PHP3

python reformatterCanvas.py 18661 52533 https://www.roc.ovh/books/2024/page/php3

01 De Som

We gaan verder met functies. Weet je nog wat een functie is? Een functie heeft (meestal) parameters, dit is de input en de output wordt door een functie teruggegeven aan waar de functie wordt aangeroepen.

image-1727530367600.png

Voorbeeld

```
<?php


function telErEenBijOp($getal) {
  $output = $getal + 1;
  return $output;
}
```

Functienaam: de functie heet `telErEenBijOp/`

Input: De input is `$getal`

Output: De output wordt in de functie berekend in regel 4. In regel 4 staat `$output` wordt `$getal` plus één.

Nu gaan we zelf fucnites maken.

De Som van drie getallen

De functie heeft drie waarden als input en returned de som van de drie getallen (dus de drie input variabelen opgeteld).

Input	Output
mijnFunctie(3,2,1)	6
mijnFunctie(1,12,3)	16

mijnFunctie(0,4,0)	4
mijnFunctie(7,0,3)	10
mijnFunctie(0,0,0)	0

Kijk naar dit schema en kijk of je begrijpt wat er gebeurt. Telkens wordt de input getransformeerd (veranderd). De input 3,2,1 wordt bijvoorbeeld verandert in 6, de input 1,12,3 in 16 etc.

Maak een functie die dit doet.

En test met alle input waarden zoals hierboven aangegeven. Gebruik de volgende template:

```
function mijnFunctie($a, $b, $c) {
   // maak hier de code voor jouw functie
   ...
   return $uitkomst;
}

echo "mijnFunctie(3,2,1): ";
echo mijnFunctie(3,2,1);
echo "<br>";
echo "mijnFunctie(1,12,3): ";
echo mijnFunctie(1,12,3);
echo "<br>";
echo "mijnFunctie(0,4,0): ";
echo mijnFunctie(0,4,0);
echo "<br>";
echo "mijnFunctie(7,0,3): ";
echo mijnFunctie(7,0,3);
echo "<br>";
echo "mijnFunctie(0,0,0): ";
echo mijnFunctie(0,0,0);
```

Klopt je output?

Je output moet er dus als volgt uit zien:

Screenshot 2022-10-05 153345.png

Opdracht

1. Maak de code en zorg dat de output klopt zoals hierboven is aangegven.

2. Voeg één extra test zelf toe, dus *onder* de mijnFunctie(0,0,0) gebruik de 3 gegevens van je geboortedatum als input (bijv. 1,4,2000 als je op 1 april 2000 bent geboren).

Inleveren (2 bestanden)

* php71-jouw-naam.php
* schermafdruk van de test waarbij alle juiste waarden worden afgedrukt zoals in het voorbeeld.

02 Optellen en de nul

De som van drie getallen, maar niet altijd

De functie krijgt drie waarden en returned de som van de drie getallen (dus de drie input variabelen opgeteld).

Er is nu een **uitzondering** . Zodra er een 0 in één van de input variabelen voorkomt, dan wordt de return waarde 0.

Input	**Output**
mijnFunctie(3,2,1)	6
mijnFunctie(1,12,3)	16
mijnFunctie(0,4,0)	0
mijnFunctie(7,0,3)	0
mijnFunctie(0,0,0)	0

Maak een functie die dit doet en plaats testcode in je code die alle bovenstaande situaties test. Gebruik eenzelfde soort template als in de vorige opgave.

Let op dat je een return gebruikt in de functie!

De output ziet er dan als volgt uit:

Screenshot 2022-10-05 154108.png

Tip : gebruik voor de situatie met een 0, een conditional statement en een extra return.

Opdracht

1. Maak de code en zorg dat de output klopt zoals hierboven is aangegven.

2. Voeg één extra test zelf toe, dus *onder* de mijnFunctie(0,0,0) bedenk je zelf die willekeurige getallen tussen 0 en 100 die je als input geeft.

Inleveren (2 bestanden)

* php72-jouw-naam.php
* schermafdruk van de test waarbij alle juiste waarden worden afgedrukt zoals in het voorbeeld.

