PHP Intro

Update

python reformatterCanvas.py 18661 56420 https://www.roc.ovh/books/2024/page/php-intro

Wat is een webserver

Wat ga je leren?

In deze les leer je het verschil tussen een front-end taal en een back-end taal.

Je leert dat PHP een back-end taal is en dat je daarom een (web)server nodig hebt om PHP te kunnen gebruiken.

Lees de tekst goed, want in de volgende opdracht krijg je vragen over deze tekst.

Een webserver is een computer die verbonden is met het internet en waar bestanden op staan op st
nodig zijn om een website te laten zien.

Elke website die jij bezoekt op het world wide web staat ergens op een webserver. Zo'n webserver h
map (folder) met daarin webpagina's die bestaan uit HTML-pagina's, CSS-bestanden, JavaScript, PHI
en dergelijke.

Bestandssoorten	Functie
HTML	Basis opmaak van een webpagina
CSS	Detail opmaak van een webpagina
JavaScript	Interactie programmeren in de browser
PHP	Interactie programmeren op de server

Tot nu toe ben je gewend om de HTML-bestanden gewoon vanaf je eigen computer te starten door e
te klikken. De browser leest de bestanden en laat deze een mooie (of lelijke) pagina zien.

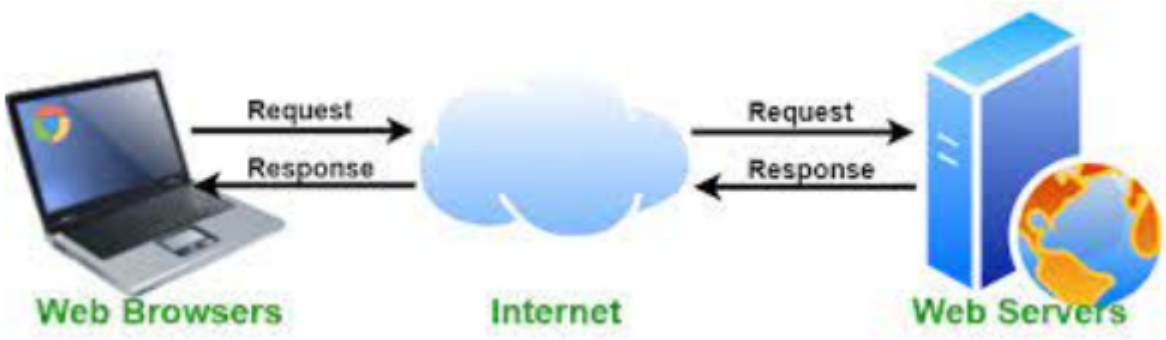
Je bent als gebruiker van webapplicaties gewend om bijvoorbeeld in te loggen, een bericht achter te
e-mail te sturen. Deze applicaties zijn *dynamisch.* Dynamische applicaties hebben een **input** (invo
deze om in een **output (uitvoer)** .

Bijvoorbeeld:

input : gebruikersnaam en wachtwoord
output : een boodschap dat het aanloggen is gelukt of een boodschap dat het aanloggen niet is ge

Om een dynamische webapplicatie te maken heb je meestal een webserver nodig die de pagina's **s** webserver krijgt van de browser, de cliënt een verzoek en de webserver behandeld het verzoek en s antwoord terug. Net als een ober die een drankje **serveert** .

Stel, je wilt een overzicht van alle studenten in een klas. Je vraagt via de browser dan om het overzi bijvoorbeeld door op een menu te klikken. Het verzoek gaat dan via internet naar een **webserver** het antwoord via internet weer **terug** naar jouw **browser** .



Omdat je nog geen toegang hebt tot een echte webserver gaan we er eentje simuleren. Simuleren i doen alsof'. We simuleren de webserver met een gratis applicatie (wij doen dat met XAMPP) die je k installeren op je eigen laptop. Dat heet dan een lokale server. Dat wordt ook wel **localhost** genoe is dus door het gebruik van XAMPP een webserver geworden.

Om een dynamische applicatie te maken gaan we in deze module gebruik maken van PHP.

De server heet ook wel back-end (achterkant). PHP is een back-end-taal. Dat betekent dat de taal d back-end, de server draait.

Een bezoeker van de site (de eindgebruiker) kan de code die op de server staat niet zien of wijzigen

Bestandssoorten	Functie	Waar
HTML	Basis opmaak van een webpagina	Front-end / browser
CSS	Detail opmaak van een webpagina	Front-end / browser
JavaScript	Interactie programmeren in de browser	Front-end / browser
PHP	Interactie programmeren op de server	Back-end / server

Database

Met PHP kun je ook een database benaderen waarin je gegevens kan opslaan voor een langere peri bijvoorbeeld de gegevens van een gebruiker zoals voornaam, achternaam, e-mail en wachtwoord.

