

# PHP1

python reformatterCanvas.py 18661 56422 <https://www.roc.ovh/books/2024/page/php1>

## Strings deel 1

### *Wat ga je leren?*

Variabelen heb je in alle soorten en maten. De twee belangrijkste zijn getallen en strings. In deze les leer je wat een string variabele is en hoe je er in PHP mee kan werken.

We hebben al kennis gemaakt met strings in de vorige les over variabelen, maar in deze les gaan we hier nog wat verder op in.

Je hebt verschillende soorten variabelen.

Om te beginnen zijn er twee hoofdsoorten: **strings en getallen**. In deze les gaan we kijken naar strings.

## Strings

Strings (letterlijk vertaald een ketting) is een ketting van letters en/of getallen achter elkaar. Om het begin en een eind van een string aan te geven staat een string altijd tussen quotes. Je mag enkele of dubbele quotes gebruiken.

```
<?php

$string = "dit is een voorbeeld";
$nogEenString = 'dit is ook een string';

echo $string;
echo $nogEenString;

?>
```

Op regel 3 staat een string tussen dubbele quotes en op regel 4 staat een string tussen enkele quotes '.

Met strings kan je verschillende dingen doen. Zo kan je strings aan elkaar plakken. Dat heet [\*concatenation\*](#)

[\*Links to an external site.\*](#) (Engels).

In PHP plak je strings aan elkaar door tussen twee string een punt te zetten.

```
<?php
$voornaam=" Alexandra";
$sachternaam=" Gaona";

echo $voornaam." ".$sachternaam;
?>
```

Op regel 5 worden eigenlijk **drie** strings aan elkaar geplakt.

1. De string die in de variabele **\$voornaam** staat,
2. dan de string " " (deze staat niet in een variabele),
3. en dan string die in de variabele **\$sachternaam** staat.

Als je de HTML code <br> afdruckt dan wordt er een regel overgeslagen. "<br>" is een string.

```
echo "<br>"; // regel overslaan
```

Dus regel 1 drukt de string "<br>" af en de browser slaat dan een regel over.

De // aan het einde van de regel is overigens **commentaar**. Het doet niets en het heeft alleen als doel om je code duidelijker te maken (voor anderen of later voor jezelf).

## Opdracht

```
01 <?php
02 $voornaam=" Alexandra";
03 $sachternaam=" Gaona";
04
05 echo $voornaam." ".$sachternaam;
06 echo "Alexandra"." ".$sachternaam;
07 ?>
```

**Test de code.**

Regel 5 en 6 doen beide hetzelfde, althans hetzelfde wordt afgedrukt. Toch is er verschil. Hoe moet je de code bijvoorbeeld aanpassen, zodat de voornaam 'Ali' wordt?

Leg in eigen woorden uit wat het verschil is tussen regel 5 en regel 6.

Tip1: verander regel 2 en geef \$voornaam een andere waarde, bijvoorbeeld jouw voornaam. Wat gebeurt er? Zie je nu het verschil?

Tip2: denk aan wat een variabele is, en waar wordt die gebruikt?

Je kunt als template het volgende gebruiken.

```
Op regel 5 wordt ..... afgedrukt omdat .....  
Op regel 6 wordt ..... afgedrukt omdat .....
```

## Inleveren

1. jouw uitleg (in text veld)

# Strings deel 2

## Opdracht

Maak een nieuw bestand php05b-jouw-naam.php

```
<?php  
$voornaam="Alexandra";  
$achternaam="Gaona";  
echo $voornaam. " ". $achternaam;  
echo "Alexandra ". $achternaam;  
?>
```

Gebruik dit laatste voorbeeld en verander de voor- en achternaam in je eigen naam en verander de code zodat het volgende wordt afgedrukt.

```
Welkom jouw-voornaam jouw achternaam!  
  
Ik ben blij om je te zien!
```

Gebruik de variabelen \$voornaam en \$achternaam. Geef de variabelen de juiste waarden. Dus \$voornaam wordt jouw voornaam en \$achternaam krijgt jouw achternaam.

