

## Hoofdstuk 9 : Werken met sessions

### 9.1 Wat zijn sessions?

Wanneer u (kort) na elkaar twee verschillende pagina's op dezelfde website opvraagt, heeft bij het bezoeken van de tweede pagina, de server er geen flauw benul van dat u daarnet een andere pagina heeft bezocht. Wat de server betreft bent u gewoon een nieuwe bezoeker die toevallig een pagina opvraagt.

Stel dat u op pagina1 inlogt, en vervolgens klikt op een andere pagina..dan is het behoorlijk vervelend dat de server al niet meer weet dat je op pagina1 ingelogd hebt.

Een session is een periode waarin je webbrower een aantal webpagina's van dezelfde server opvraagt. Gedurende deze periode kunnen alle variabelen die voor deze session bekend zijn, gebruikt worden op de diverse pagina's.

De maximale duur van een session wordt ingesteld door je webhoster. De standaard waarde is 24 minuten. Als u dus na 25 minuten weer een pagina opvraagt, zult u terug moeten inloggen.

Bij het opstarten van een session, wordt door PHP een identificatie nummer aangemaakt. Met noemt dit de session ID. Dit nummer is beschikbaar dmv het commando `session_id()`.

Wanneer een session wordt gestart, wordt dit ID nummer steeds meegestuurd, waardoor de server de variabelen die in de session zijn gedefinieerd kan meenemen naar een andere pagina.

### 9.2 Hoe wordt een session gestart?

Daarvoor gebruiken we het commando `session_start()`.

Deze regel MOET als eerste regel in een PHP bestand opgenomen worden. Dus voor het begin van eventuele HTML code. Er mag zelfs geen witregel in het script staan.

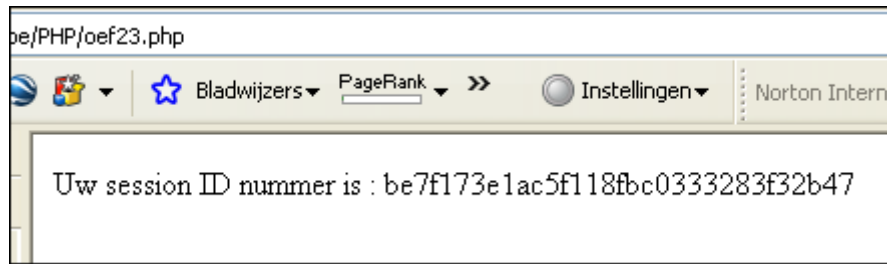
Open `start.php`, en voeg volgende code toe VÓÓR het de `<DOCTYPE...>` lijn.

```
<?php
session_start();
$sessionid = session_id();
?>
```

Voeg nu volgende code toe in de `<body>`

```
<?php
echo ("Uw session ID nummer is : $sessionid");
?>
```

Sla alles op als **oef23.php**. Download d.mv. FTP het bestand naar de map PHP op de server. Open het bestand dmv je browser (<http://www.mijnsite.com/PHP/oef23.php>)



Het identificatie nummer die jij zult zien, zal verschillend zijn. Iedere session heeft een uniek nummer.

### 9.3 Het definiëren van session variabelen

Zodra een session is gestart op de server, kunt u gegevens opslaan binnen die session. Hiervoor maakt u gebruik van het commando `session_register( "naam variabele" )`.

Bemerk dat we hier geen \$ teken gebruiken, zoals we tot nu toe altijd gedaan hebben bij definities van variabelen.

Open start.php, en voeg volgende code toe VOOR de <DOCTYPE> regel.

```
<?php
session_start( );
$username= "webgarden";
session_register( "username" );
?>
```

Voeg volgende code toe in de <body>

```
<h3>Uw gebruikersnaam is:<?php echo($_SESSION("username")); ?></h3>
```

Sla alles op als **oef24.php**. Download d.mv. FTP het bestand naar de map PHP op de server. Open het bestand dmv je browser (<http://www.mijnsite.com/PHP/oef24.php>)



We openen nu terug oef24.php. We laten bovenaan de volgende regels WEG:

```
$username= "webgarden";  
session_register ("username");
```

In de <body> vervangen we de <h3> regel door:

```
<h3>Uw gebruikersnaam is nog steeds:<?php echo($_SESSION["username"]); ?></h3>
```

We slaan alles op in **oef24B.php**. Download d.mv. FTP het bestand naar de map PHP op de server.

