

# PHP maakt een keuze: de selectie

Huub de Beer

Eindhoven, 4 juni 2011

# Voorbeeld: het punt (I)

```
1 <?php
   $punt = $_POST[ 'punt' ];
3 if ( $punt < 5.5 ) {
4   // $punt is een onvoldoende
5   echo "FAIL!!!";
6 } else {
7   // $punt is een voldoende
8   echo "WIN!!!";
9 };
10 ?>
```

1. de gebruiker voert een 6.7 in

# Voorbeeld: het punt (I)

```
1 <?php
2 $punt = $_POST[ 'punt' ];
3 if ( $punt < 5.5 ) {
4     // $punt is een onvoldoende
5     echo "FAIL!!!";
6 } else {
7     // $punt is een voldoende
8     echo "WIN!!!";
9 };
10 ?>
```

1. de gebruiker voert een 6.7 in
2. PHP controleer of de conditie  $\$punt < 5.5$  geldt. In dit geval niet:  $6.7 \geq 5.5$ .

# Voorbeeld: het punt (I)

```
1 <?php
2 $punt = $_POST[ 'punt' ];
3 if ( $punt < 5.5 ) {
4     // $punt is een onvoldoende
5     echo "FAIL!!!";
6 } else {
7     // $punt is een voldoende
8     echo "WIN!!!";
9 };
10 ?>
```

1. de gebruiker voert een 6.7 in
2. PHP controleer of de conditie  $\$punt < 5.5$  geldt. In dit geval niet:  $6.7 \geq 5.5$ .
3. PHP slaat het eerste blok over, dat hoort bij het geval dat de conditie *wel* geldt.
4. PHP komt bij het geval *anders*, dat geldt.

# Voorbeeld: het punt (I)

```
1 <?php
2 $punt = $_POST[ 'punt' ];
3 if ( $punt < 5.5 ) {
4     // $punt is een onvoldoende
5     echo "FAIL!!!";
6 } else {
7     // $punt is een voldoende
8     echo "WIN!!!";
9 };
10 ?>
```

1. de gebruiker voert een 6.7 in
2. PHP controleer of de conditie  $\$punt < 5.5$  geldt. In dit geval niet:  $6.7 \geq 5.5$ .
3. PHP slaat het eerste blok over, dat hoort bij het geval dat de conditie *wel* geldt.
4. PHP komt bij het geval *anders*, dat geldt.
5. PHP voert het bijbehorende blok uit: print *WIN!!!*

# Voorbeeld: het punt (I)

```
1 <?php
2 $punt = $_POST[ 'punt' ];
3 if ( $punt < 5.5 ) {
4     // $punt is een onvoldoende
5     echo "FAIL!!!";
6 } else {
7     // $punt is een voldoende
8     echo "WIN!!!";
9 };
10 ?>
```

1. de gebruiker voert een 6.7 in
2. PHP controleer of de conditie  $\$punt < 5.5$  geldt. In dit geval niet:  $6.7 \geq 5.5$ .
3. PHP slaat het eerste blok over, dat hoort bij het geval dat de conditie *wel* geldt.
4. PHP komt bij het geval *anders*, dat geldt.
5. PHP voert het bijbehorende blok uit: print *WIN!!!*
6. Na het laatste blok eindigt het if-statement.

# if-statement (I)

- ▶ PHP voert *conditioneel* een stuk code wel/niet uit
- ▶ zo'n stuk code, omsloten door { en } heet een **blok**
- ▶ de inhoud van een blok wordt een niveau ingesprongen
- ▶ een conditie is een Boolean expressie

## patroon if-statement

```
1 if ( conditie ) {  
2   // conditie geldt  
3   // Blok A  
4 } else {  
5   // conditie geldt *niet*  
6   // Blok B  
7 };
```

## Voorbeeld: het punt (II)

```
1 <?php
2 $punt = $_POST[ 'punt' ];
3
4 // $punt -> beoordeling:
5 if ( $punt <= 4 ) {
6     echo "slecht";
7 } elseif ( $punt < 5.5 ) {
8     // $punt tussen 4 en 5.5
9     echo "onvoldoende";
10 } elseif ( $punt < 7 ) {
11     // $punt tussen 5.5 en 7
12     echo "voldoende";
13 } elseif ( $punt < 9 ) {
14     // $punt tussen 7 en 9
15     echo "goed";
16 } else {
17     // $punt tussen 9 en 10:
18     echo "uitmuntend";
19 };
20 ?>
```

- ▶ een if-statement met meerdere condities door middel van **elseif**
- ▶ PHP gaat ze een voor een af, van boven naar beneden
- ▶ PHP voert alleen het blok van de eerst geldende conditie uit
- ▶ geldt geen enkele conditie: PHP voert het blok na **else** uit.



# Het switch-statement

## patroon switch-statement

```
1 switch ( $var ) {  
2   case waarde_1:  
3     echo '$var is gelijk aan waarde_1';  
4     break;  
5   case waarde_2:  
6     echo '$var is gelijk aan waarde_2';  
7     break;  
8   // ...  
9   case waarde_N:  
10    echo '$var is gelijk aan waarde_N';  
11  default:  
12    // In alle andere gevallen  
13    echo 'geen enkele waarde is  
14    gelijk aan de variabele';  
15  };
```

- ▶ \$var heeft waarde uit kleine verzameling
- ▶ bijvoorbeeld, ogen dobbelsteen, kaarten spel, alfabet, ...
- ▶ (alle) waarden zijn integer of string
- ▶ alle gevallen (**cases**) die gelden, worden uitgevoerd.
- ▶ gebruik **break** om dat te voorkomen

## Voorbeeld: het punt (III)

```
1 <?php
2 $beoordeling = $_POST[ 'beoordeling' ];
3 switch ( $beoordeling ) {
4   case "slecht":
5     $punt = 2;
6     break;
7   case "onvoldoende":
8     $punt = 4;
9     break;
10  case "voldoende":
11    $punt = 6;
12    break;
13  case "goed":
14    $punt = 8;
15    break;
16  case "uitmuntend":
17    $punt = "9";
18    break;
19  default:
20    // In alle andere gevallen
21    $punt = 1;
22 };
```