

به کارگیری کوکی ها و Session ها

واحد کار یازدهم

پس از مطالعه این واحد کار از فراگیر انتظار می رود که:

- ۱ - مفهوم و کاربرد کوکی ها را بداند.
- ۲ - با اصول تنظیم کوکی ها آشنا باشد.
- ۳ - با اصول تعیین تاریخ انقضا برای کوکی ها آشنا باشد.
- ۴ - اصول استفاده از کوکی ها برای اعتبارسنجی را بداند.
- ۵ - مفهوم و کاربرد Session ها را بشناسد.
- ۶ - متغیرهای Session را بشناسد.
- ۷ - با اصول فعال سازی شماره دسترسی برای کاربر آشنا باشد.
- ۸ - با اصول نگهداری تنظیمات شخصی کاربر آشنا باشد.

زمان (ساعت)	
عملی	نظری
۱۰	۶

مقدمه

کوکی‌ها (Cookie) فایل‌های کوچک متنی هستند که از طرف سرویس‌دهنده‌های اینترنت و توسط مرورگر روی سیستم کاربر ایجاد می‌شوند. کوکی‌ها می‌توانند در موارد مختلفی به سرویس‌دهنده وب و سرویس‌دهنده برنامه کاربردی کمک کنند. به عنوان مثال می‌توانند به سرویس‌دهنده وب کمک کنند تا کاربرانی را که قبلاً به سایت آن‌ها مراجعه کرده‌اند بشناسد، یا این‌که به یادآوری رمز عبور کاربر در مراجعات بعدی کمک کنند. یک کاربر می‌تواند سیستم خود را طوری تنظیم کند که سرویس‌دهنده اجازه تولید کوکی را نداشته باشد و یا این‌که کاربر را هنگام تولید کوکی مطلع کند. کوکی‌ها می‌توانند اطلاعات مهمی را درون خود نگهداری کنند و می‌توانند توسط سرویس‌دهندگان غیرمجاز اطلاعات مهمی از هویت شما را فاش کنند. به عنوان مثال اگر در یک سایت عضو شده باشید، چنانچه یک کوکی برای آن روی هارد سیستم شما ایجاد شده باشد، یک سایت غیرمجاز می‌تواند با اهداف مخرب از طریق خواندن کوکی به رمز عبور شما دسترسی پیدا کند.

نکته کوکی‌ها به تنهایی نمی‌توانند برای سیستم مشکلات امنیتی ایجاد کنند اما می‌توانند در آسیب رساندن به امنیت سیستم شما دخیل باشند.



حداکثر اندازه کوکی‌ها ۴ kB (کیلوبایت) است. یکی از ویژگی‌های کوکی‌ها این است که پس از آن‌که توسط یک مرورگر روی یک سیستم ایجاد شدند، فقط روی همان سیستم می‌توانند خوانده شوند. هر کوکی یک آدرس URL منحصر به فرد دارد و سرویس‌گیرنده از روی این آدرس تعیین می‌کند که با کدام یک کار کند. استفاده از کوکی‌ها در سرویس‌گیرنده‌ها و مرورگرهای مختلف محدودیت‌هایی دارد. به عنوان مثال حداکثر کوکی‌های مرورگر Internet Explorer، ۳۰۰ فایل است.

۱-۱۱ ایجاد کوکی

ایجاد کوکی در فایل ها و اسناد PHP توسط دستور Setcookie انجام می شود.
ساختار کلی این دستور به شکل زیر است:
("پارامتر امن", "اطلاعات دامنه", "تاریخ انقضای کوکی", "مقدار کوکی", "نام کوکی") Setcookie
مثال:

Setcookie ("Webdesign", "#FE593", "01.12.2011, 23:5:35")

توسط تاریخ انقضا می توان اعتبار یا عدم اعتبار کوکی را سنجید، اگر تاریخ انقضای یک کوکی تمام شده باشد، مرورگر دیگر آن را برای سرویس دهنده ارسال نمی کند (تاریخ انقضا با قالب استاندارد GMT^۱ تعیین می شود).