Maak gebruik van string concatenation (aan elkaar plakken van strings).

De output, de tekst in je browser bestaat uit 3 regels. Gebruik de string "<br>" om een regel over slaan. `<br>`

Test je code uit!

## Inleveren

1. screenshot waarin je laat zien dat je code werkt.
2. jouw code php05b-jouw-naam.php.

# Strings 3

## Opdracht

Bedenk nu zelf 2 variabelen, geef deze een waarde en maak een juiste Nederlandse zin met deze twee variabelen.

Voorbeeld:

```
$huisdier="hond";  
$kleur="zwart";  
  
echo "Mijn ".$huisdier." is ".$kleur;
```

Dus bedenk een **eigen** voorbeeld met eigen variabelen.

Dit zijn de regels:

1. Bedenk minimaal 2 eigen *logische* variabele namen.
2. Geef deze variabele een *logische* waarde.
3. Druk een zin af. Gebruik string concatenation (aan elkaar plakken van strings) en druk een juiste Nederlandse zin af.

## Inleveren

1. screenshot waarin je laat zien dat je code werkt.
2. jouw code php05c-jouw-naam.php.

# Getallen 1

## *Wat ga je leren?*

Een veel gebruikt type variabele is integer. In deze les je wat een integer variabele is en hoe je daar mee kan werken in PHP.

Naast strings hebben we ook getallen. Deze zijn weer onderverdeeld in gehele getallen (integers) en gebroken getallen (float). Dat verschil is in PHP niet zo belangrijk omdat PHP zelf het juiste soort kiest.

Getallen zijn wel anders dan strings want je kunt er mee rekenen. Om aan te geven dat je een getal bedoeld en geen string, zet je een getal *niet* tussen quotes.

Dus `$voorbeeld="9"` is de string "9" en `$voorbeeld=9` is het getal 9.

```
<?php

$getal1=12;
$getal2=13;
echo $getal1 + $getal2;

?>
```

Net zoals je getallen kan optellen, kun je ook andere bewerkingen uitvoeren.

Bewerking	teken	voorbeeld
optellen	+	echo \$a + \$b;
afrekken	-	echo \$a - \$b;
vermenigvuldigen (keer)	*	echo \$a * \$b;
delen	/	echo \$a / \$b;

Ga door naar de opdrachten.

## Opdracht

Test de volgende code.

```
01 $a=20;  
02 $b=8;  
03  
04 echo "$a - $b";  
05 echo $a - $b;
```

Beschrijf wat er wordt afgedrukt op regel 4 en wat er wordt afgedrukt op regel 5.

Leg in jouw eigen woorden uit waarom er op regel 4 wat anders wordt afgedrukt als op regel 5.

Kom je er niet uit? Maak dan de volgende twee zinnen af.

Op regel 4 wordt ..... afgedrukt omdat .....

Op regel 5 wordt ..... afgedrukt omdat .....

(hint: denk aan variabele en strings)

## Inleveren

Uitleg in online text

# Getallen 2

## Opdracht

Maak het bestand php06-jouw-naam.php.

Maak de variabele \$mijnLeeftijd en geef de variabele de waarde die overeenkomt met je leeftijd.

Maak een variabele \$vorigJaar en bereken met de variabele \$mijnLeeftijd hoe oud je vorige jaar was.

Maak een variabele \$overZesJaar en bereken met de variabele \$mijnLeeftijd hoe oud je over zes jaar bent.

Druk de uitkomsten af met de volgende code

```
// maak hier je eigen code  
//
```

```
//
```

```
echo "Ik ben $mijnLeeftijd jaar";  
echo "<br>";  
echo "Vorig jaar was ik $vorigJaar oud";  
echo "<br>";  
echo "Over 6 jaar ben ik $overZesJaar oud";
```

## *Test je code!*

## Inleveren (2 bestanden)

\* php06-jouw-naam.php

\* schermafdruck van je hele browser scherm, URL is zichtbaar, met output van je code

## Conditions 1

### *Wat ga je leren?*

Code wordt regel voor regel van boven naar beneden uitgevoerd,

Je hebt echter ook code die alleen maar wordt uitgevoerd als er aan een voorwaarde (of conditie) is voldaan. Hiervoor heb je conditional statements in PHP, *if* (in het Nederlands: *als*).