Open het bestand dmv je browser (<http://www.mijnsite.com/PHP/oef24B.php>)



We hebben dus een volledig nieuwe pagina geopend (oef24B). In deze pagina hebben we wel een session\_start () commando gegeven, maar geen variabele \$username gedefinieerd.

PHP heeft gemerkt dat er voor je browser al een session was gestart (oef24.php). De aanwezige session-variabelen worden dus overgenomen, en gebruikt in oef24B.php.

#### **9.4 Een login systeem maken**

Je hebt het ongetwijfeld al meegemaakt dat je jezelf moet inloggen op een site, om bepaalde informatie te kunnen krijgen.

We gaan hier samen zo een inlog systeem maken.

Let wel, het zal geen echt werkend systeem zijn. Dit vanwege de heel eenvoudige redenen dat we nog geen toegang hebben tot geregistreerde gebruikersnamen en paswoorden.

Dit kunnen we pas, als we in de volgende hoofdstukken het nodige weten over databases.

In werkelijkheid werkt het als volgt:

Iemand registreert zich op een site. Hij typt een gebruikersnaam, een paswoordn, eene-mail adres,...in een registratie-formulier in.

De ingebrachte gegevens worden door PHP naar een database op de server verstuurd.

Als de bezoeker zich vervolgens wil inloggen, typt hij zijn gebruikersnaam en paswoord in. Dan wordt gecontroleerd of deze al in de database zijn opgenomen.

Bij deze oefening kunnen we nog niet vergelijken met gegevens uit een database. M.a.w ieder combinatie van gebruikersnaam en paswoord zal hier werken.

We maken uiteindelijk 3 pagina's.

- login.php : De pagina waarop je kunt inloggen.
- geheim.php : De pagina die zichtbaar wordt als je ingelogd bent
- logout.php : De pagina die zorgt voor uitloggen (afsluiten sessie)

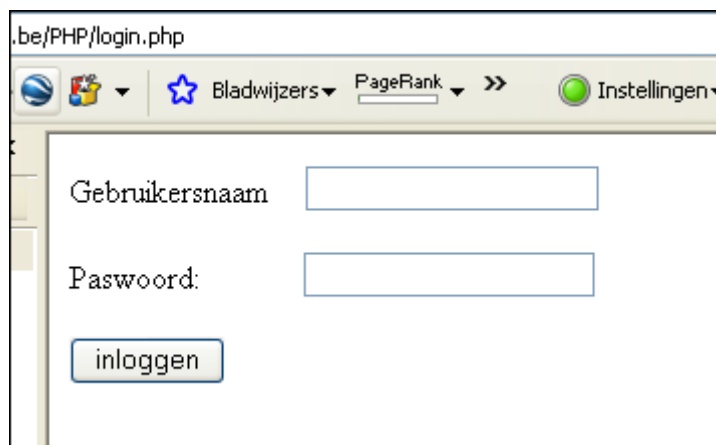
#### 9.4.a : login.php

Open start.php en voeg onderstaande code toe in de <body>

```
<form name="reset" method="post" action="login.php">
<p><span style="margin-right:50px;">Voornaam</span>
  <input name="username" type="text" id="vn">
</p>

<p><span style="margin-right:52px;">Paswoord:</span><input name="paswoord"
type="password" id="pw"></p>
<p><input type="submit" name="submit" value="inloggen"> </p>
</form>
```

Sla op in **login.php**. Download d.mv. FTP het bestand naar de map PHP op de server. Open het bestand dmv je browser (<http://www.mijnsite.com/PHP/login.php>).



Zoals u kunt zien in de HTML code, roepen we via action het bestand zelf op.

Er is een andere manier om het actieve bestand aan te roepen.

