

# 자바스크립트를 이용한 쿠키의 컨트롤

**목 차**

1. 스크립트언어와 자바스크립트의 개념
2. 자바스크립트의 규칙과 기본 문법
3. 자바스크립트 실전활용
4. 자바스크립트 실전활용 2 (이번호)
5. 향상된 자바스크립트 기법
6. 쿠키와 자바스크립트

이번 호에는 지난 호에 이어서 자바스크립트를 이용해 실제적으로 홈페이지에 활용할수 있는 기법들을 중심으로 살펴 보면서 자바스크립트의 다양한 기능들에 대해 살펴 보고자 한다. 특히 이번 시간에는 간단하지만 활용도가 높은 쿠키에 대해 알아보하고자 한다.

황희정/넥스텔 멀티미디어연구소

## 쿠키(Cookie)란 ..?

쿠키는 RFC2109 에 기술된 표준안으로서 HTTP State Management Mechanism에 관련된 표준 규약의 일부로 포함되어 있다. 쿠키가 나오게된 배경은 http 프로토콜 자체가 비연결형 프로토콜이기 때문에 한번 접속된 사용자에게 정보를 유지하기가 무척 어렵다는 특성 때문이다. 즉 A라는 홈페이지가 있고 B라는 사용자가 있다고 하자. B라는 사용자가 A라는 홈페이지를 접속 했을 때 A홈페이지는 B라고 하는 사용자에게 대해 알 수 있는 것이 없다.

물론 접속한 IP 주소, 클라이언트 프로그램의 종류 등등의 몇몇 정보는 알 수 있지만 그 역시 CGI 등을 이용해 처리해 주어야만 알 수가 있다. 또한 온라인 쇼핑몰 등에서는 사용자가 신청한 물품들은 일일이 CGI 프로그램을 통해 보관해야 하는 불편함이 있었다.

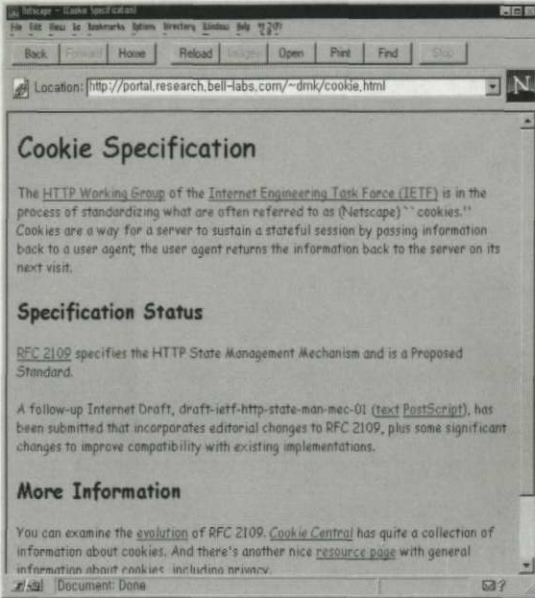
따라서 서버에는 불필요한 정보들이 쌓이게 되

고, 관리에 어려움도 많이 있었다. 쿠키의 기본 개념은 간단한 정보들을 서버가 보관하는 것이 아니라, 클라이언트 브라우저를 통해 보관을 했다가 필요할 때 꺼내볼 수 있도록 하고자 하는 방식이다.

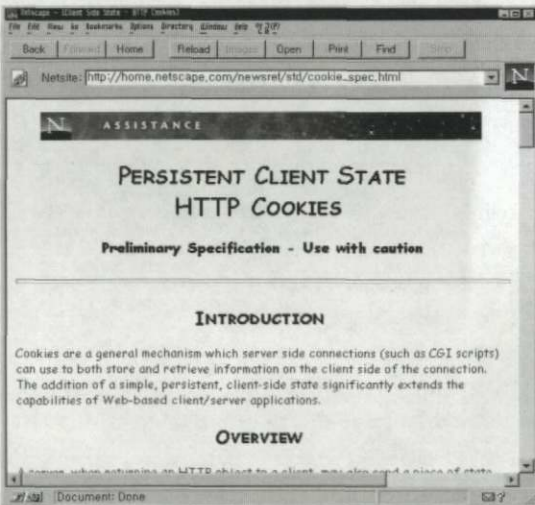
이렇게 함으로서 서버측에서는 사용자의 정보에 대해 신경 쓸 일이 없게 되는 것이다. 단지 필요한 때 설정하고 필요할때 꺼내쓰면 된다.