03 Tussen 0 en 10?

De som van drie getallen met voorwaarden

De functie telt de drie input waarden bij elkaar op.

Is de som groter of gelijk aan 10 dan wordt de return waarde 1.
Is de som kleiner dan 10 en groter dan 0 dan wordt de return waarde 0.
Is de som 0 dan wordt de return waarde -1.

Input	**Output**
mijnFunctie(3,2,1)	0
mijnFunctie(1,12,3)	1
mijnFunctie(0,4,0)	0
mijnFunctie(7,0,3)	1
mijnFunctie(0,0,0)	-1

Maak een functie die dit doet. Test alle input waarden weer net als in de vorige opgave.

Vergeet niet een return te gebruiken in de functie!

De output ziet er als volgt uit:

Screenshot 2022-10-05 160616.png

Opdracht

1. Maak de code en zorg dat de output klopt zoals hierboven is aangegeven.
2. Voeg één extra test zelf toe, dus *onder* de mijnFunctie(0,0,0) bedenk je zelf die willekeurige getallen tussen 0 en 100 die je als input geeft.

Inleveren (2 bestanden)

* php73-jouw-naam.php
* schermafdruk van je gehele browser waarin je laat zien dat je code werkt (zoals in het voorbeeld)

04 Array optellen

De som van alle waarden in een array

De functie telt alle waarden van een array bij elkaar op. De som alle getallen van het array wordt door de functie ge-returned.

Let op het input array kan verschillende lengtes hebben.

Input	**Output**
mijnFunctie([3,2,1])	6
mijnFunctie([1,12,3,1,2])	19
mijnFunctie([1,12,3,1,2,1])	20
mijnFunctie([2,4])	6
mijnFunctie([4])	4

Maak een functie die dit doet.

Je kunt een array doorlopen met een foreach-loop.

Je kunt de template van opdracht 1 gebruiken, maar je moet deze wel aanpassen.

Waarom een return gebruiken?

We maken telkens een functie die het resultaat via een `return` als het ware terug geeft aan de code die de functie aanroept. Dat doen we omdat we mogelijk met de uitkomst nog meer berekeningen willen doen. Soms wordt de output van de ene functie de input van de volgende functie. Dat kan alleen als je het resultaat via een return 'terug' geeft.

Opdracht

1. Maak de code en zorg dat de output klopt zoals hierboven is aangegeven.
2. Zet **je naam** en de **datum** als commentaar bovenin je code.

Inleveren (2 bestanden)

* php74-jouw-naam.php

* schermafdruk van de test waarbij alle juiste 5 waarden worden afgedrukt zoals in het voorbeeld.

05 Voetbal

De voetbalclub FCR (Footbal Club Royal) heeft vijf wedstrijden gespeeld.
De uitslagen waren 1-3, 4-0, 0-0, 1-1 en 0-2.

De voetbaluitslagen staan in een array.

```
$uitslagen = [ [1,3], [4,0], [0,0] ,[1,1],[0,2] ];
```

Voor een wedstrijd krijgt FCR 0, 1 of 3 punten.

Uitslag	Resultaat	Punten
1-3	verloren	0
4-0	gewonnen	3
0-0	gelijk	1
1-1	gelijk	1
0-2	verloren	0

FCR heeft in bovenstaand voorbeeld 5 punten.

Maak een functie die de punten aan de hand van de uitslagen berekend.

Input	Punten
[[1,3], [4,0], [0,0] ,[1,1],[0,2]]	5
[[1,1], [0,0], [0,2] ,[1,1],[0,2]]	3
[[1,1], [0,0], [0,0] ,[1,1],[2,2]]	5
[[1,0], [1,0], [2,0] ,[1,1],[0,2]]	10

```
function berekenScore($uitslagen) {
  // bereken score


  return $score;
}

echo berekenScore([ [1,3], [4,0], [0,0] ,[1,1],[0,2] ]);
echo "<br>";
echo berekenScore([ [1,1], [0,0], [0,2] ,[1,1],[0,2] ]);
echo "<br>";
echo berekenScore([ [1,1], [0,0], [0,0] ,[1,1],[2,2] ]);
echo "<br>";
echo berekenScore([ [1,0], [1,0], [2,0] ,[1,1],[0,2] ]);
```

Inleveren (2 bestanden)

* php75-jouw-naam.php

* schermafdruk waarin je laat zien dat je je code test met de bovenstaande test cases.