In deze les leer je hoe je met een *if* in PHP kan werken en hoe je dus kan zorgen dat code alleen wordt uitgevoerd als er aan die ene voorwaarde wordt voldaan.

## IF

Stel je maakt een spel en je moet bepalen of de speler het juiste antwoord heeft gegeven:

- Juiste antwoord? Score wordt verhoogd met 1.
- Onjuiste antwoord? Score blijft hetzelfde.

Er komt dus alleen iets bij de score als er aan een voorwaarde is voldaan.

Dat ziet er in PHP zo uit:

```

$score=0;

$vraag = "Hoeveel is 10+2?";
$antwoord = 12;

if ( $antwoord == 12 ) {
    $score = $score + 1;
}

echo "De score is: $score";

```

Op regel 6 wordt het antwoord gecontroleerd. Is het antwoord juist dan wordt alle code tussen { en } uitgevoerd. In dit geval is dat alleen regel 7.

Op regel 4 zie je = en op regel 6 ==

Dat komt omdat ze beide iets anders betekenen.

teken	voorbeeld	betekent
=	\$antwoord = 12	De waarde van \$antwoord <b>wordt</b> 12
==	\$antwoord == 12	Is de waarde van \$antwoord <b>gelijk aan</b> 12?

We kunnen ons afvragen of waarden gelijk zijn, maar er zijn nog een heel stel andere voorwaarden. De belangrijkste zijn:

teken	voorbeeld	betekent
==	\$antwoord == 12	Is de waarde van \$antwoord <b>gelijk aan</b> 12?
!=	\$antwoord != 12	Is de waarde van \$antwoord <b>ongelijk aan</b> 12?
>	\$antwoord > 12	Is de waarde van \$antwoord <b>groter dan</b> 12?
<	\$antwoord < 12	Is de waarde van \$antwoord <b>kleiner dan</b> 12?
>=	\$antwoord >= 12	Is de waarde van \$antwoord <b>groter of gelijk aan</b> 12?
<=	\$antwoord <= 12	Is de waarde van \$antwoord <b>kleiner of gelijk aan</b> 12?

**ELSE**



Stel voor dat als de score fout is dat er dan een punt van de score wordt afgetrokken, dan zou je dat zo kunnen coderen

```
$score=0;

$vraag = "Hoeveel is 10+2?";
$antwoord = 12;

if ( $antwoord == 12 ) {
    $score = $score + 1;
} else {
    $score = $score - 1;
}

echo "De score is: $score";
```

Maak de volgende opdrachten en gebruik indien nodig de volgende bronnen.

## Opdracht

Maak een nieuw bestand en noem dat php107a-jouw-naam.php

Maak de onderstaande code af op regel 4 (vervang de puntjes).

Plaats daar een vergelijking: controleer of de waarde van \$variabele1 en \$variabele2 gelijk zijn.

```
$variabele1 = 10;
$variabele2 = 10;

if(.....) {
    echo "De twee waarden zijn gelijk";
}
```

## Inleveren

1. php07a-jouw-naam.php

# Bronnen

[Jaap vd Veen - statements](#)

[W3 Schools - operators](#)

[W3 Schools - if else](#)

## Conditions 2

### Opdracht

Maak een nieuw bestand en noem dat php107b-jouw-naam.php

Maak de onderstaande code af door de puntjes op regel 4 te vervangen door een vergelijking te maken. Controleer of de waarde van \$variabele1 en \$variabele2 ongelijk zijn.

```
$variabele1 = 10;
$variabele2 = 11;

if(.....) {
    echo "De twee waarden zijn ongelijk";
}
```

### Inleveren

1. php07b-jouw-naam.php

# Bronnen

[Jaap vd Veen - statements](#)