پارامترهای کوکی ها توسط سرویس گیرنده تعیین می شود و می تواند علاوه بر پارامترهای ساختار فوق، پارامترهای دیگری را نیز شامل شود، علاوه بر این ممکن است یک کوکی فقط شامل نام و یک پارامتر دیگر باشد.

اطلاعات دامنه تعیین می کند که کوکی توسط چه دامنه هایی قابل دسترسی است.
اگر پارامتر امن نیز در خصوصیات کوکی تنظیم شده باشد، کوکی فقط مجاز است از طریق کانال های امن مانند https منتقل شود.

۱-۱-۱۱ مهم ترین پارامترهای کوکی

path: برای تعیین مسیر به کار می رود و مقدار پیش فرض آن تمام پوشه های سرویس دهنده است.

domain: دامنه مجاز را تعیین می کند که به طور پیش فرض همان دامنه سرویس دهنده ای است که آن را ایجاد کرده است.

expire: تاریخ انقضای کوکی را مشخص می کند.

secure: این گزینه سطح دسترسی به کوکی را تعیین می کند که به صورت پیش فرض غیرفعال است.

ساختار کلی ارسال کوکی:

HTTP/1.1 200 ok

Date : sun , 20 june 2011 22:44:50 GMT

Server : WAMP

Set-Cookie : Color = red ; expires = sun , 25 june 2011 22:30:00 GMT

Connection : Close

Connect-Type : text/html

۲-۱-۱۱ ایجاد کوکی

کوکی‌ها توسط یکی از دو تابع () Setcookie و () header ایجاد می‌شوند.

کوکی‌های ارسال شده از طرف هر سرویس دهنده در یک آرایه به نام \$_COOKIE ذخیره شده‌اند و دسترسی به آن‌ها مانند سایر متغیرهاست.

به عنوان مثال اگر یک کوکی با نام username داشته باشیم می‌توانیم توسط \$_COOKIE [«username»] به محتوای آن دسترسی پیدا کنیم.

به عنوان مثال می‌خواهیم یک کوکی ایجاد کنیم که تعداد بازدیدهای کاربر را از سایت تعیین کند.

می‌توانیم یک متغیر تعریف کنیم که با هر بار ملاقات توسط کاربر، به مقدار آن یک واحد افزوده شود.

```
<?php
```

```
$count=$_COOKIE["count"];
```

```
$count++;
```

```
Setcookie("count",$count,time()+36000);
```


```
?>
```

به ساختار کد فوق دقت کنید:

در سطر دوم یک متغیر جدید تعریف شده است که در صورتی که اولین بار باشد که

کوکی اجرا می‌شود این متغیر با عدد صفر مقداردهی می‌شود و در سطر سوم به آن یک

واحد اضافه می شود و در خط بعد کوکی ایجاد شده و مقدار متغیر count در آن ذخیره می شود. حال اگر بار دومی باشد که این برنامه اجرا می شود، مقدار count یک واحد دیگر افزوده شده و مقدار ۲ به خود می گیرد و به همین ترتیب در هر بار ملاقات صفحه توسط کاربر، یک واحد به این شمارنده افزوده می شود. در مثال فوق برای تعیین زمان انقضای کوکی از تابع () time استفاده شده است. تابع () time زمان جاری سیستم را نشان می دهد، مقدار تابع time با عدد ۳۶۰۰۰ جمع شده است، ۳۶۰۰۰ بر حسب ثانیه بوده و تعیین کننده ۱۰ ساعت است و این به آن معنی است که کوکی به مدت ۱۰ ساعت انقضا دارد.

 **نکته** • در یک کوکی اگر تاریخ انقضا مشخص نشود، کوکی تا زمانی اعتبار دارد که پنجره IE باز باشد و به محض بستن این پنجره تاریخ انقضای کوکی نیز به پایان خواهد رسید.

• هر تابع یا دستوری که قبل از تابع Setcookie در برنامه نوشته شود، اجرا نمی شود، لذا دستورات چاپ و ... را هرگز قبل از این دستور قرار ندهید و همیشه ساختار کوکی را در ابتدای صفحه پیاده سازی کنید.