이와 같은 쿠키의 속성은 특히 인트라넷 혹은 CUG(Close User Group)을 구현할 때 무척 유용하게 쓰인다. 즉 사용자의 로그인 정보, 또는 특정 페이지를 다녀왔는지에 대한 정보등을 활용할 수 있기 때문이다.

현재 쿠키는 CGI를 제작할 수 있는 거의모든 언어 즉, perl, C, tcl/tk, 쉘스크립트 등에서 모두 구현이 가능하지만 자바스크립트와 함께 가장 많이 이용된다. 현재 쿠키에 대한 공식 사이트는 [http://home.netscape.com/newsref/std/cookie\\_spec.html](http://home.netscape.com/newsref/std/cookie_spec.html) 이다.



(그림 1) 쿠키 규약



(그림 2) Official Cookie Spec Site

## 쿠키의 기본 문법

다음의 내용은 표준 CGI 스크립트로 구현 가능한 HTTP 헤더들이다.

```
Set-Cookie:
NAME=VALUE; expires=DATE;
path=PATH;
domain=DOMAIN_NAME; secure
```

### ▶ NAME=VALUE

name에는 설정하고 싶은 쿠키의 구별이름을 기입한다. 예를들어 이름과 아이디 두가지를 저장하고 싶으면 Set-Cookie: cname=회정, cid=

love%20me ...와 같이 사용 한다. 이때 각각의 구분 이름은 ';'를 이용해 분리 표기한다. 주의할 것은 value값의 내용중 공백문자를 포함하는 문자열은 공백 부분에 공백 ASCII 값인 20을 %와 함께 표시해 준다. 즉 위의 예에서는 cid 값은 "love me"가 되는 것이다. name 은 쿠키 지정시 반드시 있어야 하는 부분이다.

### ▶ expires=DATE

expires 는 말 그대로 쿠키를 파키할 날짜를 지정 하는 것이다. 만일 지정된 날짜가 현재 보다 과거라면 쿠키는 자동으로 지워진다. 날짜 표기는 Wdy, DD-Mon-YYYY HH:MM:SS GMT와 같이 RFC 822, RFC 850, RFC 1036, RFC 1123 에 지정된 규칙을 준수한다. expires는 쿠키 지정시 선택 지정 부분이다.

### ▶ domain=DOMAIN\_NAME

domain 은 지정된 쿠키를 특정 도메인으로 구별 지어 찾고자 할 때 이용되는 속성으로 일반적인 도메인 규칙에 따라 지정하면 된다. 따라서 다음의 설정은 모두 동일한 도메인으로 인식되게 된다. "www.uriel.net", "hawk.uriel.net", "backdu.uriel.net" domain속성은 쿠키 지정시 기본값으로 해당 서버의 도메인이 기록되게 된다.

### ▶ path=PATH

path는 URL중 쿠키가 유효한 범위를 지정하는 부분이다. 예를 들어 '/' 로 지정되면 전체 홈페이지에서 유용한 정보가 되고, '/intro/sample'로 지정하면 /intro/sample 디렉토리 밑에 있는 페이지에서만 유용한 정보가 된다. 물론 '/'는 '/intro/sample'까지도 포함한다.

### ▶ secure

secure로 지정된 쿠키는 HTTPS 즉 SSL에 의해 보안된 채널을 가진 서버에서만 유용한 정보를 읽을 수 있는 정보가 된다. 사실 쿠키 자체는 그다지 보안성이 있지는 않다. 예를 들어 card\_no=

001-12-121-1000라고 지정될 경우 다른 사이트에서 들어오는 모든 사용자에게 대해 card\_no 정보를 읽는다면 개인 사용자의 정보가 다른 곳으로 누출될 수 있기 때문이다. 따라서 쿠키 자체를 설정할 때도 해당 서버에서만 인식 할 수 있는 암호문 형식으로 변환된 값을 사용하거나, 규칙을 찾기 힘든 구성으로 쿠키 값을 설정하는 것이 일반적이다.