[W3 Schools - operators](#)

[W3 Schools - if else](#)

# Conditions 3

## Opdracht

Maak een nieuw bestand en noem dat php107c-jouw-naam.php

Maak de onderstaande code af door de puntjes op regel 4, 6 en 8 te vervangen. Maak een if-then-else constructie en zorg ervoor dat de juiste tekst op het scherm wordt weergegeven.

```
$variabele1 = 10;
$variabele2 = 10;

....
    echo "De twee waarden zijn gelijk";
....
    echo "De twee waarden zijn ongelijk";
....
```

## Inleveren

1. php107c-jouw-naam.php

## Bronnen

[Jaap vd Veen - statements](#)

[W3 Schools - operators](#)

[W3 Schools - if else](#)

## Include

# HTML, PHP en include

# Wat ga je leren?

In deze les leer je hoe je HTML en PHP samen kunt gebruiken.

Je leert ook het wat het *include* en *require* in PHP doet en hoe je dat kunt gebruiken.

## Include

Om je code overzichtelijk te houden, stop je niet al je code in één groot bestand, maar je deelt het op in meerdere kleinere bestanden. Je kunt dan snel de juiste code vinden.

Zo kun je php code hebben die een footer afdruckt. Stel je wilt je volgende footer:

```
<?php
echo "<hr>";
echo "(c) Copyright<br>";
echo "De inhoud en structuur van onze webpagina's zijn auteursrechtelijk beschermd.";
?>
```

Deze footer wil je in je pagina afdrukken dan kun je deze pagina 'includen'. Dat gaat als volgt:

```
<!DOCTYPE html>
<html>
<body>
<h1>My First Heading</h1>
<p>My first paragraph. </p>
</body>

<?php
include 'footer.php';
?>

</html>
```

Zie je wat er op regel 9 gebeurt? Je voegt als het ware de inhoud van de footer.php toe op regel 9. Hieronder zie je dan wat de include eigenlijk doet.

```
<!DOCTYPE html>
<html>
<body>
```

```
<h1>My First Heading</h1>
<p>My first paragraph. </p>
</body>

<?php
echo "<hr>";
echo "(c) copyright<br>";
echo "De inhoud en structuur van onze webpagina's zijn auteursrechtelijk beschermd.";
?>

</html>
```

De include verwijst altijd naar een bestand.

Let erop dat je alleen php code kan uitvoeren als de naam van het bestand op .php eindigt. Bovenstaand voorbeeld werkt dus alleen als de naam eindigt in .php.

Je kunt de naam van het bestand ook als string mee geven, tussen " zoals in het voorbeeld, maar je kunt natuurlijk ook een variabele meegeven:

```
include $footer;
```

De code van het bestand waarvan de naam in de variabele \$footer staat wordt nu ingevoegd.

## Nog een voorbeeld

Voorbeeld: **home.php**

```
<?php
    include 'database.php';
?>
<html>
<head>
</head>
<body>
    <div class="menu">
        <?php include 'menu.php'; ?>
    </div>
</body>
</html>
```

Je ziet in bovenstaand voorbeeld op regel 9 dat een menu met behulp van php wordt ingevoegd in de

```
<div class="menu"> .... </div>
```

Je ziet dat php geopend wordt `<?php` en er dan normale php-code volgt. Zodra de php code klaar is dan sluiten we de php code weer met `?>`.

## Opdracht

Maak een eigen footer.php en zet daarin een eigen footer. Noem het bestand php108a-footer-jouw-naam.php

Maak een eenvoudige webpagina php8a-jouw-naam.php

Maak een titel `<h2>` *Mijn Webpagina* .

Zet daaronder *welkom <jouw naam>*

Gebruik tenminste een `<body>` en `<html>` tag

[image-1654093629221.png](#)

Image not found or type unknown

Geef de variabele \$footer de naam van het bestand waarin jouw footer staat en include de footer door gebruik te maken van de variabele \$footer en het php statement include.

Test of je code werkt voordat je het inlevert!

