를 사용해서 Web Server에 요청을 전송한다.
107
          -또한 XHR 객체로부터 응답이 오면 DOM, CSS등을 사용해서 화면을 조작한다.
108
109
       5)순서
110
          a. 사용자가 event를 발생시키면 JavaScript는 DOM을 사용해서 필요한 정보를 구한 뒤, XHR객체를 통해서 Web
          Server(WAS)에 요청을 전달한다.
          b. Web Server(WAS)는 XHR로부터 요청을 알맞게 처리한 뒤, 그 결과를 XML이나 단순 Text로 생성해서 XHR에
111
          전송한다.
112
          c. Server로부터 응답이 도착하면 XHR객체는 JavaScript에 도착 사실을 알리게 되고, JavaScript는 응답 data와
          DOM을 조작해서 사용자 화면에 반영한다.
113
114
       6)Ajax 가 기존의 방식과 다른 차이점
115
          a. Web Browser 가 아닌 XHR객체가 Web Server와 통신한다.
116
          b. Web Server의 응답 결과가 HTML이 아닌 XML 또는 단순 Text이다.
          c. Page 이동없이 결과가 화면에 반영된다.
117
118
119
120
          <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
121
          < @ page import="java.util.Calendar" %>
122
            response.setHeader("cache-control", "no-cache"); //HTTP 1.1 response.setHeader("cache-control", "no-store"); //FF
123
124
125
            response.setDateHeader("expires", 0);
            response.setHeader("pragma", "no-cache"); //HTTP1.0
126
127
            Calendar now = Calendar.getInstance();
128
129
            int hour = now.get(Calendar.HOUR_OF_DAY);
```

```
130
             int minute = now.get(Calendar.MINUTE);
131
             int second = now.get(Calendar.SECOND);
             out.println("" + hour + " : " + minute + " : " + second);
132
          %>
133
134
135
          [test.html]
136
          <!DOCTYPE html>
137
          <html>
138
          <head>
             <meta charset="UTF-8"/>
139
140
             <title>Insert title here</title>
141
             <script>
142
                window.addEventListener("load", setup, false);
143
                function setup(){
144
                  document.getElementById('mytext').addEventListener('change', startAjax, false);
145
146
                function startAjax(){
147
                  var xmlObj;
148
                  xmlObj = new XMLHttpRequest();
149
                  xmlObj.onreadystatechange = function(){
150
                     if(xmlObj.readyState == 4){
                       document.getElementById("mytime").value = xmlObj.responseText;
151
                       //alert(xmlObj.responseText);
152
153
                     }
154
                  }
155
                  xmlObj.open("get", "date.jsp", true);
156
                  xmlObj.send(null);
157
158
             </script>
159
          </head>
160
          <body>
             <form>
161
162
                Name : <input type="text" id="mytext" /><br />
163
                <input type="text" id="mytime" size="30"/>
             </form>
164
165
          </body>
166
          </html>
```