۳-۱-۱۱ دسترسی به اطلاعات cookie

cookie های ارسال شده از طرف هر سرویس دهنده در یک آرایه به نام \$_COOKIE ذخیره شده اند و دسترسی به آن ها مانند سایر متغیرهاست. به عنوان مثال اگر یک cookie با نام username داشته باشیم می توانیم توسط \$_COOKIE["username"] به محتوای آن دسترسی پیدا کنیم.

مثال: می خواهیم یک cookie ایجاد کنیم که تعداد بازدیدهای کاربر را از سایت تعیین کند. برای این کار ابتدا یک متغیر تعریف می کنیم که با هر بار ملاقات توسط کاربر به مقدار آن یک واحد افزوده شود.

```
<?php
```

```
$count=$_COOKIE["count"];
```

```
$count++;  
Setcookie("count",$count,time()+36000);
```

?>

به ساختار کد فوق دقت کنید:

در سطر دوم یک متغیر جدید تعریف شده است که اگر اولین بار باشد که cookie اجرا می شود این متغیر با عدد صفر مقداردهی می شود و در سطر سوم به آن یک واحد اضافه می شود و در خط بعد cookie ایجاد شده و مقدار متغیر count در آن ذخیره می شود. حال اگر بار اول اجرای این برنامه نباشد، به مقدار قبلی count یک واحد افزوده شده و به همین ترتیب در هر بار ملاقات صفحه توسط کاربر، یک واحد به شمارنده افزوده می شود.

سؤال: تاریخ انقضای cookie در مثال فوق چه مدت است؟

نکته • در یک cookie اگر تاریخ انقضای مشخص نشود، cookie تا زمانی زنده است که پنجره IE باز باشد و به محض بستن این پنجره زمان انقضای cookie نیز به پایان خواهد رسید.

• هر تابع یا دستوری که قبل از تابع Setcookie در برنامه نوشته شود، اجرا نمی شود، لذا دستورات چاپ و... را هرگز قبل از این دستور قرار ندهید و همیشه ساختار cookie را در ابتدای صفحه پیاده سازی کنید.

۴-۱-۱۱ تابع isset()

یکی از توابع مهم و کاربردی در PHP تابع isset() است، توسط این تابع می توان از معرفی شدن متغیرها اطمینان حاصل کرد.

اگر متغیر مورد نظر قبلاً تعریف شده باشد، مقدار true و در غیر این صورت مقدار false را بر می گرداند.

فرض کنید در برنامه ای به مقدار یک متغیر نیاز داریم، اما نمی دانیم که این متغیر قبلاً معرفی و مقداردهی شده است یا خیر، در این صورت می توانیم از تابع isset() برای بررسی این مسأله استفاده کنیم.

مثال:

```
if (isset($first_name))  
{  
print '$first_name is set';  
}
```

در PHP علاوه بر دستور echo می توان از دستور print برای نمایش در خروجی استفاده نمود.

تمرین 

با استفاده از تابع `isset()` تعداد بازدیدهای کاربر از سایت را به دست آورید.

۵-۱-۱۱ حذف یک کوکی

برای حذف یک کوکی فقط کافی است تابع `Setcookie` را با نام فایل فراخوانی کنیم:
`Setcookie ("نام کوکی")`;