Verander action="login.php" door

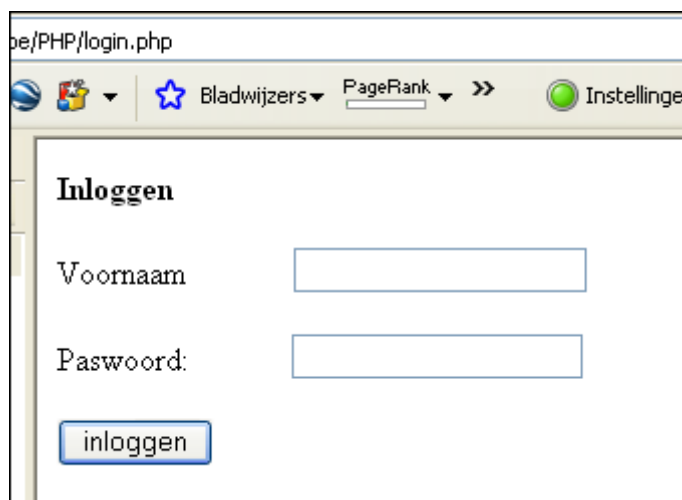
```
action="<?php echo ($_SERVER["PHP_SELF"]);?>"
```

Het voordeel van deze methode is het action commando onafhankelijk is van de naam van het bestand. Stel dat we dit stukje code zouden gebruiken in een andere pagina, dan moeten we niet steeds de naam van het bestand opgeven bij 'action'.

Open login.php, en voeg volgende code toe helemaal bovenaan de pagina (**VÓÓR** de <DOCTYPE> regel)

```
<?php
session_start();
$action=$_POST["submit"];
if ($action=="inloggen"){
if(empty($_POST["username"])){
echo(" u veld 'voornaam' is leeg");}
else{
$_SESSION["username"] = $_POST["username"];
if (empty($_POST["password"])){
echo("U heeft geen paswoord ingebracht");}
else{
$_SESSION["password"] = $_POST["password"];
header("Location:geheim.php");}} }
else{
echo("<b>Inloggen</b>");}
?>
```

Sla terug op als **login.php**. Download d.mv. FTP het bestand naar de map PHP op de server. Open het bestand dmv je browser (<http://www.mijnsite.com/PHP/login.php>).



The screenshot shows a web browser window with the address bar displaying 'pe/PHP/login.php'. The browser's toolbar includes icons for search, home, and settings, along with 'Bladwijzers', 'PageRank', and 'Instellingen' menus. The main content area displays a login form with the title 'Inloggen'. It contains two text input fields: 'Voornaam' and 'Paswoord:'. Below these fields is a blue button labeled 'inloggen'.

Probeer even in te loggen zonder iets in te vullen. Probeer in te loggen zonder paswoord in te vullen. Probeer in te loggen zonder voornaam, maar met paswoord in te vullen

Een script die verre van volledig is, maar je toch verplicht om alles in te vullen.

Er is wel een woordje uitleg nodig bij de PHP code. Laat me toe, met onderstaande beschrijving aan te geven wat we doen:

We beginnen met de waarde van de knop submit te lezen (name='submit' - `$_POST[ 'submit' ]`).

De uitgelezen waarde bewaren we als de variabele '\$actie'.

Als iemand de knop 'inloggen' heeft aangeklikt, dan is de waarde ='inloggen'. Hiervoor verwijst ik naar de gebruikte HTML code

```
<input type="submit" name="submit" value="inloggen">
```

Dus als iemand aangeklikt heeft dan moet de waarde van \$actie gelijk zijn aan 'inloggen'

Als dit zo is (`if ( $actie=="inloggen" )`), dan gaan we na of het veld 'voornaam' (name="username") leeg is (`if( empty( $_POST[ "username" ] )`)).

Als het inderdaad leeg is dan, volgt er de boodschap "u veld 'voornaam' is leeg". (echo lijn).

Als het niet leeg is, dan wordt gekeken of het veld 'paswoord' leeg is (`( empty( $_POST[ "paswoord" ] )`)).

Als het inderdaad leeg is, dan komt via een echo-lijn de boodschap "U heeft geen paswoord ingebracht".