## 쿠키 사용 예제

```
Set-Cookie: CUSTOMER=WILE_E_COYOTE;
path=/; expires=Wednesday, 09-Nov-99 23:12:40 GMT
```

▶ 위와 같이 쿠키를 설정 했을 때 클라이언트가 홈페이지의 루트에 접속했을 때 다음과 같은 정보를 응답한다.

```
Cookie: CUSTOMER=WILE_E_COYOTE
```

동일 클라이언트에 대해 다시 다음과 같은 정보가 설정 된다고 하자.

```
Set-Cookie: PART_NUMBER=
ROCKET_LAUNCHER_0001; path=/  

```

▶ 이때 클라이언트가 홈페이지의 루트에 접속했을 때 1의 정보와 추가된 정보가 함께 설정 되게 된다.

```
Cookie: CUSTOMER=WILE_E_COYOTE;
PART_NUMBER=ROCKET_LAUNCHER_0001
```

위에서 살펴본 것 처럼 쿠키 정보는 설정 할 때마다 새롭게 지정되는 것이 아니라 기존의 정보를 보관하고 새로운 정보를 추가로 보관하게 된다. 따라서 온라인 쇼핑몰에서 장바구니 시스템 등을 운영할 때 새로 구입하고자 하는 물품에 대한 정보들이 하나씩 추가되게 되는 것이다.

## 쿠키 - 자바스크립트

자 이제 다시 우리의 주제인 자바스크립트를 살펴보자. 자바스크립트는 여러 가지 용도로도 활용할 수 있지만 쿠키를 처리하는데 있어서도 탁월한 기능을 제공하고 있다. 전에 설명했듯이 자바스크립트는 내장된 객체들을 이용하는데 쿠키를 사용할 수 있는 객체는 document이다. 즉 document객체

의 속성중 하나를 이용하면 될 것이다. 그렇다면 그 속성은?

혹 기억하고 있는 독자분들도 있겠지만 아마 대부분 기억하지 못할 것이다. 살펴보면 document의 속성중에는 쿠키라는 것이 있다.

### ▶ document.cookie 예

```
document.cookie = "cname=hwanghj";
document.cookie = "cid = 0001";
```

위의 예처럼 자바스크립트에서 쿠키를 설정하는 것은 비교적 간단하다.

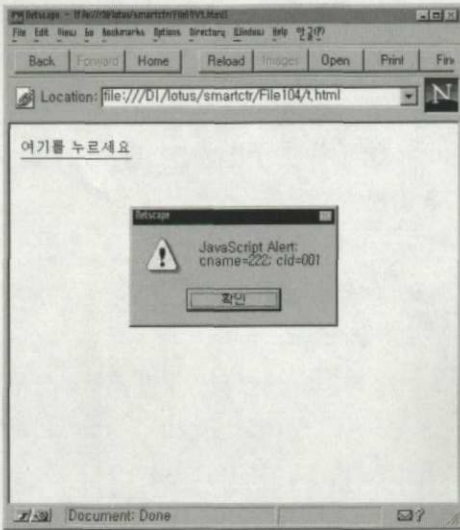
하지만 사용자에게 저장되어 있는 쿠키를 꺼내는 것은 생각처럼 쉽지 않다. 예를 들어 위와 같이 cname과 cid 두 개의 쿠키가 설정되어 있다고 할 때 사용자가 원하는 값을 어떻게 호출할 것인지 생각해 보자. 이해를 돕기 위해 다음의 간단한 예를 통해 문제점을 알아보도록 한다.

```
<html>
<head>
<script language=javascript>
function test (str) {

str.cookie = "cname=222";
str.cookie = "cid=001";
alert(str.cookie);
}
</script>
</head>
<body bgcolor=#ffffff>
<a href="" onClick="test(window.document):return false">
여기를 누르세요</a>
</body>
</html>
```

아주 간단한 소스인데 "여기를 누르세요"라는 링크를 누르면 2개의 쿠키값을 설정하고, 다시 alert을 통해 현재 설정된 쿠키값들을 알아보는 내용이다. 직접 실험해 보면 알겠지만 결과는 다음과 같다.

이처럼 설정할 때는 각각 설정하기 때문에 문제가 없지만 호출할 경우에는 전체 설정된 값이 한꺼



(그림 3) 실행 결과

변에 읽혀지기 때문에 필요한 값을 이름 가지고 내용을 분리해 내야 하는 부분이 필요한 것이다. 바로 이 부분이 우리가 이번에 살펴보는 중심적인 내용이라 할 수 있겠다.

이제 기본적인 원리와 방법들에 대해서는 어느정도 이해가 되었으리라 생각하고 현재 널리 사용되고 있는 자바스크립트-쿠키 라이브러리를 살펴보면서 내용을 분석해 보도록 한다.