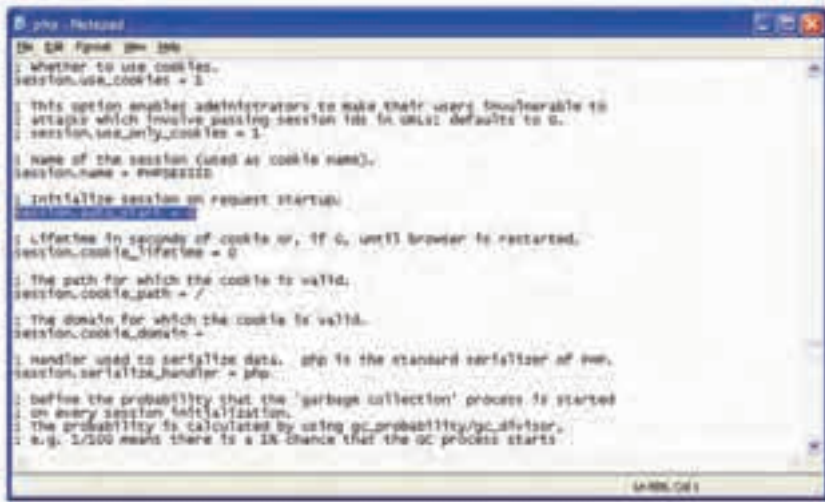
۲-۱۱ جلسه Session

Session یکی از مباحث اصلی شبکه و اینترنت است و توسط آن می توان اطلاعات کاربران را نگهداری کرد. هر کاربر که به سایتی وارد می شود با اجرای اسکریپت PHP یک جلسه ایجاد می شود و در صورتی که اولین بار باشد که وارد آن سایت می شود یک شناسه منحصر به فرد به او اختصاص داده می شود.

جلسه با دستور `Session_Start ()` شروع و ایجاد می شود.

شناسه جلسه توسط دستور `Session_id ()` تعیین می شود.

در فایل `PHP.ini` یک شناسه با عنوان `Session.auto_Start` وجود دارد، این شناسه به طور پیش فرض مقدار صفر دارد، اگر این مقدار را به ۱ تغییر دهید، برای هر سند PHP به طور خودکار یک جلسه ایجاد می شود اما اگر مقدار آن صفر باشد باید با استفاده از دستورات برنامه نویسی برای ایجاد جلسه اقدام شود.



شکل ۱-۱۱

زبان PHP اطلاعات جلسه را در فایل‌ها ذخیره می‌کند، مسیر ذخیره این فایل‌ها از طریق دستور `Session.Save_Path` تعیین می‌شود. پس از ایجاد یک جلسه می‌توان شناسه آن را به دست آورد.

```
<code>
</code>
```

۱-۲-۱۱ متغیرهای جلسه

هر جلسه می‌تواند تعداد زیادی متغیر داشته باشد و توسط آن‌ها می‌توان اطلاعات کاربران را در صفحات مختلف مدیریت کرد.

این متغیرها نیز در یک آرایه فوق سراسری نگهداری می شوند.
تعریف متغیرهای جلسه به صورت زیر است:

مقدار متغیر [= 'نام متغیر'] \$ SESSION

مثال: \$ SESSION ['number'] = 1239;

مثال: \$ SESSION ['name'] = "Zahra";

پس از تعریف متغیرهای جلسه در یک صفحه، در صفحات دیگر می توان به آنها دسترسی پیدا کرد، دسترسی به متغیرهای جلسه با همان نامی صورت می گیرد که هنگام تعریف متغیرها به کار رفته بود.

به عنوان مثال متغیرهای جلسه فوق در فایل دیگری از همان سایت از طریق دستورات زیر می توانند در خروجی چاپ شوند.

```
echo "$_SESSION ['number']";
```

```
echo "$_SESSION ['name']";
```

۱۱-۲-۲ حذف جلسه

حذف جلسه از طریق کد () Session_destroy صورت می گیرد.

اگر این تابع در یک جلسه استفاده شود، تمام متغیرها و شناسه های جلسه نیز حذف می شوند.

```
<?php
```

```
Session_start ();
```

```
Session_destroy ();
```

```
?>
```

۱۱-۲-۳ نگهداری تنظیمات شخصی کاربر

همان طور که گفته شد جلسه ها می توانند متغیرهای زیادی در صفحه ایجاد کنند، این متغیرها می توانند حتی شامل اطلاعات محرمانه کاربر باشند و به دلیل این که اطلاعات جلسه به صورت یک آرایه فوق سراسری ذخیره می شود، می تواند مورد سوء استفاده هکرها قرار گیرد به همین دلیل باید به نوعی به دنبال مخفی کردن این اطلاعات بود.

رمزگذاری اطلاعات جلسه روشی است که توسط PHP قابل پشتیبانی است، این عمل توسط دستور () session_encode صورت می گیرد، در مقابل اگر اطلاعات session بخواند مورد استفاده فایل PHP قرار گیرد باید از حالت رمز خارج شود و این عمل توسط دستور () session_decode صورت می گیرد.