06 Swap values

De opdracht heet "Swap values", dit betekent "waarden verwisselen"

In de vorige opgave hadden we het over de uitslag van een voetbalwedstrijd. Een uitslag 1,0 betekende dat de thuisploeg (FCR) 1 doelpunt heeft gemaakt en de uit spelende ploeg had 0 punten gescoord.

Soms staat de uitslag in de verkeerde volgorde. FCR heeft met 3-2 gewonnen en de uitslag is 2-3.

Nu moet jij een functie maken die twee getallen in een array omdraait.

Input	Output
swap([1,2])	[2,1]
swap([4,2])	[2,4]
swap([1,1])	[1,1]

De input en output zijn beiden een array.

Template

```
function swap($uitslag) {
  // swap de twee elementen van het array



  return $swap;
}

print "<pre>";
print_r( swap([1,2]) );
echo "<br>";
print_r( swap([4,2]) );
echo "<br>";
print_r( swap([1,1]) );
```

Output

Je ziet als het goed is dat je een array terug krijgt waarbij de waarden zijn omgedraaid. De output ziet er dan als volgt uit:

```
Array
(
    [0] => 2
    [1] => 1
)

Array
(
    [0] => 2
    [1] => 4
)

Array
(
    [0] => 1
    [1] => 1
)
```

Inleveren

* php76-jouw-naam.php
* schermafdruk van de test waarbij alle juiste waarden worden afgedrukt

07 Druk ass. array af

Terugblik

Tot nu toe hebben we twee soorten arrays gezien.

- Eenvoudige arrays
- Complexe arrays (multidimensionale arrays).

Voorbeelden

```
$array1=['waarde 1', 'waarde 2', 'waarde 3', 'waarde 4'];
$array2=[ ['waarde 1a', 'waarde 2a', 'waarde 3a', 'waarde 4a'],
          ['waarde 1b', 'waarde 2b', 'waarde 3b', 'waarde 4b'],
          ['waarde 1c', 'waarde 2c', 'waarde 3c', 'waarde 4c']
        ];
```

Herken je het complexe array?

$array2 is een array van 3 arrays en elk van die drie arrays heeft 4 elementen.

Associatieve array

In deze les gaan we kennis maken met nog een type array, het *associatieve array*.

Er bestaan verschillende namen voor een associatieve array:
In Python heet het een dictionairy en soms wordt ook wel de meer algemenere term "key-value array" gebruikt. Je zult aan het eind van de les wel begrijpen waarom.

> associatieve array = dictionary = key-value array

Het verschil tussen een 'gewoon' (indexed) array en een associatieve array is de key (sleutel).

Snap je dit nog niet helemaal, of wil je nog meer voorbeelden zien, kijk dan deze video.

https://www.youtube.com/watch?v=5lJLecI0BTA



De key

Met een key bepaal je welke elementen je wilt zien in een array. Bijvoorbeeld:

```
$array=['Amsterdam','Rotterdam','Den Haag','Almere'];
echo $array[1];
```

Kijk naar regel 2, daar staat een [1]. Dat is de key . De key is 1 en dat betekent dat je het tweede element van het array afdrukt. In dit geval '*Rotterdam*' dus. Nee niet '*Amsterdam*' want een array begint te tellen bij 0.

Stel je wilt de plaats 'Almere' overschrijven? Dat kan met:

```
$array[3]='Utrecht';
```

Bij een associatieve array moet jij zelf voor elk element de key bepalen. Dat ziet er bijvoorbeeld zo uit:

```
$array[ 'naam' => 'Nike Chui' , 'datum' => '12 november 2020' , 'project' => 'PHP Project'];
```

Om nu het tweede element van dit array af te drukken gebruik je:

```
echo $array['datum'];
```

Het maakt niet uit op welke plaats de datum staat, de key is datum. De waarde die bij deze key hoort, wordt afgedrukt.