전체 라이브러리는 다음과 같은 함수로 구성되어 있다.

1. getCookieVal (offsetd) : 설정된 쿠키값중 원하는 항목 까지의 오프셋 값을 가지고 해당 항목의 값만을 추출하는 함수이다. GetCookie()에 의해 호출 된다.
- 2.. FixCookieDate (date) : 쿠키의 날짜를 설정 하는 함수이다.
3. GetCookie (name) : 원하는 항목의 이름으로 쿠키 값을 얻고자 할 때 이용하는 메인 함수이다.
4. SetCookie (name,value,expires,path,domain,secure) : 원하는 항목의 이름과 기타 속성을 지정해 쿠키를 설정하는 함수이다.
5. DeleteCookie (name,path,domain) : 지정된 항목의 이름으로 쿠키의 날짜를 1970년으로 지정해 쿠키를 지워 버리는 함수이다.

```
<html>
<head>
```

```
<title>Cookie Functions</title>
</head>
```

```
<body>
<script language="javascript">
```

```
function getCookieVal (offset) {
    var endstr = document.cookie.
        indexOf (";", offset);
    if (endstr == -1)
        endstr = document.cookie.length;
    return unescape(document.cookie.substring
    (offset, endstr));
}
```

```
function FixCookieDate (date) {
    var base = new Date(0);
    var skew = base.getTime(); //
    dawn of (Unix) time - should be 0
    if (skew > 0) // Except on the
    Mac - ahead of its time
        date.setTime (date.getTime() - skew);
}
```

```
function GetCookie (name) {
    var arg = name + "=";
    var alen = arg.length;
    var clen = document.cookie.length;
    var i = 0;
    while (i < clen) {
        var j = i + alen;
        if (document.cookie.substring(i, j)
        == arg)
            return getCookieVal (j);
        i = document.cookie.
        indexOf(" ", i) + 1;
        if (i == 0) break;
    }
    return null;
}
```

```

}

function SetCookie
(name,value,expires,path,domain,secure) {
  document.cookie =
    name + "=" + escape (value) +
      ((expires) ? "; expires=" +
expires.toGMTString() : "") +
      ((path) ? "; path=" + path : "") +
      ((domain) ? "; domain="
+ domain : "") +
      ((secure) ? "; secure" : "");
}

function DeleteCookie (name,path,domain)
{
  if (GetCookie(name)) {
    document.cookie = name + "=" +
      ((path) ? "; path=" + path : "") +
      ((domain) ? ";
domain=" + domain : "") +
      "; expires=Thu, 01-Jan-
70 00:00:01 GMT";
  }
}
</script>
</head>

```

▶ 위에서 살펴본 Cookie - JavaScript 라이브러리에 주로 사용된 함수들은 대부분 String 객체 관련 함수들이다.

그러면 String 과 관련된 대표적인 함수를 살펴 보도록 하자.

- substring : 전체 스트링에서 특정 부분 만을 분리해 내고자 할 때 사용 된다.

```

(예) var str = "This is string";
var str2 = str.substring(9,14);
alert(str2);

```

결과는 str2 에 "string" 이라는 문자열이 들어가게 된다.

- indexOf : 특정 구분자 까지의 문자열을 이용하고자 할 때 사용된다.

```

(예) var str = "This is: String"
var str2 = str.indexOf(":",5);
alert(str2);

```

결과는 str2 에 "is" 라는 문자열이 들어가게 된다.

- length : 스트링의 길이를 나타낸다.

```

(예) var str = "This is String";
var slen = str.length;
alert(slen);

```

결과는 slen 에 str 의 길이인 14 가 들어가게 된다.

- 스트링 조합

자바스크립트에서 스트링은 자유자재로 조합될 수 있다. 다음은 활용 예이다.

```

1. var str = "This";
var str2 = "is";
var str3 = str + str2;

```

결과는 str3 에 "This is" 라는 문자열이 들어가게 된다.

```

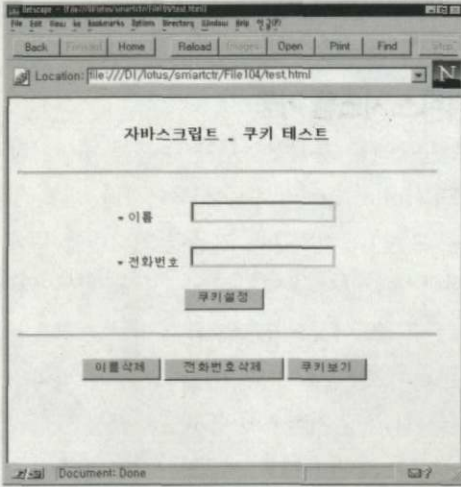
2. var str = "This";
var str2 = "string";
var str3 = str + " is " + str2;

```

결과는 str3 에 "This is string" 이라는 문자열이 들어가게 된다.