واژه‌نامه

Expire	تاریخ انقضا
Path	مسیر
Secure	ایمن
Session	جلسه

خلاصه مطالب

- کوکی‌ها فایل‌های متنی هستند که از طرف سرویس‌دهنده‌های اینترنت و توسط مرورگر روی سیستم کاربر ایجاد می‌شوند.
- حداکثر اندازه فایل‌های کوکی ۴ کیلوبایت است.
- کوکی‌ها به تنهایی نمی‌توانند برای سیستم مشکلات امنیتی پیش بیاورند، اما می‌توانند در آسیب رساندن به امنیت سیستم دخیل باشند.
- کوکی‌ها پس از این‌که توسط مرورگر روی یک سیستم ایجاد شدند، فقط روی همان سیستم می‌توانند خوانده شوند.
- در PHP برای ایجاد کوکی از دستور Setcookie استفاده می‌شود.
- یکی از مهم‌ترین ویژگی‌های کوکی‌ها، تاریخ انقضا است. توسط تاریخ انقضا می‌توان اعتبار یا عدم اعتبار کوکی را سنجید. اگر تاریخ انقضای یک کوکی تمام شده باشد، مرورگر دیگر آن را برای سرویس‌دهنده ارسال نمی‌کند.
- جلسه یکی از مباحث اصلی شبکه و اینترنت است، توسط جلسه می‌توان اطلاعات کاربران را نگهداری کرد، در یک صفحه وب که به زبان PHP نوشته شده است، به محض اجرای اسکریپت `<?php` یک جلسه ایجاد می‌شود.
- جلسه با دستور `Session_Start()` ایجاد و شناسه آن توسط دستور `Session_id()` تعیین می‌شود.

آزمون نظری

- ۱ - حداکثر اندازه فایل های کوکی چقدر است؟
الف- kB1 ب- kB4 ج- MB1 د- MB4
- ۲ - حداکثر کوکی های مرورگر IE چند عدد است؟
الف- ۳۰ ب- ۳۰۰ ج- ۱۰۰ د- نامحدود
- ۳ - کدام یک از دستورات زیر برای ساخت کوکی به کار می رود؟
الف- Setcookie ب- Pathcookie
ج- Cookie د- Savecookie
- ۴ - برای حذف یک کوکی باید کدام یک از دستورات زیر استفاده شود؟
الف- Delete ب- DeleteCookie
ج- Setcookie د- Removecookie
- ۵ - پارامتر domain به طور پیش فرض کدام دامنه مجازی را برای کوکی در نظر می گیرد؟
الف- com ب- org ج- net د- دامنه سرویس دهنده
- ۶ - دستور حذف یک کوکی چند پارامتر ورودی دارد؟
الف- ۱ ب- ۲ ج- ۳ د- پارامتر ورودی ندارد.
- ۷ - پس از تعریف متغیرهای جلسه در یک صفحه، دسترسی به آنها در صفحات دیگر با چه نامی صورت می گیرد؟
الف- با نام صفحه جدید
ب- با همان نام متغیر جلسه
ج- با نام صفحه ای که در آن تعریف شده است.
د- دسترسی به متغیرهای جلسه در صفحات دیگر امکان پذیر نیست.
- ۸ - برای حذف جلسه از چه دستوری استفاده می شود؟
الف- Sessiondestroy () ب- Sessionkill
ج- Session_destroy () د- Session_kill ()

آزمون عملی

- ۱ - یک کوکی جدید با عنوان Color با مقدار #ffffff با تاریخ انقضای به مدت ۱ ساعت بسازید.
- ۲ - تاریخ انقضای کوکی سؤال اول را طوری تنظیم کنید که به محض بستن مرورگر، کوکی نیز از بین برود.
- ۳ - تنظیمات سیستم را طوری انجام دهید که برای هر سند به طور خودکار یک جلسه ساخته شود.
- ۴ - در یک فایل با نام Page1 یک جلسه ایجاد کنید.
- ۵ - در یک فایل با نام Page2 پارامترهای جلسه Page1 را چاپ کنید.