2.1 Strings en quotes

In deze les gaan we nog wat meer kijken naar strings en het verschil tussen " en ' (double quotes en single quotes).

Wat is een string?

Een string is een rijtje van karakters. Deze karakters kunnen van alles zijn: letters, cijfers, leestekens, spaties, ...

Als je een string maakt in PHP (en andere programmeertalen) dan moet je quotes gebruiken om aan te geven dat je een string bedoeld. Als je dat niet doet dan zou PHP niet weten wat je bedoeld:

Bijvoorbeeld echo en "echo" zijn twee verschillende dingen! Zonder quotes is het een commando en met quotes is het string.

Nog een voorbeeld $naam is wat anders dan "$naam". De eerste is de variabele en de tweede is een string.

Dus:

```
echo $naam;
```

is wat anders als

```
echo '$naam';
```

en

```
echo echo;
```

is wat anders als

```
echo "echo";
```

Probeer het maar!

Quotes

Een string hoort dus tussen quotes, maar wat is het verschil tussen double quotes en singel quotes?

Single quotes en escape

Dit is de meeste eenvoudige vorm. Een string tussen single quotes is gewoon een string en alles wat tussen de singel quotes staat is onderdeel van de string. Er is dan wel een klein probleempje: namelijk hoe maak je een string met een single quote?

Probeer maar: `echo 'Dit is een ' single quote'` werkt niet!

Dat komt omdat PHP denkt dat de string wordt afgesloten door de tweede quote.

Er geldt dus een uitzonderingsregel voor het maken van een string met een single quote. En die regel is dat als een singel quote onderdeel uitmaakt van een string je de singel quote moet 'escapen' en dat doe je als volgt:

```
echo 'Dit is een \' single quote.';


echo 'I\'ll be back!';
```

Double quotes

Double quotes zijn eigenlijk ingewikkelder omdat veel zaken die je tussen double quotes intypt worden gerenderd. Renderen betekent 'omzetten'. Als je bijvoorbeeld een variabele in een string tussen double quotes zet dan wordt niet de variabele naam, maar de waarde van de variabele afgedrukt.

Probeer de volgende code maar eens uit en zie wat het verschil is.

```
$naam="The Big Boss";
echo "Who is $naam";
echo "<br>";
echo 'Who is $naam';
```

Er zijn nog een paar zaken die tussen double quotes worden gerendered maar die zijn voor webtoepassingen niet zo belangrijk. Zo wordt een \n een nieuwe regel en \t een tab.
Als je een dubbele quote of een enkel quote wilt afdrukken moet je er een \ voor plaatsten en voor een backslash moet je zelf ook weer een \ plaatsten.

Dus samengevat:

expressie	tussen double quotes resulteert in	tussen singel quotes resulteert in
\"	"	"
\'	'	'
\\	\	\

$naam	de waarde van de variabele $naam	$naam
\n	nieuwe regel	\n
\t	tab	\t

Heredoc

In PHP zit een extra hulpje om grote blokken text (bijvoorbeeld html code) op een eenvoudige manier aan een variabele toe te kennen. Stel je wilt een tabel in een variabele zetten, dan kan dat op de volgende manier.

```
$html = '':
$html .= '<table><tr>';
$html .= '<td> kolom 1 </td>';
$html .= '<td> kolom 2 </td>';
$html .= '</tr><tr>';
$html .= '<td> kolom 1 </td>';
$html .= '<td> kolom 2 </td>';
$html .= '</tr><table>';

echo $html;
```

Dat werkt en ziet er best overzichtelijk uit, maar het kan ook zo:

```
$html = <<< END_HTML
<table>
<tr>
    <td> kolom 1 </td>
        <td> kolom 1 </td>
    </tr><tr>
        <td> kolom 1 </td>
        <td> kolom 1 </td>
    </tr>
</table>
END_HTML;

echo $html;
```

De END_HTML is een willekeurig gekozen identifier. Je mag elk woord gebruiken, maar het moet wel aan het begin van de regel staan anders werkt het niet als markering van het einde.

Revision #11
Created 2020-02-01 10:26:51 UTC by Admin
Updated 2021-12-17 23:39:53 UTC by Max