Stel je wilt de datum aanpassen dan kun je dat als volgt doen:

```
$array['datum']='1 december 2020';
```

Samengevat

	Andere namen	Hoe benader je een element?	Voorbeeld
Indexed Array	gewoon array, 1 dimensionaal array, eenvoudig array	via de key, die is altijd 0,1,2,3,4....	$myArray[13]
Associatieve Array	key-value array, dictionary (Python), hash table	via de key, dit moet een unieke string zijn waarvan je zelf de waarde bepaald.	$myArray['datum'];

Opdracht

Lees eerst de tekst/uitleg hierboven.

Neem dan deze template en vul de code aan zodat je het hele associatieve array afdrukt.

```
$array=[ 'naam' => 'Nike Chui' , 'datum' => '12 november 2020' , 'project' => 'PHP Project'];
```

De output moet er als volgt uit komen te zien (dus elke waarde op een nieuwe regel).

```
Nike Chui
12 november 2020
PHP Project
```

Let op, het gebruik van een **(for) loop** is bij deze opgave ***niet*** toegestaan.

Inleveren (2 bestanden)

1. php77-jouw-naam.php
2. schermafdruk met de juiste output (mag in ander font als in het voorbeeld)

08 Key and value

In de vorige opdracht werden alle waarden afgedrukt. We kunnen ook de keys en de waarden afdrukken met een loop. Dat is vooral handig als we niet weten hoe groot het array. We maken een loop en drukken gewoon het hele associatieve array af.

We gebruiken hetzelfde array, maar de output moet anders worden, namelijk:

```
key=naam, value=Nike Chui
key=datum, value=12 november 2020
key=project, value=PHP Project
```

Zoek zelf op hoe je dat doet, bijvoorbeeld op:

https://www.w3schools.com/php/php_arrays_associative.asp

Gebruik de foreach loop om alle keys en values af te drukken. De output moet er precies zo uitzien als in deze opdracht staat (let dus op de komma's en de regels).

Inleveren (2 bestanden)

* php78-jouw-naam.php

* schermafdruk van de output

09 Maak ass. array

Maak zelf een associatieve array met de volgende 4 elementen:

```
voornaam, achternaam, geboortedatum, woonplaats
```

Maak het array door gebruik te maken van [en].

Gebruik als waarden je eigen voornaam/achternaam/geboortedatum/woonplaats.

Druk alle waarden uit dit associatieve array af (hoe, met of zonder loop, dat mag je zelf weten).

Inleveren (2 bestanden)

* php79-jouw-naam.php.

* schermafdruk van de output in de browser.

10 array from database

Wat je in de praktijk vaak tegenkomt zijn arrays van associatieve arrays. Als je (later) gegevens uit de database haalt dan krijg je ook vaak een array van associatieve arrays terug. Hoe ziet dat eruit?

```
$uitslagen=[
[ 'thuis' => 'FC Twente', 'uit' => 'FC Utrecht', 'uitslag' => [0,1] ],
[ 'thuis' => 'FC Twente', 'uit' => 'FC Volendam', 'uitslag' => [4,2] ],
[ 'thuis' => 'FC Emmen', 'uit' => 'Feyenoord', 'uitslag' => [1,3] ],
[ 'thuis' => 'Vitesse', 'uit' => 'FC Twente', 'uitslag' => [0,1] ],
];
```

Elke regel is eigenlijk een associatieve array. Er staan 4 associatieve arrays in het array $uitslagen.

Probeer maar eens:

```
echo "<pre>";
print_r($uitslagen[0]);
echo "</pre>";
```

Met print_r() kun je een array afdrukken.

Je kunt op deze manier dus $uitslagen[0], $uitslagen[1], $uitslagen[2] en $uitslagen[3] afdrukken.

Stel je wilt de thuisploeg afdrukken van de tweede uitslag. Hoe gaat dat?

```
echo $uitslagen[1]['thuis'];
```

En... stel je wilt weten hoeveel de thuisploeg van de derde wedstrijd heeft gescoord.

```
echo $uitslagen[3]['uitslag'][0]
```

Je begint dus met de vierde regel (3) dan neem je de 'uitslag' en van de uitslag neem je dan het eerste element (0).