## Inleveren

- php8a-jouw-naam.php
- php108a-footer-jouw-naam.php

## Bronnen

[W3schools Include](#)

# Logische operators 1

## Wat ga je leren?

In deze les gaan we leren hoe we condities kunnen combineren.

We leren hoe we code alleen kunnen laten uitvoeren als er aan **twee** voorwaarden wordt voldaan.

En we leren hoe we code kunnen uitvoeren als er aan **één van de twee** voorwaarden wordt voldaan.

We hebben het gehad over *if-then-else*.

Met een *if-then-else* kun je *voorwaardelijk* code uitvoeren. Alleen als een bepaalde vergelijking waar is dan wordt bepaalde code uitgevoerd.

Stel je hebt een variabele \$temperatuur waarin de temperatuur staat en stel je hebt een andere variabele waarin de dag van de week staat. Dus bijvoorbeeld:

```
$temperatuur = 21;  
$dagVanDeWeek = 'ma' ;
```

Dus in dit voorbeeld is het maandag en de temperatuur is 21 graden. Stel dat ik code moet schrijven die mij advies geeft over het dragen van een korte broek. Ik wil alleen een korte broek aan doen als het zaterdag is én als de temperatuur minimaal 21

is. Dat zijn dus twee voorwaarden die *waar* moeten zijn.

Hoe doe je dat in code?

```
$temperatuur = 21;  
$dagVanDeWeek = 'ma' ;  
  
if ( $temperatuur >= 21 && $dagVanDeWeek == "za" ) {  
    echo "Advies is om vandaag een korte broek aan te trekken.";   
} else {  
    echo "Advies is om vandaag geen korte broek aan te trekken.";   
}
```

Zie je het **&&** - teken?

Dat betekent dat **beide** voorwaarden waar moeten zijn. Dus de temperatuur is groter of gelijk aan 21 graden **én** de dag van de week is 'za'.

Stel dat we nu op zaterdag of zondag altijd een korte broek willen dragen, hoe zou de code er dan uit zien?

```

$temperatuur = 21;
$dagVanDeWeek = 'ma';

if ( $dagVanDeWeek == "za" || $dagVanDeWeek == "zo" ) {
    echo "Advies is om vandaag een korte broek aan te trekken.";
} else {
    echo "Advies is om vandaag geen korte broek aan te trekken.";
}

```

Zie je het `||` - teken?

Dat betekent dat aan **één van de** voorwaarden moet worden voldaan. Dus als de dag van de week zaterdag **óf** zondag is dan volgt het advies voor het dragen van een korte broek.

In de voorbeelden wordt `&&` gebruikt voor *and* en `||` voor *or*. In de meeste programmeertalen kun je ook gewoon het woord **and** en **or** gebruiken. Dat werkt in PHP ook.

Dus de volgende twee regels doen hetzelfde.

```

if ( $temperatuur >= 21 && $dagVanDeWeek == "za" ) {...

if ( $temperatuur >= 21 and $dagVanDeWeek == "za" ) {...

```

## Dus samengevat

		Betekenis	Voorbeeld
<code>&amp;&amp;</code>	and	AND, aan <b>beide</b> voorwaarden moet zijn voldaan	if ( \$temperatuur >= 21 && \$dagVanDeWeek == "za" ) {
<code>  </code>	or	OR, aan <b>één van beide</b> voorwaarden zijn voldaan	if ( \$dagVanDeWeek == "za"    \$dagVanDeWeek == "zo" ) {

## Haakjes in condities

Stel we willen alleen een korte broek aan als de temperatuur minimaal 21 graden is én de dag zaterdag is of de dag zondag is.

