



Chapter 08 쿠키와 세션

목차

01 쿠키

02 세션

학습목표

- 쿠키와 세션의 동작 원리를 이해할 수 있다.
- 쿠키를 생성해서 사용하고 삭제할 수 있다.
- 세션을 시작하고 등록할 수 있다.
- 등록된 세션을 사용하고 삭제할 수 있다.

- 쿠키(cookie)는 사용자가 웹 페이지에 접속할 때 웹 서버가 사용자의 컴퓨터에 저장하는 작은 양의 데이터.
- 웹 페이지에서는 쿠키를 확인하여 로그인 상태나 장바구니에 상품이 담겨 있는지를 확인.
- 웹 서버는 쿠키를 사용자의 컴퓨터에 저장한 뒤 쿠키가 필요할 때 사용자의 컴퓨터에 요청하고 사용자의 컴퓨터는 저장된 쿠키를 웹 서버에 전송.



그림 8-1 쿠키의 개념

구분	웹 서버	사용자의 컴퓨터
데이터의 위치		○
쿠키의 위치		○
데이터 가공		○

그림 8-2 쿠키와 데이터(가공)의 위치

- 쿠키는 사용자의 컴퓨터에 텍스트 파일 형태로 저장되며 주로 다음과 같은 목적으로 사용.
 - ✓ 특정 웹 사이트를 재방문하거나 웹 사이트 내 다른 페이지 이동 시 다시 로그인할 필요 없음.
 - ✓ 사용자의 컴퓨터에서 아이디나 비밀번호 기억.
 - ✓ 사용자의 웹 페이지 이용 패턴 분석.

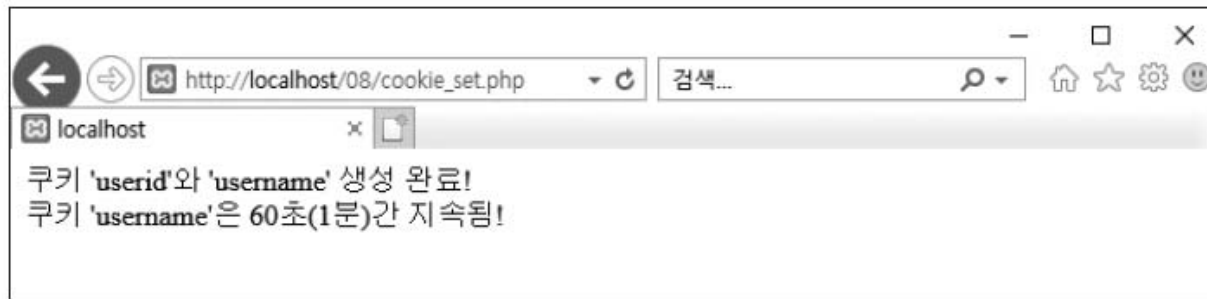
[예제 8-1] 쿠키 생성하기 08/cookie_set.php

```
<?php
    $a = setcookie("userid", "rubato");           ①
    $b = setcookie("username","김채린", time()+60); ②

    if ($a and $b)                                ③
    {
        echo "쿠키 'userid'와 'username' 생성 완료!<br>";
        echo "쿠키 'username'은 60초(1분)간 지속됨!"; ④
    }
?>
```

[예제 8-1] 쿠키 생성하기 08/cookie_set.php

실행 결과



- ① setcookie() 함수를 이용하여 userid라는 쿠키에 문자열 'rubato'를 저장.
 - Setcookie() 함수의 실행이 성공하면 true를 반환하고 실패하면 false를 반환.
 - \$a를 확인함으로써 쿠키가 제대로 생성되었는지 알 수 있음.
- ② username 쿠키에 '김채린' 저장. 60초(1분)간 유효. 쿠키가 제대로 생성되면 \$b는 true 값을 가짐.
- ③ if문 조건식인 \$a and \$b가 참이면 ④를 실행.
- ④ 특별한 문제가 없는 한 쿠키가 생성되고 echo문의 내용이 출력.