OK, nu jij. Gebruik het array $uitslagen en druk de volgende gegevens af:

Het aantal doelpunten dat FC Twente heeft gescoord.

De output moet er als volg uit zien.

```
FC Twente scoort 0 punten
FC Twente scoort 4 punten
FC Twente scoort 1 punten
```

De waarden 0, 4 en 1 moet je uit het (gegeven) associatieve array halen.

Inleveren (2 bestanden)

* php84-jouw-naam.php
* schermafdruk van je gehele browser en laat daarbij de output zien

Bonusvraag

Deze vraag levert (2) bonuspunten op

Je weet nu waar precies de uitslag van 'FC Twente' staat. Stel dat je dat niet weet en je moet regel voor regel eerst kijken of 'FC Twente' heeft gespeeld en zo ja of het de thuis- of uitploeg was. Hoe zou je code er dan uit zien?

Tip: je moet hierbij een loop gebruiken!

10 array from database versie b

Dit is een extra/alternatieve opgave. Alleen maken indien je te weinig punten hebt gehaald met de andere opdrachten.

Wat je in de praktijk vaak tegenkomt zijn arrays van assioicative arrays. Als je (later) gegevens uit de database haalt dan krijg je ook vaak een array van assiociative arrays terug. Hoe ziet dat eruit?

```
$uitslagen=[
[ 'thuis' => 'FC Twente', 'uit' => 'FC Utrecht', 'uitslag' => [3,1] ],
[ 'thuis' => 'FC Twente', 'uit' => 'FC Volendam', 'uitslag' => [4,4] ],
[ 'thuis' => 'Vitesse', 'uit' => 'Feyenoord', 'uitslag' => [0,2] ],
[ 'thuis' => 'FC Emmen', 'uit' => 'Feyenoord', 'uitslag' => [0,3] ],
[ 'thuis' => 'Vitesse', 'uit' => 'FC Twente', 'uitslag' => [1,0] ],
[ 'thuis' => 'Feyenoord', 'uit' => 'FC Twente', 'uitslag' => [1,1] ],
];
```

Elke regel is eigenlijk een assioicative array. Er staan 6 assiociative arrays in het array $uitlsagen.

Probeer maar eens:

```
echo "<pre>";
print_r($uitslagen[0]);
```

Met print_r() kun je een array afdrukken.

Je kunt op deze manier dus $uitslagen[0], $uitslagen[1], $uitslagen[2] en $uitslagen[3] afdrukken.

Stel je wilt de thuisploeg afdrukken van de tweede uitslag. Hoe gaat dat?

```
echo $uitslagen[1]['thuis'];
```

En Stel je wilt weten hoeveel de thuisploeg van de derde wedstrijd heeft gescoord.

```
echo $uitslagen[3]['uitslag'][0]
```

Je begint dus met de vierde regel (3) dan neem je de 'uitslag' en van de uitslag neem je dan het eerste element (0).

OK, nu jij. Gebruik het array $uitslagen en druk de volgende gegevens af:

Het aantal doelpunten dat FC Twente heeft gescoord.

De output moet er als volg uit zien.

```
FC Twente scoort 3 punten
FC Twente scoort 4 punten
FC Twente scoort 0 punten
FC Twente scoort 1 punten
```

De waarden 3, 4, 0 en 1 moet je uit het assiociative array halen.

Inleveren

1. Lever de code in en gebruik de naam php84-jouw-naam.php
2. Lever een schermafdruk van je gehele browser en laat daarbij de output zien, gebruik de naam php84-jouw-naam.png

11 Thuisploeg

Gebruik het associatieve array (van de vorige opdracht);

```
$uitslagen=[
[ 'thuis' => 'FC Twente', 'uit' => 'FC Utrecht', 'uitslag' => [0,2] ],
[ 'thuis' => 'FC Twente', 'uit' => 'FC Volendam', 'uitslag' => [5,1] ],
[ 'thuis' => 'FC Emmen', 'uit' => 'Feyenoord', 'uitslag' => [0,1] ],
[ 'thuis' => 'Vitesse', 'uit' => 'FC Twente', 'uitslag' => [0,0] ],
];
```

Je gaat nu in een loop de punten van de thuisploeg uitrekenen.