이것으로 자바스크립트를 이용한 쿠키 라이브러리에 대해 어느 정도 이해가 되었으리라 생각한다. 그러면 이제 실제적으로 위에서 살펴본 라이브러리를 이용해 쿠키를 활용 하는 예를 소개한다.

다음 예는 2개의 쿠키를 설정 하고 각각 해제 하는 기능을 가진 버튼을 가진 웹페이지에서 쿠키의 동작을 시험해 보는 예제 이다.



(그림 4) 기본화면

이것으로 미흡하지만 2회에 걸친 자바 스크립트 실전 활용을 마칠까 한다. 다음 시간에는 보다 향상된 자바스크립트 기법으로 자바스크립트를 이용해 마우스가 특정 이미지로 이동하면 다른 이미지로 대치시켜 마치 특수한 버튼 효과를 내는 기능을 구현해 보도록 한다. **DC**

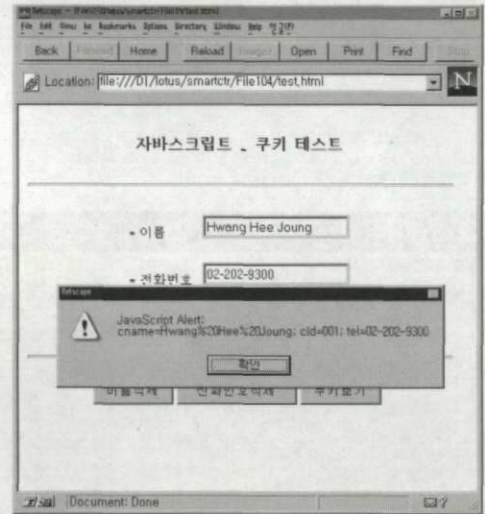
```

.....
.....
..... > 자바스크립트 - 쿠키 라이브러리(앞것과 동일)

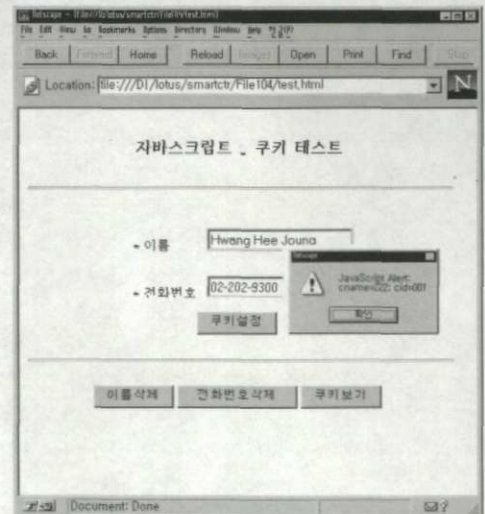
// 새롭게 추가된 함수로 입력된 쿠키값을 설정하기 위한 함수 이다.
function my_submit(obj) {
    SetCookie(obj.cform.cname.name,obj.cform.cname.value);
    SetCookie(obj.cform.tel.name,obj.cform.tel.value);
    alert(obj.cookie);
}

</script>
</head>
<body bgcolor=#ffffff>
<BR>
<center>
<H3> 자바스크립트 - 쿠키 테스트</h3>
<HR>
<table>
<form name=cform>
<tr><td>
* 이름
<td><input type=text name=cname><P>
<tr><td>
* 전화번호
<td>
<input type=text name=tel><P>
</table>
<input type=button value=쿠키설정 onClick=my_submit(window.document)>
</form>
<HR>
<form name=cform2>
<input type=button value=이름삭제 onClick="DeleteCookie('cname')">
<input type=button value=전화번호삭제 onClick="DeleteCookie('tel')">
<input type=button value=쿠키보기 onClick=alert(window.document.cookie)>
</form>
</center>
</body>
</html>

```



(그림 5) 쿠키 설정화면(1)



(그림 6) 쿠키 설정화면(2)