

3

VERSCHILLENDENDE VARIABELEN

3.1

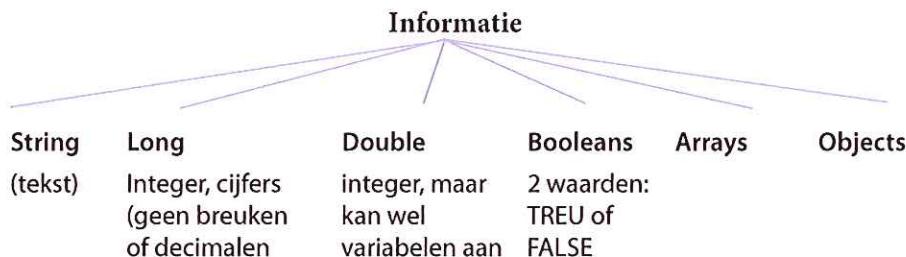
Informatie, rekenen en arrya's

Zoals elke programmeertaal, beschikt PHP over variabelen.

Variabelen in PHP zijn te herkennen aan het dollar teken (\$), een variabele genaamd 'var' zal er dus uitzien als: \$var.

Elke variabele wordt voorafgegaan door een dollarteken \$

Een variabele kan maar een van deze zijn.



String

Een **String** is een stuk tekst. Het maakt niet uit hoe lang een string is, het is altijd een tekst en kan ook getallen bevatten (zoals een telefoonnummer of een postcode). Zoals eerder gezegd begint elke variabele met een \$. Verder moet de naam beginnen met een letter a tot z, A. tot Z of een plat streepje (underscore). Deze worden zo gedeclareerd:

`$var = "STRING"; // waarbij STRING een stuk tekst is.`

Voorbeeld: `$var = "Groningen";`

Long

Een **Long** (kortweg 'integer') is altijd **een getal**. Dit getal ligt meestal tussen (-2^31-1) en (2^31-1). Integers zijn ook altijd natuurlijke getallen, **dus geen breuken of decimalen**.

Deze worden zo gedeclareerd:

`$var = n; // waarbij n een getal is`

Voorbeeld: `$var=25;`

Double

Een **Double** is net zoals een integer een getal, alleen kan het wél **decimalen** aan.

Deze worden zo gedeclareerd:

\$var = n.m; // waarbij n.m een getal is (met evt. decimalen)

Voorbeeld: **\$var= 12.95;**

Booleans

Booleans hebben maar 2 waarden, **WAAR** en **ONWAAR**, of in het engels

TRUE en **FALSE**, ze kunnen ook 1 en 0 zijn, maar in PHP wordt vooral gebruik gemaakt van **TRUE** en **FALSE**

Deze worden zo gedeclareerd:

\$var = TRUE; // wat dus alleen een lege variabele maakt

\$var = FALSE; // FALSE betekent dat het er niet is, en als een \$var dus FALSE is, zal die niet aangemaakt worden.

Arrays

Arrays zijn een speciaal onderdeel van PHP. Arrays kan je het beste voorstellen als tabellen met maar 2 kolommen. De eerste kolom genaamd KEY en de tweede VALUE.

Je kunt array ook voorstellen als soort bibliotheek, waar allemaal boeken in staan, die weer geraadpleegd kunnen worden.

Je kan in 1 PHP-Array verschillende (en grote hoeveelheden) data opslaan.

De VAL (value: waarde) kan elk type zijn behalve een Object. Het kan dus ook zelf een array zijn!

Deze worden zo gedeclareerd:

\$array[KEY] = "VAL" // Waarbij KEY de sleutel is, die verwijst naar de VALUE of:

\$array = array (key => 'val' key2 => 'val2');

Je kunt Arrys aanroepen met echo "**\$array[key]**";

Nu opent je array met KEY, dus je krijgt VAL te zien.

Objects

Objects zijn zeer aparte onderdelen van PHP en eigenlijk niet volledig ondersteund. Objecten zijn grote hoeveelheden functies en variabelen.

Deze worden zo gedeclareerd:

```
CLASS name {  
function(){  
}  
$var = name;  
}
```

3.2 Een mooie opmaak

We gaan vanaf de volgende opdracht een mooie opmaak aan onze opdrachten geven. Om alle opdrachten dezelfde opmaak te geven, maken we eerst een mooi CSS bestand (**opmaak.css**):