Als je met PHP aan de slag gaat dan kun je pagina's dynamisch maken. Dit betekent dat de output a
van de input van de gebruiker.

XAMPP

Tijdens deze module maak je gebruik van een lokale server. We gebruiken hiervoor XAMPP.

Je kunt XAMPP downloaden en installeren als je dat nog niet hebt gedaan.

Weet je (nog) niet hoe je XAMPP moet installeren op jouw Windows machine of heb je een "port con

of heb je andere problemen, kijk dan eens of de info op deze pagina je kan helpen.

Opdracht

1. Installeer XAMPP op de manier zoals uitgelegd in de les.
2. Volg de instructies in de les en voer de kennischeck uit in de les, 8 vragen en minimale score is 80%

Inleveren

1. Volg de instructies in de les en lever een screenshot van je XAMPP in.
2. Lever een screenshot van de kennis-check die je tijdens de les hebt uitgevoerd in.

Welkom!

Je hebt XAMPP geïnstalleerd.

Voer de volgende stappen uit.

1. Start XAMPP.
2. In de folder C:\xampp\htdocs staat een bestand index.php. Verwijder dit bestand!
3. Maak een nieuwe folder aan en noem die *PHP1*
4. Met VSC open je de net aangemaakte folder.
5. In deze folder maak je met VCS een nieuw bestand en je noemt dit bestand opdracht1-jouw-naam.php.

- Vervang *jouw-naam* door jouw naam
  Overal waar in het vervolg j*ouw-naam* staat, vervang je de tekst door jouw naam.

6. Met VSC open je het net aangemaakte bestand.
7. Je maakt de volgende code.

```
<?php


echo "Hello world!";


?>
```

8. Ga nu naar de browser en type als url, *localhost* in.
9. Open nu de folder en het bestand dat je net hebt gemaakt, wat gebeurt er?

Als het goed is zie er output in de browser verschijnt:

Uitleg code

Wat gebeurt er in deze code?

- **Op regel 1 <?php**
  wordt tegen de webserver verteld dat we code in php gaan maken. De server/XAMPP weet dan welke taal jij gata praten.
- **Op regel 3, echo**
  Het woordje *echo* betekent dat de server output moet gaan maken. Deze output wordt naar de web browser gestuurd.
- **Op regel 3 "Hello world!"**
  echo moet iets naar de browser sturen, dan moet je natuurlijk wel vertellen wat dat is. Het zinnetje na de echo is dat wat er naar de browser wordt gestuurd. Dit zinnetje staat tussen "" om aan te geven waar het zinnetje begint en waar het eindigt.
- **Op regel 3 ;**
  Na ELK commando in PHP staat een ; om aan te geven dat het commando afgelopen is en. De meest gemaakte fout in PHP is dat je een ; vergeet. Wedden dat jij dat dit jaar ook tig keer gaat vergeten?
- **Op regel 5 ?>**
  We zijn klaar met onze commando's in PHP. We vertellen de server/XAMPP dat we stoppen met "PHP praten".

Test de code en controleer of er in je browser de tekst Hello World! verschijnt.

Opdracht

Verander de (php) code zodat de volgende output in de browser verschijnt:

```
Welkom jouw-Voornaam !
```

Vergeet niet jouw-Voornaam te vervangen door jouw voornaam.

Inleveren

Je moet twee bestanden inleveren:

1. Het bestand *opdracht1-jouw-naam.php*
2. Een schermafdruk van jouw hele browser scherm waarin dus ook de URL (web adres te zien is) te zien is en waarin je de output van jouw code laat zien.

Variabelen

Wat ga je leren?

In ***alle*** programmeertalen gebruik je variabelen. In deze les leer je wat een variabele is en hoe je deze kan gebruiken.

Computer geheugen

Een variabele is een plek in het geheugen van de computer waar je informatie in op kan slaan.

Het geheugen van een computer loopt bijvoorbeeld van adres 0 tot aan adres 32768.

Dan kan het zijn dat je op plaats 21 311 een plaats hebt waar je jouw naam hebt opgeslagen en op plaats 18711 heb je een plaats in het geheugen waar je jouw leeftijd hebt opgeslagen. Omdat deze getallen lastig zijn te onthouden kunnen we deze geheugenplaatsen een naam geven. Dat heet een variabele.

