2.1 Extra - oefeningen

Extra oefenstof

De volgende opgaven staan in oplopende moeilijkheidsgraad en dienen als extra oefenmateriaal. Bij de moeilijkere opgave zul je soms zelf op zoek moeten gaan naar antwoorden. Er worden zaken gevraagd die niet allemaal in de les zijn behandeld. net zoals in het echte leven van een Software Developer!

Opgave 1

```
<?php $maanden = array ('Januari', 'Februari', 'Maart', 'April', 'Mei', 'Juni', 'Juli',
'Augustus',
                        'September', 'Oktober', 'November', 'December'); ?>
...
...
```

Maak deze code af zodat de output er zo uit ziet en gebruik daarbij een loop.

```
Maand 1 is Januari.
Maand 2 is Februari.
Maand 3 is Maart.
Maand 4 is April.
Maand 5 is Mei.
Maand 6 is Juni.
Maand 7 is Juli.
Maand 8 is Augustus.
Maand 9 is September.
Maand 10 is Oktober.
Maand 11 is November.
Maand 12 is December.
```

Opgave 2

Maak een drop-down menu met de jaartallen 1940 tot en met het huidige jaar. Het huidige jaar kan worden opgevraagd met de PHP-functie `date('Y')`. We gaan nu niet in op de werking hiervan, maar hiermee kan worden voorkomen dat het formulier ieder jaar moet worden aangepast. Vul `date('Y')`

dus in plaats van 2019 in.

Gebruik een loop.

Pas het drop-down menu aan, zodat de jaartallen in aflopende volgorde worden weergegeven en dat de geboortejaren voor mensen jonger dan 18 worden weggelaten.

Opgave 3

```
<?php
$eten = array(
    'fruit' => array('geel' => array('banaan', 'citroen'), 'oranje' => array('sinaasappel',
'mandarijn')),
    'groente ' => array('asperge', 'broccoli', 'courgette')
);
...
...
```

Maak het stukje code af en druk daarmee alle groenten onder elkaar af, zoals:

- asperge
- broccoli
- courgette

Gebruik de HTML <UL> om een *unnumbered list* te krijgen zoals hierboven aangegeven.

Opgave 4

Gebruik het stukje code van opgave 3. Maak een extra stuk code. De code zorgt ervoor dat alle fruit van een bepaalde kleur afgedrukt wordt. Maak daarvoor een *functie* die als parameter het array en de kleur mee krijgt, dus

`drukGeelFruitAf($eten,$kleur);`

Opgave 5

Maak een functie die twee parameters mee krijgt, $string en $nummer. De output van de functie is de return value is het karakter van de string op de positie n, bijvoorbeeld:

```
myFunct("Hallo",2) -> "a"
myFunct("Hallo",4) -> "l
myFunct("Hallo",9) -> ""
```

Opgave 6

Maak nu een functie die letter n uit een string haalt. Bijvoorbeeld:

```
myFunct("Hallo",2) -> "Hllo"
myFunct("Hallo",4) -> "Halo
myFunct("Hallo",9) -> "Hallo"
```

Opgave 7

logfile

Maak de onderstaande code af. De functie stuurt de regel tekst in $string naar een logfile. De naam van de logfile wordt gedefinieerd in de constante LOGFILE. Gebruikt deze voor de logfile naam.

```
<?

define("LOGFILE", "logfile.txt");

myLog("Oops er ging iets mis in de code op regel 5");

function myLog($string) {
  .....
}
```

Elke keer als je de functie aanroept, wordt er een nieuwe regel aan de logfile toegevoegd. Deze regel begint met de datum en tijd en dan volgt de regel tekst. De log file zou er zo kunnen uitzien:

```
...
17-09-2019 21:35 Oops er ging iets mis in de code
17-09-2019 21:36 Oops er ging weer iets is in de code op regel 12
17-09-2019 21:36 User heeft 12 keer proberen aan te loggen met user id admin, inloggen
geblokeerd
...
```

Opgave 8

Page teller

Maak een teller die bijhoud hoe vaak een pagina wordt bezocht. Elke keer als een pagina wordt bezocht dan roep jet de functie teller() aan. De functie teller leest een getal uit een file. De file name is gedefineerd in een CONSTANTE (net als LOGFLE bij opgave 7). In de file staat gewoon een nummer, bijvoorbeeld 12. Dit nummer wordt ingelezen, met één opgehoogd en terug geschreven naar de file. In de file staat nu dus het nummer 13.

Opgave 9

Maak en functie die als parameter een string krijgt. Deze string is een regel text en deze regel wordt omgezet zodat elk woord met een hoofdletter begint. Dus:

```
myFunct("HALLO, Ik ben een goede programmeur!") -> Hallo, Ik Ben Een Goede Programmeur!
```

Let op je mag *geen* gebruik maken van de functie php ucfirst()

Opgave 10

Maak een functie die een aan de hand van de geboortedatum de leeftijd van een persoon bepaald. Dus input is de geboortedatum en output is de leeftijd. De geboortedatum wordt in drie verschillende variabelen aan de functie gegeven zodat je deze variabelen later makkelijk via een form (apart) kunt opvragen.

Dus:

```
(dit voorbeeld is van 19 september 2019)
myFunct(23,8,2010) -> 9 jaar
myFuntt(23,12,2010) -> 8 jaar
```

Opgave 11a

Wat doet de php functie scandir("c:")?

Voer de functie uit en maak een lijst van alle directories in je C-Drive, de output ziet er bijvoorbeeld zo uit:

```
Config.Msi
DRIVERS
```

```
Documents and Settings
Intel
PerfLogs
Program Files
Program Files (x86)
ProgramData
Recovery
Reflect_Install.log
....
```

Opgave 11b

De lijst die je bij opgave 6 afdrukt bevat directories en files. Met de fuctie is_dir() kun je bepalen of een item een directory (map) is af dat het bijvoorbeeld een file is. Pas de code aan en Druk het woord Directory af voor elke directory.

Zet de code in een *functie*. De functie krijgt als parameter de directory die moet worden getoond. In het voorbeeld is dat "C:".

Opgave 11c

Als je een directory anders dan . of .. tegenkomt dat voer je opnieuw de functie die je hebt gecreëerd uit. Dus de functie roept zichzelf aan. Je noemt dit recursief. Als het goed is worden nu alle directories en files afgedrukt. Filter de output nu zodanig dat je alleen alle files ziet. Directories worden dus niet afgedrukt.

Opgave 11d

Files zoeken

Je hebt nu een functie die alle files op jouw harde schijf afdrukt. Maak nu een zoekfunctie die zoekt naar een bepalade file naam. Maak daarbij gebruik van een wild card. mySearch('Te*') zoekt dus alle documenten op die met Te beginnen. Maar de seacht is case insensensitive, dus mySearch('Te*') vindt alle documenten die met te, TE, Te of tE begint.

--

Revision #6
Created 2019-09-19 19:04:04 UTC by Admin
Updated 2020-04-04 21:46:04 UTC by Max