CSS bestand maken

1. Open het programma Crimson Editor
2. Typ de volgende codes over.

```
body {background-color: Greenyellow }
h1 {
    background-color: blue;
    color: Yellow;
    font-size: 25px;
    font-weight: normal;
    text-align: center;
    border: 10px outset Tomato;
}
h2 {
    background-color: black;
    color:white;
    font-size: 20px;
    text-align: center;
    border: 8px outset red;
}
div#onder {
    background-color: Powderblue;
    padding: 20px;
    border: 10px solid Darkblue;
    margin: 5px 100px 5px 100px;
    text-align: center;
}
div#tekstblok {
    background: Lightcyan;
    border: 10px ridge Olive;
    color: black;
    font-family: "Arial";
    font-size: 14px;
    margin: 20px 60px;
    padding: 10px;
}
b {
    display: inline;
    background-color: yellow;
}
```

```
.button {  
background: black;  
border: 8px outset red;  
color: white;  
font-family: "Arial black";  
text-align:center;  
font-weight: normal;  
font-size: 12px;  
margin: 4xp 20px;  
padding: 4px 20px;  
}  
  
form input {  
    color: #220022;  
    font-family: "Comic Sans MS";  
    font-size: 12px;  
}  
  
table{border:outset 15pt; border-color:Orange}  
TH {font-family:Arial; font-weight:bold; background-color:yellow; color:red;}  
TR{ background-color:gold;color:black}  
TR.even { background-color:gold;color:black }  
TR.oneeven {background-color:Greenyellow; color:red; }  
Td{border:inset 3pt;}  
  
a.:link {border-style: outset; width:150; background-color:black; text-align:left;text-decoration: none; color:yellow; font-family:arial;font-size:12;}  
a.:visited {border-style: outset; width:145; background-color:black; text-align:left;text-decoration: none; color:yellow; font-family:arial;font-size:12;}  
a.:hover {border-style: outset; width:155; background-color:gray; text-align:center;text-decoration: none; color:ffffff; font-family:arial;font-size:12;}  
a.:active {border-style: inset; width:155; background-color:silver; text-align:right; text-decoration: none; color:black; font-family:arial;font-size:12;}
```

3. Sla deze op onder de naam: **opmaak.css**

Voorbeeld 3-1: String (tekst)

Een String is een stuk tekst. Het maakt niet uit hoe lang een string is, het is altijd een tekst, en kan ook getallen bevatten.

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>voorbeeld3-1</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<h1> String</h1>
<?php
$naam1= "Y.Gareb";
$naam2= "ICT-opleiding";
$naam3= "PHP-lessen";
$naam4= "Groningen";
echo "Ik krijg $naam3 van de heer $naam1. Ik volg een $naam2 in $naam4.";
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-1.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-1.php. <http://localhost/Voorbeeld3-1.php>, dan zie je het volgende venster:



Voorbeeld 3-2: String (tekst)

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>voorbeeld3-2</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<?php
$boek1 = "Hardware";
$boek2 = "Software";
$boek3 = "ECDL";
$boek4 = "Webdesign 1 (HTML, CSS en Flash)";
$boek5 = "Webdesign 2 (PHP en My SQL)";
$kopregel = "<H2>Mijn favoriete ICT boeken:</H2>";

echo "$kopregel";
echo "<ul>";
echo "<li>$boek4";
echo "<li>$boek5";
echo "</ul>";
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-2.php**
(in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-2.php. <http://localhost/Voorbeeld3-2.php>, dan zie je het volgende venster:



Opdracht 3-A

- Maak de onderstaande webpagina zo nauwkeurig mogelijk na. Gebruik voor iedere naam een variabele.



- Sla de pagina op onder de naam: **Opdracht 3-A.php**
- Wat moet je voor de opdracht inleveren:
 - Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

Opdracht 3-B

- Bedenk een opdracht over dit onderwerp (**String- Variabel**).
- Sla deze opdracht op onder de naam **Opdracht 3-B**.
- Wat moet je voor de opdracht inleveren:
 - Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren..

3.3 Getallen en operatoren

Long (integer)

Een **Long** (kortweg 'integer') is altijd een getal. Dit getal ligt meestal tussen (-2³¹-1) en (2³¹-1). Integers zijn ook altijd natuurlijke getallen (**zonder dubbele aanhalingstekens**), dus geen breuken of decimalen.

Deze worden zo gedeclareerd:

`$var = n; // waarbij n een getal is(zonder dubbele aanhalingstekens)`

Voorbeeld: `$var= 25;`

Double (decimalen)

Een **Double** is net zoals een integer een getal, alleen kan het wél **decimalen** aan.

Deze worden zo gedeclareerd:

`$var = n.m; // waarbij n.m een getal is (met evt. decimalen)`

Voorbeeld: `$var= 12.95;`

Net zoals met iedere programmeertaal, kun je ook rekenen met PHP. Het is niet mijn bedoeling om jullie hier wiskunde lessen te geven, we houden het hier op de eenvoudige rekenkundige bewerkingen, leerstof van het lager onderwijs.

De rekenkundige operatoren

Heel eenvoudig, je hoeft slechts de 7 rekenkundige operatoren te gebruiken, die je kent uit spreadsheets:

+,-,* , / , kwadraat, wortel en machten.

