

# INDEX.PHP – IMPORTING CSV

In de functie `make_model` importeer ik alle modellen en zet ik ze in een dictionary. Dit wordt dan teruggegeven en in de juiste variabele gezet

```
1
2
3 function make_model(string $file_name)
4 {
5
6     $open = fopen("./".$file_name.".csv", "r");
7     $data = fgetcsv($open, 1000, ",");
8     $total_list = array();
9
10
11 while (($data = fgetcsv($open, 1000, ",")) != FALSE)
12 {
13     if ( $file_name == "Product" ) {
14
15         $list_item = new Product(
16             $itemnr      = count($total_list),
17             $naam        = $data[0],
18             $minimumVoorraad = (int) $data[1],
19             $aantalInVoorraad = (int) $data[2],
20             $prijs        = (float) $data[3],
21             $actief       = (bool) $data[4]);
22
23     }
24
25     if ( $file_name == "DVD" ) {
26
27         $list_item = new DVD(
28             $lengteInMinuten = $data[0],
29             $jaarUitgifte    = (int) $data[1],
30             $filmStudio      = $data[2],
31
32             $itemnr      = count($data),
33             $naam        = $data[3],
34             $minimumVoorraad = (int) $data[5],
35             $aantalInVoorraad = (int) $data[4],
36             $prijs        = (float) $data[6],
37             $actief       = (bool) $data[7]
38
39         );
40     }
41
42     if ( $file_name == "CD" ) {
43         $list_item = new CD(
44             $artiest      = $data[0],
45             $aantal_songs = (int) $data[1],
46             $label        = $data[2],
47
48             $itemnr      = count($data),
49             $naam        = $data[3],
50             $minimumVoorraad = (int) $data[4],
51             $aantalInVoorraad = (int) $data[5],
52             $prijs        = (float) $data[6],
53             $actief       = (bool) $data[7]
54
55         );
56     }
57
58     array_push($total_list, $list_item);
59 }
60
61 fclose($open);
62
63 return $total_list;
64 }
65
```

## DVD.PHP – MODEL

In de DVD.php worden alle variabelen toegewezen via de bovenliggende klasse van Product.

```
1  <?php
2
3  class DVD extends Product {
4
5      public $lengteInMinuten;
6      public $jaarUitgifte;
7      public $filmStudio;
8
9      /**
10     * Class constructor.
11     */
12     public function __construct(
13         int     $lengteInMinuten    ,
14         float   $jaarUitgifte       ,
15         string  $filmStudio         ,
16
17         int     $itemnr             ,
18         string  $naam               ,
19         int     $minimumVoorraad    ,
20         int     $aantalInVoorraad   ,
21         float   $prijs              ,
22         bool    $actief              = false )
23     {
24         $this->lengteInMinuten = $lengteInMinuten;
25         $this->jaarUitgifte    = $jaarUitgifte;
26         $this->filmStudio      = $filmStudio;
27         parent::__construct($itemnr, $naam, $minimumVoorraad, $aantalInVoorraad, $prijs, $actief);
28     }
29
30     public function totalWorthDvd()
31     {
32         return parent::getTotalWorth() * 1.05;
33     }
34
35     public function calcTime()
36     {
37         $rawTimeInMinutes = strval($this->lengteInMinuten / 60);
38         return str_replace(".", ":", $rawTimeInMinutes);
39     }
40
41
42 }
43
44 }
45
46
```

## CD.PHP – MODEL

In de CD.php worden alle variabelen toegewezen via de bovenliggende klasse van Product. Dit is dus hetzelfde als de DVD.php

```
1 <?php
2
3 class DVD extends Product {
4
5     public $lengteInMinuten;
6     public $jaarUitgifte;
7     public $filmStudio;
8
9     /**
10      * Class constructor.
11      */
12     public function __construct(
13         int    $lengteInMinuten    ,
14         float  $jaarUitgifte       ,
15         string $filmStudio        ,
16
17         int    $itemnr            ,
18         string $naam              ,
19         int    $minimumVoorraad   ,
20         int    $aantalInVoorraad  ,
21         float  $prijs             ,
22         bool   $actief            = false )
23     {
24         $this->lengteInMinuten = $lengteInMinuten;
25         $this->jaarUitgifte    = $jaarUitgifte;
26         $this->filmStudio      = $filmStudio;
27         parent::__construct($itemnr, $naam, $minimumVoorraad, $aantalInVoorraad, $prijs, $actief);
28     }
29
30     public function totalWorthDvd()
31     {
32         return parent::getTotalWorth() * 1.05;
33     }
34
35     public function calcTime()
36     {
37         $rawTimeInMinutes = strval($this->lengteInMinuten / 60);
38         return str_replace(".", ":", $rawTimeInMinutes);
39     }
40
41 }
42
43 }
44
45 }
46
```