[예제 8-1] 쿠키 생성하기 08/cookie_set.php

setcookie() 함수	
형식	bool setcookie(string name, string value, int expire, string path)
인자	name: 쿠키 이름이며 대문자와 소문자를 구분한다.
	value: 쿠키 값이며 사용자의 컴퓨터에 저장된다.
	expire: 쿠키 유효 시간이며 초 단위로 표시된다. 예: time()+60*60*24*30→30일 동안 쿠키 값 유지
	path: 쿠키를 이용할 수 있는 경로를 의미한다. <ul style="list-style-type: none"> • 기본 값: 현재 디렉터리 • /. 전체 도메인에서 쿠키 사용 가능 • /home/: [home] 디렉터리와 하위 디렉터리에서 쿠키 사용 가능
반환 값	성공: true, 실패: false

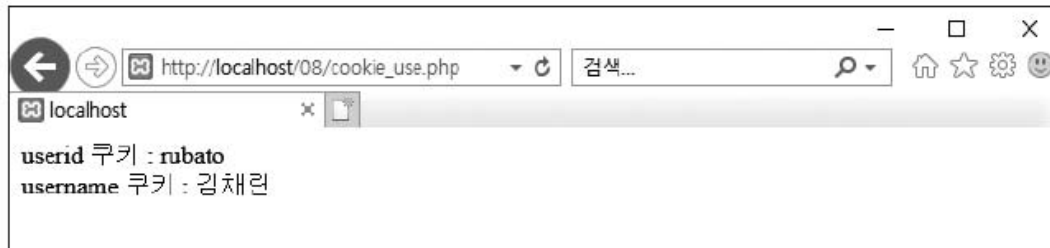
time() 함수	
형식	int time()
반환 값	성공: 현재 시간

[예제 8-2] 쿠키 사용하기 08/cookie_use.php

```
<?php
    if (isset($_COOKIE["userid"]) && isset($_COOKIE["username"])) ①
    {
        $userid = $_COOKIE["userid"]; ②
        $username = $_COOKIE["username"]; ③
        echo "userid 쿠키 : " . $userid."<br>"; ④
        echo "username 쿠키 : " . $username."<br>";
    }
    else
    {
        echo "쿠키가 생성되지 않았다!";
    }
?>
```

[예제 8-2] 쿠키 사용하기 08/cookie_use.php

실행 결과



① `$_COOKIE["userid"]`는 [예제 8-1]의 ①에서 `setcookie()` 함수로 생성한 `userid` 쿠키를, `$_COOKIE["username"]`은 [예제 8-1]의 ②에서 생성한 `username` 쿠키를 의미.

- `isset()` 함수는 변수 값이 설정되어 있는지 판단. 설정되어 있으면 참, 설정되어 있지 않으면 거짓 반환.
- if문 조건식은 `userid`, `username` 쿠키가 설정되어 있으면 ②~④를 실행.

② `userid` 쿠키의 값인 `$_COOKIE["userid"]`를 `$userid`에 저장.

③ `username` 쿠키의 값인 `$_COOKIE["username"]`을 `$username`에 저장.

④ `$userid`와 `$username`의 값을 출력.

[예제 8-3] 쿠키 삭제하기 08/cookie_delete.php

```
<?php
    setcookie("userid", "", time() - 3600);           ①
    setcookie("username", "", time() - 3600);         ②
?>
<html> <head>
<meta charset="utf-8">
</head>
<body>
<?php
    {
        echo "userid와 username 쿠키가 삭제되었다!";
    }
?>
</body> </html>
```

[예제 8-3] 쿠키 삭제하기 08/cookie_delete.php

실행 결과



- ① setcookie() 함수를 이용하여 userid 쿠키에 null 값인 " "를 설정하면 userid 쿠키를 삭제할 수 있음. 이때 쿠키 유효 시간을 현재 시간보다 1시간 짧게 설정. 문법상으로는 쿠키 유효 시간을 과거 시간으로 설정하는 것.
- ② 같은 방식으로 username 쿠키를 삭제.