De voornaamste rekenkundige operatoren

Operator	Betekenis	Voorbeeld
+	optellen	<code>\$a=10+\$b</code>
-	aftrekken	<code>\$c=\$a-1</code>
*	vermenigvuldigen	<code>10*3</code>
/	delen	<code>\$a/\$g</code>
Kwadraat	(geen functie)	<code>\$a = 9; \$kwadraat = \$a * \$a; echo \$kwadraat; // 81</code>
wortel	$\sqrt{}$ sqrt	<code>\$a = 9; \$wortel = sqrt(\$a); echo \$wortel; // 3</code>
machten	pow	<code>\$a = 9; \$macht = pow(\$a,3); echo \$macht; // 729</code>
++	1 toevoegen	<code>\$a++ (is gewoon de korte schrijfwijze van \$a=\$a+1)</code>
--	1 aftrekken	<code>\$a-- (\$a=\$a-1)</code>

Voorbeeld 3-3 Long of integer (Getallen)

Een **Long** (kortweg 'integer') is altijd **een getal**. Integers zijn ook altijd natuurlijke getallen (**zonder dubbele aanhalingsstekens**), dus **geen breuken of decimalen**.

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>voorbeeld3-3</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<h1>De som</h1>
<?php
$getal1= 25;
$getal2= 50;
$som= $getal1+$getal2;
echo ("De som van $getal1 +$getal2 = $som");
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-3.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-3.php. <http://localhost/Voorbeeld3-3.php>, dan zie je het volgende venster:

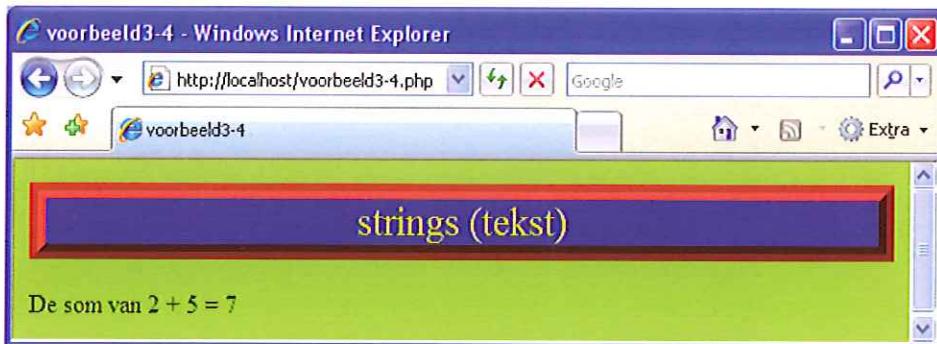


Voorbeeld 3-4 strings (tekst met dubbele aanhalingstekens)

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>voorbeeld3-4</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<h1> strings (tekst)</h1>
<?php
$getal1= "2";
$getal2= "5";
$som= $getal1+$getal2;
echo ("De som van $getal1 + $getal2 = $som");
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-4.php**
(in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-4.php. <http://localhost/Voorbeeld3-4.php>, dan zie je het volgende venster:



In **Voorbeeld 3-3** zijn **\$getal1** en **\$getal2** getallen. In de tweede **voorbeeld3-4** zijn de **strings** (tekst met dubbele aanhalingstekens) en zou je verwachten dat PHP daar "25" als resultaat geeft.

Optellen van variabelen is altijd **numeriek**

Wat getoond wordt, is vreemd. Het verschijnt niet zoals verwacht

De Som van \$getal1 en \$getal2 is \$som

Het resultaat van **voorbeeld3-4** wordt juist weergegeven, ondanks dat "2" en "5" als **strings** gebruikt. Dit maakt het toch wel flexibel.

3.5 Concatineren (aan elkaar plakken)

Als je het strikt zou willen dan moet je **concatineren** of aan elkaar plakken. Wil je als resultaat 25 krijgen, het aan elkaar plakken van tekst dus, dan doe je als volgt:

Teksten aan elkaar zetten doe je met een **punt (.)**

Voorbeeld 3-5 Teksten aan elkaar zetten met een punt (.)

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>Voorbeeld3-5</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<?php
echo "<H2> Getallen aan elkaar plakken doe je met een punt(.)</H2>";
$getal1=2;
$getal2=5;
$som= $getal1 . $getal2;
echo ("$getal1 plakt aan $getal2 = $som");
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-5.php**
(in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-5.php. <http://localhost/Voorbeeld3-5.php>, dan zie je het volgende venster:



Voorbeeld 3-6

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>voorbeeld3-6</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<?php
echo "<h1> Het verschil</h1>";
$getal1=5;
$getal2= 2;
$aftrekken= $getal1 - $getal2;
echo ("Het verschil tussen \"$getal1.\" en \"$getal2.\" is $aftrekken");
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-6.php**
(in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-6.php. <http://localhost/Voorbeeld3-6.php>, dan zie je het volgende venster:



Voorbeeld 3-7

Een nog andere voorbeeld is het volgende; we maken van 2 getallen een kommagetal als **string** (tekst) en laten **vermenigvuldigen**:

1. Open het programma Crimson Editor
2. Typ in.

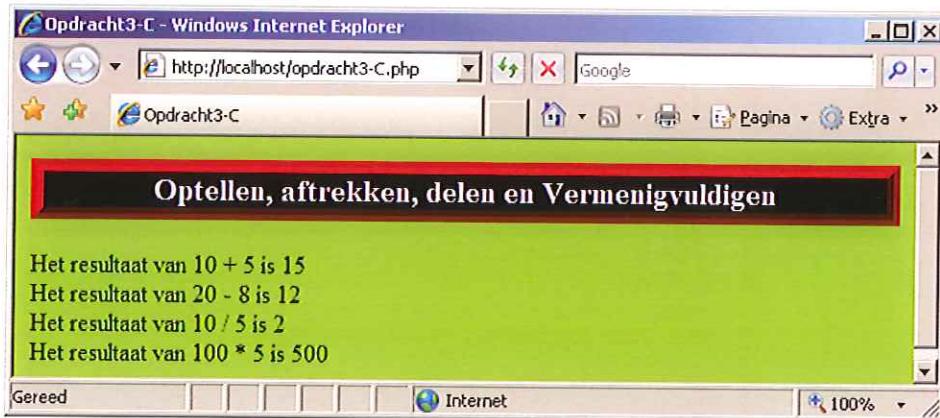
```
<html>
<head>
<title>voorbeeld3-7</title>
<LINK rel=STYLESHHEET href="opmaak.css" Type="text/css">
</head>
<body>
<?php
echo "<h1> Het product (X) van</h1>";
$getal1="7";
$getal2="5";
$kommagetal=$getal1. ".$getal2;
$resultaat=$kommagetal*3;
echo ($getal1. "." . $getal2." X 3 = ".$resultaat."<br>");
echo ("$getal1.$getal2 X 3 = $resultaat");
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-7.php**
(in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-7.php. <http://localhost/Voorbeeld3-7.php>, dan zie je het volgende venster:



Opdracht 3-C

- Maak de onderstaande webpagina zo nauwkeurig mogelijk na. Gebruik variabelen. Laat PHP de berekeningen uitvoeren.



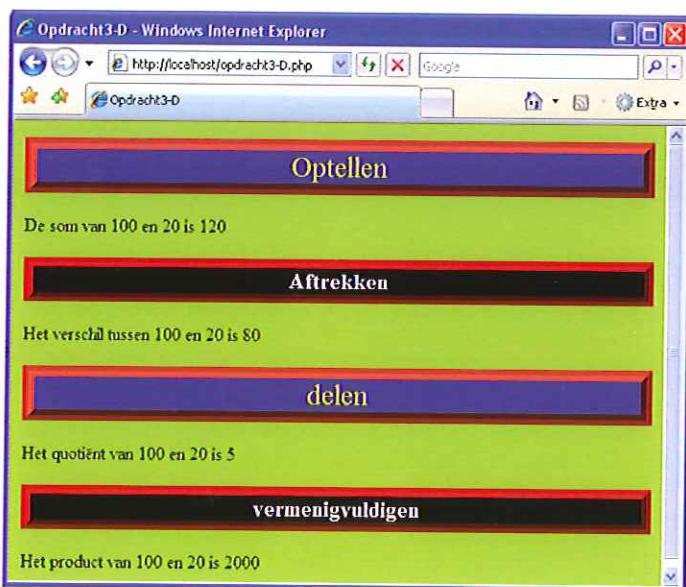
- Sla de pagina op, onder de naam: **Opdracht3-C.php**

- Wat moet je voor de opdracht inleveren:

- Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
- Kopieer de broncode van de opdracht naar dit Worddocument.
- Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
- Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

Opdracht3-D

- Maak de onderstaande webpagina zo nauwkeurig mogelijk na. Gebruik variabelen. Laat PHP de berekeningen uitvoeren.



2. Sla de pagina op, onder de naam: **Opdracht 3-D.php**
3. Wat moet je voor de opdracht inleveren:
 - Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

3.6 Afronden

Je kunt afronden naar beneden, naar boven en naar het dichtbijstaande getal:

floor() rondt af naar beneden,

ceil() rondt af naar boven.

Wil je afronden op een aantal decimalen, dan gebruik je:

round(Getal, AantalDecimelen)

Voorbeeld 3-8: Afronden

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>Voorbeeld3-8</title>
<LINK rel=STYLESHEET href= "opmaak.css" Type="text/css">
</head>
<body>
<H1>Afronden</H1>
<div id=tekstblok>
    <?php
        echo "Het getal <b>1.9</b> naar beneden afronden:<br>";
        $beneden=floor(1.9);
        echo "<b>het resultaat is $beneden</b><br>";

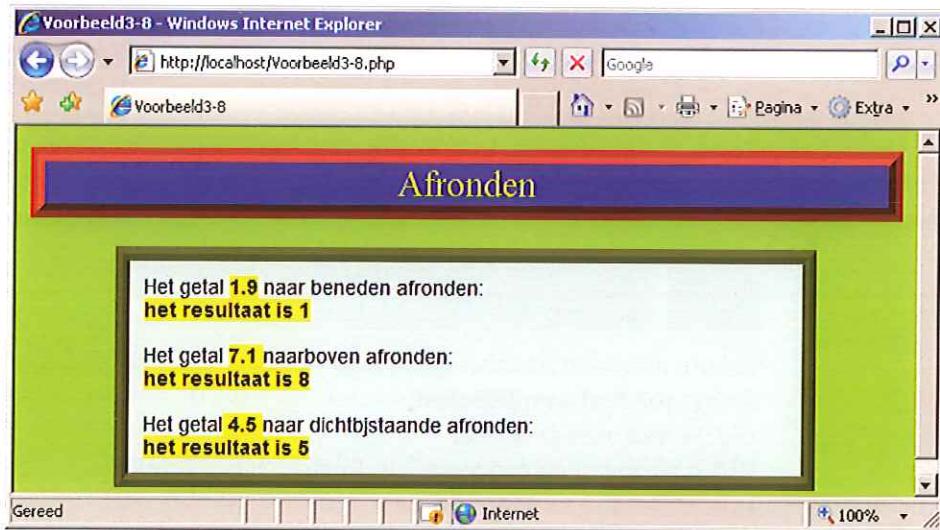
        echo "<br>";
        echo "Het getal <b>7.1 </b>naarboven afronden:<br>";
        $boven=ceil (7.1);