Dan krijgen we de volgende code:

```

if ( $temperatuur >= 21 && $dagVanDeWeek == "za" || $dagVanDeWeek == "zo" ) {

```



Maar stel dat het anders is. Op zondag willen we altijd een korte broek dragen en op zaterdag alleen als de temperatuur minimaal 21 graden is. Dus in code:

```
if ( $dagVanDeWeek == "zo" || $temperatuur >= 21 && $dagVanDeWeek == "za" ) {
```

Maar deze code is bijna hetzelfde als de vorige code, alleen de volgorde is anders. Om de code goed uit te voeren moeten we haakjes gebruiken:

Korte broek in het weekend als de temperatuur 21 graden of hoger is:

```
if ( $temperatuur >= 21 && ( $dagVanDeWeek == "za" || $dagVanDeWeek == "zo" ) ) {
```

Het geen tussen haakjes, de || wordt eerst uitgevoerd, daarna volgt de &&

Altijd een korte broek op zondag én op zaterdag alleen een korte broek als de temperatuur 21 graden of hoger is.

```
if ( ($temperatuur >= 21 && $dagVanDeWeek == "za" ) || $dagVanDeWeek == "zo" ) {
```

Hier wordt eerst de && uitgevoerd en daarna volgt de ||.

## Opdracht

Neem onderstaande code over:

```
<?php
$werkdagVanDeWeek=[ 'ma','di','wo','do','vr' ];
$vrjiedag=[ 'ja','nee' ];

echo "<table border=1>";
echo "<tr><th>dag</th><th>vrjiedag</th><th>advies</th></tr>";

foreach($werkdagVanDeWeek as $dag) {
    foreach($vrjiedag as $vd) {

        echo "<tr>";
        echo "<td>$dag</td><td>$vd</td><td>";

        // je moet vroeg opstaan op vrijdagen als het geen vrije dag is
        if ( $dag .. .... && $vd.. .... ) {
```

```
        echo "07:00 uur op staan";
    } else {
        echo "uitslapen";
    }

    echo "</td>";
}
}
echo "</table>";
```

Maak de conditie in het if-statement af door op de puntjes de code aan te vullen.

Je moet alleen vroeg opstaan op vrijdagen als het geen vrije dag is.

De output van de code moet er dus als volgt uit zien:

VdagOpdracht.PNG

Image not found or type unknown

## Inleveren (2 bestanden)

\* Screenshot van de output inclusief de URL

\* php01-conditions01-jouw-naam.php

# Logische operators 2

Neem onderstaande code over:

```
<?php
$dagVanDeWeek=[ 'ma','di','wo','do','vr','za','zo' ];
$vrijeDag=[ 'ja', 'nee' ];

echo "<table border=1>";
echo "<tr><th>dag</th><th>vrij dag</th><th>advies</th></tr>";

foreach($dagVanDeWeek as $dag) {
    foreach($vrijeDag as $vd) {
```

```

echo "<tr>";
echo "<td>$dag</td><td>$vd</td><td>";

// je moet vroeg opstaan op elke doordeweekse dag (maandag, dinsdag, woensdag, donderdag of
if ( .... .. ) {
    echo "06:30 uur op staan";
} else {
    echo "uitslapen";
}

echo "</td>";
}
}
echo "</table>";

```

Maak de conditie in het if-statement af door op de puntjes de code aan te vullen.

Je moet vroeg opstaan op elke doordeweekse dag (maandag, dinsdag, woensdag, donderdag of vrijdag).

De output van de code moet er dus als volgt uit zien:

Screenshot 2022-09-26 174318.png

Image not found or type unknown

## Inleveren

1. Screenshot van de output inclusief de URL.
2. Je code php01-conditions02-jouw-naam.php

## Logische operators 3

Neem onderstaande code over:

```

<?php
$dagVanDeWeek=[ 'ma','di','wo','do','vr','za','zo' ];
$vrjDag=[ 'ja','nee'];

echo "<table border=1>";
echo "<tr><th>dag</th><th>vrij dag</th><th>advies</th></tr>";

foreach($dagVanDeWeek as $dag) {
    foreach($vrjDag as $vd) {

```

```

echo "<tr>";
echo "<td>$dag</td><td>$vd</td><td>";

// je werkt op maandag en dinsdag; je moet dus vroeg opstaan op maandag of dinsdag als het g
if ( ..... ) {
    echo "06:30 uur op staan";
} else {
    echo "uitslapen";
}

echo "</td>";
}
}
echo "</table>";

```

Maak de conditie in het if-statement af door op de puntjes de code aan te vullen.