- 사용자의 컴퓨터에 정보가 저장되는 쿠키와 달리 세션(session)은 보안을 이유로 사용자의 컴퓨터와 웹 서버에 모두 정보를 저장.
- 웹 사이트를 방문하는 사용자의 컴퓨터에는 세션 ID(session ID, SID) 정보를 저장하고, 웹 서버에는 사용자의 컴퓨터 세션 ID에 대응되는 세션 정보를 저장.
- 사용자의 컴퓨터 세션 ID가 유출되더라도 별다른 정보가 없고 주요 정보가 웹 서버에 저장되어 있기 때문에 쿠키보다 보안성이 강함.

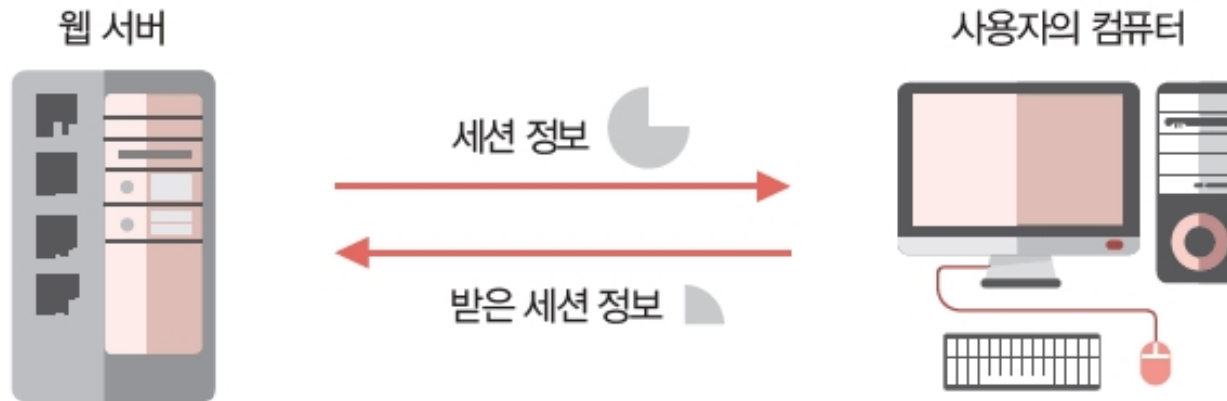


그림 8-3 세션의 개념

구분	웹 서버	사용자의 컴퓨터	비고
데이터의 위치	○		
세션 정보의 위치	○	○	서로 대응되는 정보(같지 않음)
데이터 가공	○		

그림 8-4 세션과 데이터(가공)의 위치

[예제 8-4] 세션 시작하고 등록하기

08/session_set.php

```
<html> <head>
<meta charset="utf-8">
</head>
<body>
<?php
    session_start();
    echo "세션 시작!!!<br>";
    $_SESSION['userid'] = "ocella";
    $_SESSION['username'] = "박영준";
    echo '세션 등록 완료!!!<br>';
    echo $_SESSION['userid']."<br>";
    echo $_SESSION['username']."<br>";
?>
</body> </html>
```

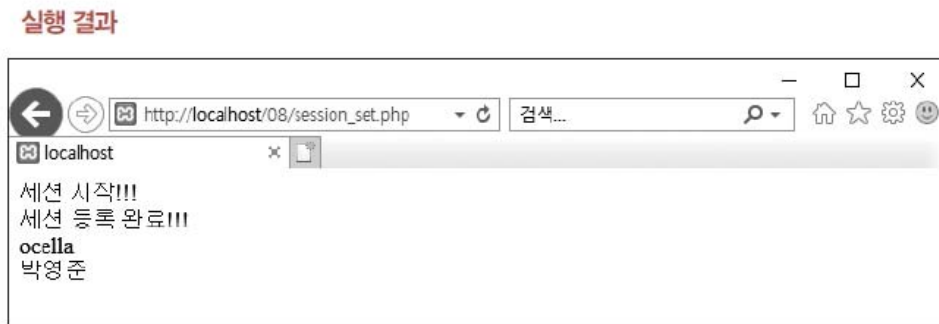
①

②

③

[예제 8-4] 세션 시작하고 등록하기

08/session_set.php



① 세션 초기화

- 세션을 사용하려면 반드시 `session_start()`로 초기화해야 함.