        echo "<b>het resultaat is $boven</b><br>";

        echo "<br>";
        echo "Het getal<b> 4.5 </b>naar dichtbijstaande afronden:<br>";
        $twijfel=round(4.5);
        echo "<b>het resultaat is $twijfel</b>";

    ?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-8.php**
(in de map C:\wamp\www)

4. Open het bestand Voorbeeld3-8.php. <http://localhost/Voorbeeld3-8.php>, dan zie je het volgende venster:



Opdracht 3-E: Gebruik van variabelen

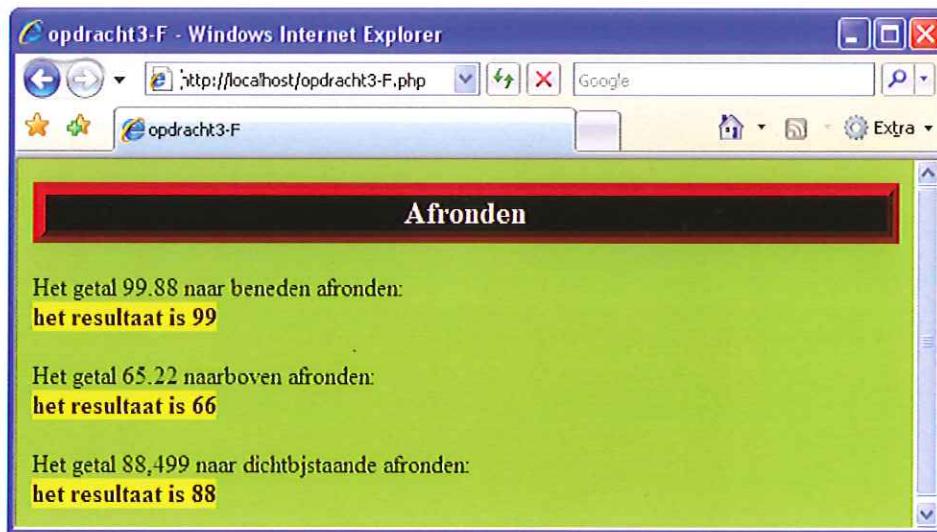
1. Maak de onderstaande webpagina zo nauwkeurig mogelijk na. Laat PHP de gemiddelde leeftijd berekenen.



2. Sla de pagina op, onder de naam: **opdracht3-E.php**
3. Wat moet je voor de opdracht inleveren:
- Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

Opdracht3-F: Afronden

- Maak de onderstaande webpagina zo nauwkeurig mogelijk na. Laat PHP het afronden uitvoeren.



- Sla de pagina op onder de naam: **Opdracht3-F.php**
- Wat moet je voor de opdracht inleveren:
 - Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

3.7 Kwadraat, wortel(sqrt) en machten(pow)

Voorbeeld 3-9 (Kwadraat, wortel en machten)

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>voorbeeld3-9</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<?php
$getal1= 9;
$getal2= 10000;
$getal3= 5;
$Kwadraat=$getal1*$getal1;
$wortel= sqrt($getal2);
$macht=pow($getal3,6);
Echo "<H1>Kwadraat, wortel en machten</h1><br>";
echo "<H2> De Kwadraat van 9 is $Kwadraat</H2>";
echo "<H2> De wortel van 10000 is $wortel</h2>";
echo "<h2> 5 tot de macht 6 is $macht</h2>";
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-9.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-9.php. <http://localhost/Voorbeeld3-9.php>, dan zie je het volgende venster:



Opdracht3-G

1. Bedenk een opdracht over deze onderwerp (**De rekenkundige operatoren**) gebruik alle mogelijkheden .
2. Sla deze opdracht op onder de naam **Opdracht3-G.php**.
3. Wat moet je voor de opdracht inleveren:
 - Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw eindopdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

3.8 Een array maken

Een **array** is eigenlijk geen type, maar een variabele die een rij waarden kan bevatten.

Voorbeeld:

```
$Boek[0] = "MS Word";
$Boek [1] = "MS Excel";
$Boek [2] = "MS Access";
$Boek[3] = "MS PowerPoint";
$Boek[4] = "Windows XP";
```

Hierbij geef je met de index (bv. 0, 1, 3 of 4) aan naar welk element in het array je wijst. Dat kun je als volgt voorstellen:

Naam v/d Variabele	Element 0	Element 1	Element 2	Element 3	Element 4
	MS Word	MS Excel	MS Access	MS Power-Point	Windows XP

Je kunt op verschillende manieren waarden toekennen aan een array:

- De eerste manier:

```
$b[0] = "Gareb";
$b[1] = "Hans";
$b[2] = "Jan";
```

\$b[0] = "Gareb"

- De tweede manier:

```
$b = array ( "Gareb" , "Hans" , "Jan");
```

Nog steeds geldt dan **\$b[0] = "Gareb"** enz.

- De derde manier:

```
$b[ ] = "Gareb";
$b[ ] = "Hans";
$b[ ] = "Jan";
```

Ook nu geldt `$b[0] = "Gareb"` enz.

Als je een nieuw element, zonder index, toevoegt aan de rij, dan wordt het achteraan de rij toegevoegd.

- De vierde manier:

```
$b = array ("5"=>"Gareb", "6"=> "Hans", "7"=> "Jan");
```

Dan geldt `$b["5"] = "Gareb"` enz.

- De vijfde manier:

```
$b = "Gareb,Hans,Jan"; $b = explode( , $namen);
```

Dan geldt `$b[0] = "Gareb"` enz.

Voorbeeld 3-10 De eerste manier

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>Voorbeeld3-10 </title>
<LINK rel=STYLESHHEET href="opmaak.css" Type="text/css">
</head>
<body>
<h2>Arrays</h2>
<?php
$Boek[0] = "MS Word";
$Boek [1] = "MS Excel";
$Boek [2] = "MS Access";
$Boek[3] = "MS PowerPoint";
$Boek[4] = "Windows XP";
echo "<b>Mijn favoriete boeken zijn: $Boek[3] en $Boek[1]</b>";
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-10.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-10.php. <http://localhost/Voorbeeld3-10.php> , dan zie je het volgende venster:



Voorbeeld 3-11 De tweede manier

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>voorbeeld3-11</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<?php
Echo "<h1> Array waarde 1=</h1>";
Echo "A-straat 9, Adm. de Ruyterlaan 2, Kobaltstraat 21";
$adres= array("A-straat 9", "Adm. de Ruyterlaan 2", "Kobaltstraat 21");
Echo "<br>";
echo "Array waarde 1=";
echo $adres[0];
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-11.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-11.php. http://localhost/Voorbeeld3-11.php , dan zie je het volgende venster:



Voorbeeld 3-12 De derde manier

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>voorbeeld3-12</title>
<LINK rel=STYLESHEET href= "opmaak.css" Type= "text/css">
</head>
<body>
<?php
echo "<h1> Array waarde 2=</h1>";
echo "De adressen: (A-straat 9, Adm. de Ruyterlaan 2, Kobaltstraat 21)";
$adres=array();
$adres[]= "A-straat 9";
$adres[]= "Adm. de Ruyterlaan 2";
$adres[]= "Kobaltstraat 21";
echo "<br>";
echo "Array waarde 2=";
echo $adres[1];
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-12.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-12.php. http://localhost/Voorbeeld3-12.php , dan zie je het volgende venster:



Voorbeeld 3-13 (Meer dimensionele arrays)

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>Voorbeeld3-13</title>
<LINK rel=STYLESHHEET href="opmaak.css" Type="text/css">
</head>
<body>
<h2>Meer dimensionele arrays</h2>
<?php
$Boek["Uitgeverij X"][] = "C";
$Boek["Uitgeverij X"][] = "C++";
$Boek["Uitgeverij X"][] = "PHP";
$Boek["Uitgeverij X"][] = "ECDL";
$Boek["Uitgeverij X"][] = "Java";
$Boek["Van buurtboek"][] = "C";
$Boek["Van buurtboek"][] = "C++";
$Boek["Van buurtboek"][] = "Perl";
$Boek["Van buurtboek"][] = "PHP";
$Boek["Van buurtboek"][] = "ECDL";
echo $Boek["Uitgeverij X"][2] . "<br>";
echo $Boek["Van buurtboek"][4] . "<br>";
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-13.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-13.php. http://localhost/Voorbeeld3-13.php , dan zie je het volgende venster:



3.9 Count (Het aantal elementen)

functie count()

Wil je nu weten hoeveel elementen een array bevat, dan heb je de functie **count()** ter beschikking.

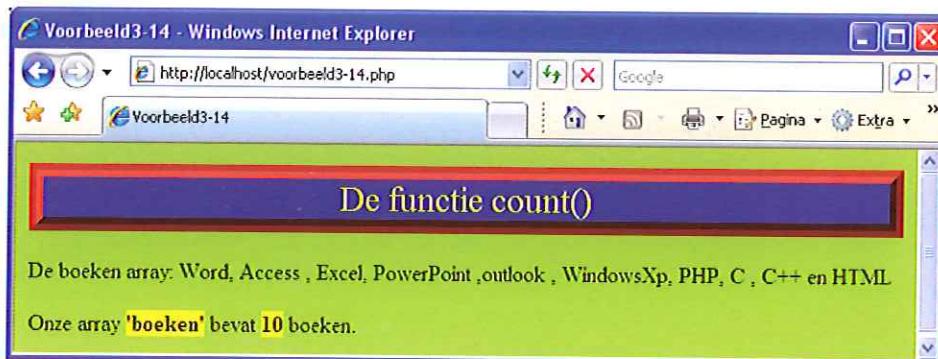
Met **count(\$b)** wordt in dit geval bedoeld: het aantal elementen van die rij.

Voorbeeld 3-14: count()

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>Voorbeeld3-14</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<?php
echo "<h1>De functie count()</h1>";
$boeken=array( "Word" , "Access" , "Excel" , "PowerPoint" , "outlook" , "WindowsXp" ,
"PHP" , "C" , "C++" , "HTML" );
$aalant_elementen=count($boeken);
echo "De boeken array: Word, Access , Excel, PowerPoint,outlook , WindowsXp, PHP,
C, C++ en HTML<br><br>";
echo "Onze array <b>'boeken'</b> bevat <b>$aalant_elementen</b> boeken." ;
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-14.php** (in de map C:\wamp\www) \
4. Open het bestand Voorbeeld3-14.php. http://localhost/Voorbeeld3-14.php , dan zie je het volgende venster:



opdracht 3-H

- Maak de onderstaande webpagina zo nauwkeurig mogelijk na.



- Sla de pagina op, onder de naam: **opdracht 3-H.php**
- Wat moet je voor de opdracht inleveren:
 - Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

Voorbeeld 3-15

Array_count_values (array):

Dit telt alle waarden in een array.

print_r() geeft informatie over een variabele weer op een manier dat mensen het kunnen leven. Als een **string, integer of float** wordt gegeven, zal de waarde ervan worden geprint. Als je een **array** meegeeft, zullen de waarden worden geprint in een opmaak met daarin sleutels en waarden.

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>Voorbeeld3-15 </title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<h1> Array_count_values (array)</h1>
<?php
$array = array (7, "Jan", 7, "Jan", "Gareb", "Gareb",7, "Gareb", "Gareb", "Gareb",);

echo "array: 7, Jan, 7, Jan, Gareb, Gareb,7, Gareb, Gareb, Gareb<br><br>";
echo "<h2>Dat Code telt alle waarden in een array</h2>";
print_r(array_count_values ($array));
?>
```

3. Sla de tekst op onder de naam **Voorbeeld3-15.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-15.php. http://localhost/Voorbeeld3-15.php , dan zie je het volgende venster:



Opdracht3-I

1. Bedenk een opdracht over dit onderwerp(**Array**) gebruik alle mogelijkheden
2. Sla deze opdracht op onder de naam **Opdracht 3-I.php**.
3. Wat moet je voor de opdracht inleveren:
 - Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

3.10 **Implode en explode**

Met **implode** kan je alle elementen van een array aan elkaar plakken. Tussen elk element wordt een koppelteken (\$koppelteken) gezet. Het geheel van array-elementen en tussengevoegde koppeltekens wordt door de functie als string geretourneerd.

Implode() heeft een koppelteken (\$koppelteken, een string) en een array (\$array) als invoer, en retourneert als een string.

Voorbeeld 3-16

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>Voorbeeld3-16</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<H1>Implode() </H1>
<?php
$array = array('voetballen', 'basketballen', 'computeren', 'zwemmen en fietsen');
$string = implode (", ", $array);
echo "Mijn hobby's zijn:". $string;
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-16.php** (in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-16.php. <http://localhost/Voorbeeld3-16.php> , dan zie je het volgende venster:



```
$array = explode($koppelteken, $string);
```

Explode() is het tegenovergestelde van **implode()**. Het deelt een string (\$string) op in stukken op grond van een koppelteken (\$koppelteken) - de verschillende delen worden opgeslagen in een array en deze array wordt door de functie getourneerd (\$array).