We keren even terug naar het veld 'voornaam'. Als dit veld NIET leeg is, dan slaan we de inhoud van het veld 'voornaam' met name="username" (`$_POST[ "username" ]`) op in een session variabele (`$_SESSION[ "username" ]`). Door het op te slaan in een SESSION variabele zullen we het straks kunnen gebruiken op een andere pagina.

```
$_SESSION[ "username" ] = $_POST[ "username" ];
```

Als er inderdaad in het veld 'paswoord' (met name="paswoord") iets hebben ingebracht, dan wordt de inhoud van dit veld (`$_POST[ "paswoord" ]`) opgeslagen in een session variabele (`$_SESSION[ "paswoord" ]`). Door het op te slaan in een SESSION variabele zullen we het straks kunnen gebruiken op een andere pagina.

```
$_SESSION[ "paswoord" ] = $_POST[ "paswoord" ];
```

Als zowel iets ingevuld is in veld 'voornaam' EN in veld 'paswoord' dan volgt er tevens het commando

```
header( "Location:geheim.php" )
```

Dit is het PHP commando om een nieuwe pagina met naam 'geheim.php' te laden.

Zoals het hier is weergegeven bevinden logging.php en geheim.php zich in dezelfde directory.

#### 9.4.b geheim.php

Nu maken we de pagina die geladen wordt indien een voornaam en paswoord is ingebracht.

Open start.php, en breng volgende code in de <body>

```
<?php
if (!empty($_SESSION["username"])){
    echo("<h3>Ons kleine geheimpje:<br> Je voornaam is: " . $_SESSION["username"]
    ."<br>
    Je wachtwoord is: " . $_SESSION["paswoord"]. " </h3>");
    echo("<br><a href='\"logout.php\"'>Uitloggen</a>");
    // overige 'geheime' gegevens :-)
}else{
    echo("<h2>U hebt zich nog niet aangemeld;<br>u kunt zich
    <a href='\"login.php\"'>hier registreren</a></h2>");
}
?>
```

Voor de <DOCTYPE> lijn plaats je nu volgende code

```
<?php session_start(); ?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

Sla op als **geheim.php**. Download d.mv. FTP het bestand naar de map PHP op de server. Open het bestand dmv je browser (<http://www.mijnsite.com/PHP/geheim.php>).

Open in je browser



Klik nu op 'hier registreren'. Dan wordt terug verwezen naar onze login.php. Vul nu een voornaam en paswoord in. En klik dan op inloggen



(Ik heb als voornaam 'wees' ingetypt en als paswoord 'welkom')

M.a.w we hebben de session variabelen uit `login.php` overgenomen en gebruikt in `geheim.php`.

Een woordje uitleg bij de code:

```
if ( !empty($_SESSION["username"]) )
```

We kijken of de SESSION variabele ["username"] NIET leeg is. Let dus op het uitroepteken voor empty.

We hebben keurig ingelogd...dan kan het dus niet leeg zijn.

In dit geval komt er een echo-lijn die begint met "Ons kleine geheimpje:....".

In die echo-lijn geven we de ingebrachte voornaam (`$_SESSION["voornaam"]`) en het ingebrachte paswoord (`$_SESSION["paswoord"]`) weer op het scherm.

#### 9.4.c. `logout.php`

Nu rest er ons alleen nog de `logout.php` pagina te maken.

Open `start.php`. Voeg onderstaande code toe VOOR de `<DOCTYPE>` regel.

```
<?php session_start();  
session_unset();  
session_destroy();?>  
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

Voeg in de `<body>` volgende HTML code toe:

```
<p><h3>U bent uitgelogd</h3></p>
<hr>
<p>De opgevraagde pagina <a href="geheim.php">geheim.php</a> is niet meer
beschikbaar</p>
<p>Indien u de pagina terug wil bekijken, dient u eerst <a href="login.php">in te
loggen</a></p>
```

Sla op als **logout.php**. Download d.mv. FTP het bestand naar de map PHP op de server. Open het bestand dmv je browser (<http://www.mijnsite.com/PHP/geheim.php>).

Open in de browser



Als we even kijken naar de PHP code:

```
session_unset();
```

Dit commando geeft alle variabelen vrijdag

```
session_destroy();
```

Dit commando sluit de sessie af.

PS: Als laatste opmerking bij deze oefening geef ik tenslotte een tekortkoming van dit script weer.

Als je naar de login.php gaat, en je brengt spaties in de velden 'voornaam' en 'paswoord' (maw geen letters of cijfers) dan kun je toch inloggen☺

Maar het was hier niet de bedoeling om een volledig afgewerkt login script te schrijven.