② 세션 등록

- 세션 변수 `userid`와 `username` 등록.
- 배열 형태의 전역 변수인 `$_SESSION[]`의 인덱스에 세션 변수의 이름을 넣어 등록.
- `userid`를 등록하기 위해 `$_SESSION['userid']`에 `ocella` 입력, `username`을 등록하기 위해 `$_SESSION['username']`에 `'박영준'` 입력.

③ 세션 변수 출력

- 등록한 2개의 세션 변수를 출력.

[예제 8-4] 세션 시작하고 등록하기

[08/session_set.php](#)

session_start() 함수	
형식	bool session_start()
반환 값	항상 true

[예제 8-5] 세션 사용하기

08/session_use.php

```
<?php
    session_start();                                ①
    $userid = $_SESSION["userid"];                  ②
    $username = $_SESSION["username"];
?>
<html> <head>
<meta charset="utf-8">
</head>
<body>
<h3>등록된 세션의 사용</h3>
<ul>
    <li>등록된 세션(userid) : <?= $userid?></li>          ③
    <li>등록된 세션(username) : <?= $username?></li>
</ul>
</table>
</body> </html>
```

[예제 8-5] 세션 사용하기

08/session_use.php

실행 결과



① session_start()로 세션을 시작.

② 등록된 세션인 `userid($_SESSION["userid"])`와 `username($_SESSION["username"])`을 각각 `$userid`와 `$username`에 저장.

③ `` 태그를 이용하여 변수 `$userid`와 `$username`을 리스트 형태로 출력.

[예제 8-6] 세션 삭제하기

08/session_delete.php

```
<?php
    session_start();                                ①
    unset($_SESSION["username"]);                  ②
?>
<html>
<head>
<meta charset="utf-8">
<link href="../css/style.css" rel="stylesheet">
</head>
<body>
<ul>
    <li><?= "userid 세션 : ".$_SESSION["userid"]." <br>" ?> </li>          ③
    <li><?= "username 세션 : ".$_SESSION["username"]." <br>" ?> </li>      ④
    <li><?= "username 세션이 삭제되었습니다!" ?> </li>
</ul>
</body> </html>
```

[예제 8-6] 세션 삭제하기

08/session_delete.php

실행 결과



- ① session_start()로 세션 시작.
- ② unset() 함수를 사용하여 username 세션 삭제.
- ③ 실행 결과의 첫 번째 줄에 userid 세션 값 출력.
- ④ username 세션 값 출력. 하지만 ②에서 username 세션이 삭제되었으므로 \$_SESSION의 username 인덱스가 정의되지 않았다는 오류 메시지가 나타남.

[예제 8-6] 세션 삭제하기

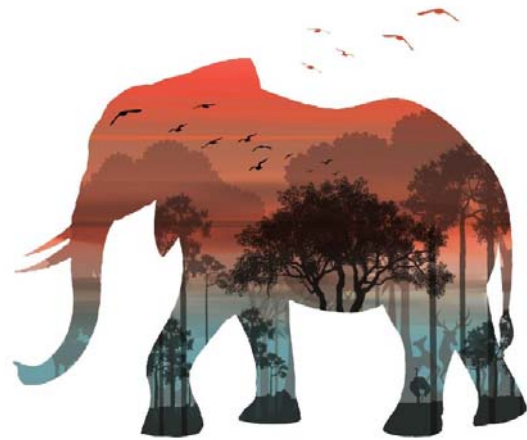
08/session_delete.php

unset() 함수	
형식	unset(mixed \$var, ...)

- unset() 함수는 입력 값으로 주어지는 세션 변수나 변수의 존재를 파괴할 때 사용.
- mixed는 다양한 변수형이 사용될 수 있다는 것을 의미.

[TIP] 세션 변수를 한꺼번에 삭제할 때는 session_unset()을 사용.

PHP 프로그래밍 입문 3판



감사합니다.