Je krijgt 3 punten als je de wedstrijd wint, 1 bij een gelijkspel en 0 als je de wedstrijd verliest.

In de eerste wedstrijd heeft de thuisploeg 0 punten gescoord, in de tweede wedstrijd 3, in de derde wedstrijd 0 en in de vierde wedstrijd 1.

De output van jouw code ziet er als volgt uit:

```
Wedstrijd 1: de thuisploeg krijgt 0 punten
Wedstrijd 2: de thuisploeg krijgt 3 punten
Wedstrijd 3: de thuisploeg krijgt 0 punten
Wedstrijd 4: de thuisploeg krijgt 1 punten
```

Lastig? Probeer dit in stapjes te doen:

Stap 1, maak eerst een loop waarin je de uitslagen afdrukt (gebruik print_r).

Stap 2, druk nu de uitslagen niet af met print_r maar druk de afzonderlijke scores af, dus 0 1 en dan 3 1 enzovoorts.

Stap 3 gebruik nu een if, en test of het eerste getal groter is dan het tweede. Als dat zo is dan druk je 3 af omdat de thuisploeg dan 3 punten krijgt.

Stap 4 maak nog een if, en test of het eerste en tweede getal gelijk zijn. Als dat zo is dan druk je 1 af.

Stap 5 maak nog een if, en test of het tweede getal groter is dan het eerste. als dat zo is dan druk je 0 af.

Stap 6 je bent bijna klaar je moet nu alleen nog text om de uitslag heen printen. Dus druk nu niet 0 af, maar 'de thuisploeg krijgt 0 punten.

Stap 7, nu nog het eerste gedeelte. Maak een variabele $teller en zet de waarde op 1. Druk af Wedstrijd $teller en hoog daarna de $teller op met 1 ($teller = $teller + 1)

Inleveren (2 bestanden)

* php85-jouw-naam.php

* schermafdruk van je gehele browser en laat daarbij de output zien, gebruik de naam php85-jouw-naam.png

*12 Eindopdracht Boeken **

Opdracht

Je krijgt een associatief array en daarmee ga je het volgende overzicht maken.

image.png

De rode lijn is de cursor, als je met de muis over de tabel gaat dan wordt de regel die waar je muis over heen gaat een andere kleur. Dat geldt natuurlijk niet voor de titel regel van de tabel.

Het array waar je mee gaat werken is:

```
$boeken = [
    "Harry Potter en de Steen der Wijzen" => ["auteur" => "j.k. rowling", "jaar" => 1997],
    "De Hobbit" => ["auteur" => "J.r.r. Tolkien", "jaar" => 1937],
    "Het spel der tronen" => ["auteur" => "George R.R. martin", "jaar" => 1996],
    "1984" => ["auteur" => "George Orwell", "jaar" => 1949],
    "Moord op de Oriënt-Expres" => ["auteur" => "Agatha Christie", "jaar" => 1934],
    "Moby Dick" => ["auteur" => "herman melville", "jaar" => 1851]
];
```

Deel 1 van de opdracht, HTML, PHP en CSS (10 punten)

- Voeg zelf **twee boeken** toe aan dit array!
- Voeg de CSS-styling toe aan HTML (inline).

Deel 2 van de opdracht, functie (10 punten)

Zoals je ziet staan niet alle namen van de auteurs goed in de data $boeken.

Zo staat er:

j.k. rowling in plaats van J.K. Rowling

J.r.r. Tolkien in plaats van J.R.R. Tolkien

Maak een functie genaamd:

formatName($naam)

deze functie herstelt de naam zodat de hoofdletters goed komen te staan.

Dus elk woord in de $naam en elke letter die wordt gevolgd door een punt (.) moet een hoofdletter worden.

Met behulp van de functie worden dan alle namen zo afgedrukt als in het voorbeeld.

Je mag ChatGPT gebruiken maar zorg er wel voor dat je de code kan uitleggen. Het kan zijn dat docenten hier om vragen!

Inleveren

1. php/html bestand met inline css.

Revision #7
Created 2024-06-08 15:41:07 UTC by Max
Updated 2024-09-28 13:46:55 UTC by Max