Variabele namen

Als je gaat programmeren ga je best veel gebruik maken van variabelen. We geven dus een *naam aan* een geheugenplekje, bijvoorbeeld plaats 21311 is waar jouw naam staat, noemen we *mijnnaam* en de plaats 18711 waar jouw leeftijd in staat noemen we *mijnleeftijd*.

Doosje of lade

n doosje c





Net als bij een doos of een lade, kan je de doos/lade openen en kijken wat er in zit; er kan niets in zitten, maar er kunnen ook getallen of zinnetjes in zitten. En....je kunt de inhoud ook veranderen.

In de eerste 2 minuten van deze video (NL)

Links to an external site.

wordt dit nog een keer uitgelegd.
In dit filmpje gaat het over de programmeertaal **Python**, maar dit geldt ook voor **PHP** (en bijna **alle** andere computertalen)

PHP Variabele

In PHP kun je een variabele makkelijk herkennen omdat die altijd met een $-teken begint, bijvoorbeeld `$mijnnaam`, `$mijnleeftijd` zijn twee variabelen.

Hoe ken je een waarde toe aan een variabele?

Stel jij heet *Mohammed* en je wilt de variable `$voornaam` de waarde *Mohammed* geven. Je gebruikt daar voor het = teken. Zie dit niet als '*is*' maar als '*wordt*' .

```
<?php
$mijnnaam="Mohammed";
?>
```

op regel 1 wordt de computer verteld dat je PHP code gaat gebruiken.

op regel 2 hier staat dat de variabele met de naam $*mijnnaam* de waarde Mohammed krijgt. Dus $*mijnnaam* wordt "Mohammed".

Mohammed is een zinnetje dus dat staat tussen "".

In dit geval is het een zinnetje van één woord, maar het zou ook meer woorden kunnen bevatten daarom staat een zinnetje, ook al is het maar één woordt altijd tussen "". De computer weet dan precies waar het zinnetje begint en waar het eindigt.

En niet vergeten de ; om aan te geven dat het commando klaar is!

> Elk PHP commando wordt in PHP (en ook in bijv. JavaScript en C#, C, C++ en Java) afgesloten met een ;

Even terug naar het doosje of de lade; we hebben dus een doosje gemaakt met het label *mijnnaam* en in het doosje hebben we de string-waarde "Mohammed" gezet.

Het = teken dat we hebben gebruikt kun je dus lezen als 'stop de volgende waarde in dit doosje'.

Variabele naamgeving

Zoals gezegd, een variabele naam in PHP begint altijd met een $-teken met daarachter een woord zonder spaties. De variabele bestaat dus altijd uit één woord. Omdat je soms duidelijk wil maken wat er in de variabele staat gebruik je goede namen die vaak wat langer zijn.

Bijvoorbeeld $*hoofdpersonagevandefilm*.

Om dit leesbaar te maken, maken we hiervan *hoofdPersonageVanDeFilm*.

Dus $*hoofdpersonagevandefilm* wordt $*hoofdPersonageVanDeFilm*

Zie je het verschil?

In code ziet het er dan zo uit.

```
$hoofdPersonageVanDeFilm = "Ryan Reynolds";
```

Zo bestaat de naam van de variabele nog steeds uit één woord, maar is het wel beter leesbaar.

Let op want de variabele $*mijnnaam* en $*mijnNaam* zijn twee verschillende variabelen. We zeggen dan dat variabelenamen hoofdlettergevoelig zijn (in het Engels; *case sensitive*).

Opdracht

Lees de tekst aandachtig door en maak de kennis-check in de les.

Inleveren

Schermafdruk van de kennischeck met een score van 80% of meer.

Variabelen 1

We gaan met deze code aan de slag.

```
<?php
$mijnNaam="Mohammed";
?>
```

Maak een nieuw bestand in c:\xampp\htdocs\phplevel1-jouw-naam

Noem het bestand php02-jouw-naam.php

Maak je eigen code en geef daarbij de variabele $mijnnaam de waarde van jouw naam.

Als je een variabele als een doosje ziet waar je iets in stopt, dan stop je dus jouw naam in het doosje $mijnnaam.

Test de code uit. Wat zie je? Niets? Dat klopt want we hebben geen output gemaakt.

We gaan een regel toevoegen aan onze code waarin we output maken. Tussen regel 2 en 3 **voegen** we een **extra** regel toe.

```
echo $mijnNaam;
```

Let op! De variabele naam is hier veranderd van `$ mijnnaam` naar `$ mijnNaam` . Je zult de naam van de variabele dus ook op regel 2 moeten aanpassen anders heb je twee verschillende variabelen. `$ mijnnaam` is namelijk ***niet*** dezelfde variabele als `$ mijnNaam` .