## PRODUCT.PHP

In de zogenaamde

Bovenliggende class. In dit geval is het Product.php is waar alle belangrijke variabelen zijn geset zodat het programma later weet hoe hij ermee om moet gaan of om dat a op te slaan.

Er worden ook functies gemaakt die in elke klasse worden toegevoegd om ervoor te zorgen dat deze niet overbodig vaak moet worden bijgevoegt en maakt de code duidelijker en dus gestructureerder.

Er staan hier ook getters en setters, die in dit geval dynamisch worden gemaakt, maar voor de veiligheid vaak apart aangemaakt worden.

```
1  <?php
2
3  class Product {
4
5      public $itemnr;
6      public $naam;
7      public $aantalInVoorraad;
8      public $minimumVoorraad;
9      public $prijs;
10     public $actief;
11
12     public function __construct(
13         int     $itemnr           ,
14         string  $naam             ,
15         int     $minimumVoorraad ,
16         int     $aantalInVoorraad ,
17         float   $prijs            ,
18         bool    $actief           = false )
19     {
20         $this->itemnr       = $itemnr;
21         $this->naam         = $naam;
22         $this->minimumVoorraad = $minimumVoorraad;
23         $this->aantalInVoorraad = $aantalInVoorraad;
24         $this->prijs        = $prijs;
25         $this->actief       = $actief;
26     } // END Class constructor
27
28
29     public function downerStock($aantal) {
30         $stockSizeAfterAction = $this->aantalInVoorraad - $aantal;
31
32         if ( $stockSizeAfterAction > $this->minimumVoorraad )
33         {
34             $this->aantalInVoorraad = $stockSizeAfterAction;
35         }
36     }
37
38     public function getTotalWorth()
39     {
40         return ($this->prijs * $this->aantalInVoorraad);
41     }
42
43     public function upperStock($aantal) {
44         if ( $this->actief )
45         {
46             // Kijkt of het nummer niet lager dan 0 is
47             if ( $aantal > 0 )
48             {
49                 $this->aantalInVoorraad += $aantal;
50             }
51         }
52     }
53
54 }
55
56 // GETTERS & SETTERS
57 public function __get($key)
58 {
59     if( isset($this->$key) )
60     {
61         return $this->$key;
62     }
63 }
64
65 public function toString($key)
66 {
67     if( isset($this->$key) )
68     {
69         return strval($this->$key);
70     }
71 }
72
73 public function __set($key,$val)
74 {
75     $this->change($key,$val);
76 }
77
78 private function change($key,$val)
79 {
80     if( isset($this->$key) )
81     {
82         $this->$key = $val;
83     }
84 }
85
86 /**
87  * Class constructor.
88  */
89
90
91 }
```

## INDEX.php

En om terug te komen op de INDEX.php.  
Dit hiernaast staande code beschrijft hoe het model moet worden aangemaakt met gebruik van de functie.

En hier zie je hoe gegevens gewijzigd kunnen worden en hoe je deze weer kunt ophalen.

```
1
2 $products =make_model("Product");
3 $dvds     =make_model("DVD");
4 $cds      =make_model("CD");
5
6
7 // TEST CODE
8 foreach ($products as $product) {
9     echo "product  : " . $product->__get('naam') . "<br />";
10 }
11
12 foreach ($dvds as $dvd) {
13     echo "dvd      : " . $dvd->__get('filmStudio') . "<br />";
14 }
15
16 foreach ($cds as $cd) {
17     echo "cd       : " . $cd->__get('label') . "<br />";
18 }
19
20 echo "Nieuwe naam voor 2 items <br />";
21 foreach ($products as $product) {
22     echo "product  : " . $product->__get('label') . "<br />";
23     $product->__set("naam", "Nieuwe naam");
24 }
25
26 foreach ($cds as $cd) {
27     echo "cd       : " . $cd->__get('label') . "<br />";
28     $cd->__set("label", "metaalmoeheid");
29 }
30
31
32
```