Let op dit is een lastige, controleer je antwoord goed!

Kom je er niet uit? Lees dan het stuk over " **Logische operatoren en korte broeken** " en dan vooral het stukje " **haakjes in condities** ".

Je moet alleen vroeg opstaan op maandagen als het geen vrije dag is.

De output van de code moet er dus als volgt uit zien:

Screenshot 2022-09-26 192055.png

Image not found or type unknown

Alleen op de maandag en dinsdag die geen vrije dagen zijn, moet je vroeg op staan.

## Inleveren

1. Screenshot van de output inclusief de URL.
2. Je code php01-conditions03-jouw-naam.php

# Eindopdracht coderen - deel 1

## *Wat ga je leren?*

Met alles wat we hebben geleerd gaan we een PHP programma maken.

Let's go!

Een webshop geeft zomerse korting op alle artikelen als dit weekend bepaalde zomerse temperaturen worden bereikt.

De korting bepaal je in de code aan de hand van de temperatuur.

10%: boven de 21 graden, maar niet boven de 25 graden  
15%: boven de 25 graden, maar niet boven de 30 graden  
20%: boven de 30 graden

## Bezorgkosten

De bezorgkosten zijn standaard 4 euro per bestelling, maar als de prijs van de bestelling (na zomerse korting) meer dan 40 euro is dan zijn de bezorgkosten 0.

```
<?php

// vul waarden in voor de volgende variabelen
$temperatuur=
$prijsBestelling= // bepaal de korting
...
...

// bepaal de $bezorgkosten
...
...

// druk resultaat af
```

```
echo "Bedrag bestelling: $prijsBestelling";  
echo "<br>";  
echo "Temperatuur dit weekend: $temperatuur";  
echo "<br>";  
echo "Korting: $korting";  
echo "<br>";  
echo "Bezorgkosten: $bezorgkosten";  
echo "<br>";  
echo "<br>";  
echo "Eindbedrag: ".$prijsBestelling-$korting+$bezorgkosten;
```

Maak de **code** af en zorgen ervoor dat het eindbedrag juist wordt **berekend** .

Bepaal eerst de \$bezorgkosten op regel 8, dan de korting op regel 12.

## Inleveren

- **code** php10-jouw-naam.php

# Eindopdracht testen - deel 2

## Testen

In de code die je in deel één hebt gemaakt kun je met verschillende waarden testen. Laten we bijvoorbeeld testen wat er uit komt als de temperatuur 20 is, en de prijs van de bestelling 40.

Vul testwaarden in op regel 4 en 5 en test de code. Wat komt er uit jouw code en klopt dat?

## Testlog

Maak een **testlog** . Dit is een kort documentje waarin je telkens als je test de testwaarden (gebruik handige waarden!) noteert en daarbij zet je het eindbedrag neer.

In het test documentje staat *bijvoorbeeld* het volgende:

temperatuur	prijsBestelling	Eindbedrag	Correct (ja/nee)
20	40	44	ja/nee
20	42	42	ja/nee

22	42	41,8	ja/nee
30	10	12,5	ja/nee
50	5	8	ja/nee

In de laatste kolom moet je ja of nee invullen afhankelijk van het resultaat. Bepaal **minimaal** 10 test scenario's. Je hebt er al 5 (hierboven), dus je moet er zelf nog minimaal 5 bij bedenken. Je moet dus 10 combinaties kiezen, uitrekenen wat er uit moet komen, opschrijven en dan testen of dat er ook uit komt.

## Inleveren

- **testlog** test10-jouw-naam.pdf (gebruik "save as" in MS Word en kies PDF als bestands type)

# PHP Eindtest

PHP Eindtest. Je krijgt 5 willekeurige vragen. Je krijgt 6 minuten de tijd om de quiz te maken.

Je mag deze test maximaal 6x doen.