Test je code en als het goed is zie je nu dat de waarde van jouw variabele op het scherm (in je browser) wordt afgedrukt.

Inleveren

1. Het bestand php02-jouw-naam.php

Variabelen 2

Breid de code uit de vorige opdracht uit.

Voeg een **extra** variabele toe mijnLeeftijd en geef die de waarde van jouw-leeftijd. Druk ook de leeftijd af.

De leeftijd is een nummer en dus geen zinnetje. De leeftijd daarom niet tussen "" plaatsen!
Test je code!

Inleveren

1. Het bestand php03-jouw-naam.php
2. Een screenshot waarop jouw PHP-code te zien is.

mijnNaam en mijnLeeftijd

Wat ga je leren?

Code lezen en begrijpen wat er gebeurt.
Hiervoor krijg je code en worden er vragen over deze code gesteld. Je kunt alles gewoon *zelf* uitproberen, ga niet gokken, maar controleer je antwoorden door het zelf te proberen!
Als tweede opdracht in deze les krijg je een stukje code met twee foutjes. Kun jij deze foutjes vinden en oplossen?

We beginnen met deze foutloze code:

```
1  <?php
2
3  $mijnNaam="Random";
4  $mijnLeeftijd=23;
5
6  $mijnNaam="Foobar";
7  $mijnLeeftijd=$mijnLeeftijd+1;
8
9   echo "ik heet $mijnNaam en ik ben $mijnLeeftijd jaar oud.";
10
11  ?>
```

Kijk naar de code hierboven en probeer te *voorspellen* wat de output is. Wat er dus wordt afgedrukt.

Tip: als je de waarde van een variabele wil veranderen dan kun je de waarde gewoon overschrijven. Je hoeft de variabele niet eerst leeg te maken of.zo.

Maak een nieuw bestand in c:\xampp\htdocs\phplevel1-jouw-naam

Noem het bestand php04-jouw-naam.php

Zet de code erin en kijk of er inderdaad gebeurt wat jij had voorspelt.

Open MS Word, kopieer de vraag en tik daaronder jouw antwoord. Test aan de hand van code of jouw antwoord klopt.

```
1: Stel je drukt op regel 5 de waarde van $mijnNaam af, wat is dan de waarde?


2: Stel je drukt op regel 8 de waarde van $mijnNaam af, wat is dan de waarde?


3: Stel je drukt op regel 5 de waarde van $mijnLeeftijd af, wat is dan de waarde?


4: Stel dat je regel 4 twee maal uitvoert.


5: Stel dat je regel 7 twee maal uitvoert.
   Je kopieert dus regel 7 naar regel 8, zodat deze twee maal voorkomt.
   Wat is nu de waarde van $mijnLeeftijd vlak na regel 8?


6: Leg bij opdracht 5 in eigen woorden uit wat er gebeurt.
```

Inleveren

Sla het MS Word bestand op in PDF formaat met de naam PHP04-jouw-naam.pdf en lever dat in.

Fouten vinden

Als programmeur maak je code, maar een veel belangrijker onderdeel van je werk wordt het opsporen en verbeteren van fouten.

Soms zijn dat fouten die anderen hebben gemaakt, maar vaak zijn het ook fouten die je zelf hebt gemaakt. Het verbeteren van fouten wordt ook wel *debuggen* genoemd.

Maak een bestand PHP05-jouw-naam.php (let op dat dit bestand een PHP bestand is).

Kopieer de volgende code naar PHP05-jouw-naam.php en test de code uit.

```
<?php

$mijnNaam = "Random";
$mijnLeeftijdNu = 23;
$mijnLeeftijdVolgendJaar = $mijnLeeftijdNu+1

echo "ik heet $mijnnaam en ik ben $mijnLeeftijd jaar oud, volgend jaar ben ik
$mijnLeeftijdVolgendJaar oud.";

?>
```

De programmeur heeft op 2 regels foutjes gemaakt. Lees de foutmelding en probeer de twee fouten op te lossen.

Verbeter de code, test het en als alles goed werkt lever je dit bestand ook in.

Inleveren

1. PHP05-jouw-naam.php , hierin staat de *verbeterde* en *getest* code.

KennisCheck

 KC.jpg

Als je alles hebt gelezen en hebt uitgevoerd dan gaan jullie klassikaal een Kennischeck uitvoeren.

- Voer de juiste naam en klas in bij de kennis-check
- Zorg ervoor dat je 80% of hoger scoort.

Inleveren

1. Screenshot van je resultaat.

Revision #14
Created 2024-06-08 18:32:54 UTC by Max

Updated 2025-06-04 08:53:12 UTC by Max