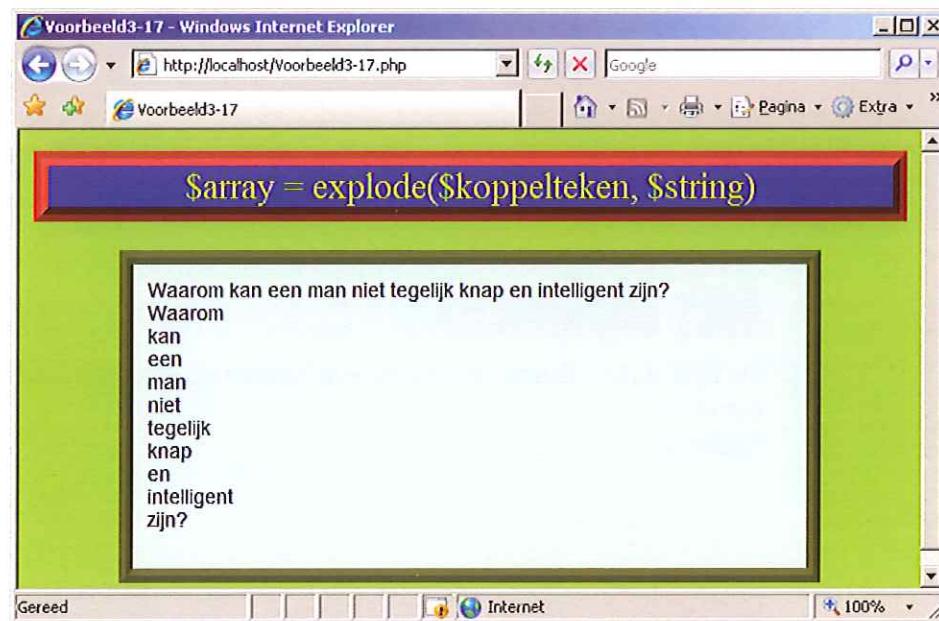
Explode() heeft twee strings als invoer (\$koppelteken en \$string) en retourneert een array.

Voorbeeld 3-17

1. Open het programma Crimson Editor
2. Typ in.

```
<html>
<head>
<title>Voorbeeld3-17</title>
<LINK rel=STYLESHEET href="opmaak.css" Type="text/css">
</head>
<body>
<H1>$array = explode($koppelteken, $string) </H1>
<div id=tekstblok>
<?php
echo "Waarom kan een man niet tegelijk knap en intelligent zijn?";
echo "<br>";
$string = "Waarom kan een man niet tegelijk knap en intelligent zijn?";
$array = explode(" ", $string);
echo "$array[0]<br>";
echo "$array[1]<br>";
echo "$array[2]<br>";
echo "$array[3]<br>";
echo "$array[4]<br>";
echo "$array[5]<br>";
echo "$array[6]<br>";
echo "$array[7]<br>";
echo "$array[8]<br>";
echo "$array[9]<br>";
echo "$array[10]<br>";
?>
</body>
</html>
```

3. Sla de tekst op onder de naam **Voorbeeld3-17.php**
(in de map C:\wamp\www)
4. Open het bestand Voorbeeld3-17.php. <http://localhost/Voorbeeld3-17.php>,
dan zie je het volgende venster:



Opdracht 3-J

1. Bedenk een opdracht over deze onderwerp([Array](#), [Explode\(\)](#) en [implode\(\)](#))
gebruik alle mogelijkheden .
2. Sla deze opdracht op onder de naam **Opdracht 3-J.php**
3. Wat moet je voor de opdracht inleveren:
 - Lever deze opdracht in als Worddocument, want voor jouw docent zijn de codes in een Worddocument gemakkelijker na te kijken.
 - Kopieer de broncode van de opdracht naar dit Worddocument.
 - Maak een Print Screen van jouw opdracht en plak deze onder de broncode in het Worddocument.
 - Vraag jouw docent op welke manier (e-mail of print) je de opdracht moet inleveren.

4 TIJD EN DATUM

4.1 Datumfunctie

De PHP **date()** functie maakt van een timestamp een beter leesbare tijd en datum.

Syntaxis:

date(format,timestamp)

Parameter	Omschrijving
format	Verplicht. Specificeert het formaat van de timestamp
timestamp	Optioneel. Specificeert een timestamp. Standaard. De huidige tijd en datum.(als timestamp)

PHP Date - Wat is een Timestamp?

Een timestamp is het aantal seconden sinds 1 Januari 1970 om 00:00:00 GMT.
Dit is ook bekend als de Unix Timestamp.

PHP Date – Datum weergave

De eerste parameter in de **date()** functie specificeert hoe de **datum/tijd** moet worden weergegeven. Het gebruikt letters om tijd en datum formaten weer te geven. Hier zijn enkele letters die gebruikt kunnen worden:

d – De dag van de maand 2 cijfers met nullen aan het begin; dus "01" tot "31"

m – De huidige maand, als getal dus "01" tot "12"

Y – Het huidige jaar in vier getallen (bijvoorbeeld: 2008)