

PHP Variablen

Die Variablen in PHP sind zu mindestens bei der Initialisierung Typen undefiniert. Es ist so, als würde man unter C# ausschließlich mit dem var arbeiten.

Weiter kann die Variable entweder local, global oder static sein. Ist eine Variable außerhalb von Scopes {} definiert, so ist diese global. Kann aber nicht in den Scopes verwendet werden.

Wird eine Variable in den Scopes definiert, so gilt sie als local und kann nur in den Scopes verwendet werden.

Befindet sich eine Variable in den Scopes und ist diese als static deklariert, so behält sie ihren Wert, auch wenn der Programmcode die Scopes bereits abgeschlossen hat.

```
[crayon-67678a86af32b905372296/]
```

Das Resultat wäre so eben 1 2 3. Hätte man hier nicht static genutzt, wäre das Ergebnis immer 1, da die Variable, wie bereits angesprochen gelöscht wird, sobald sie die Scopes verlässt.

Interessant sind auch die anonymen Methoden unter PHP. Man hat die Möglichkeit die Funktion quasi in eine Variable zu stecken. Nimmt man das obere Beispiel, so kann man einfach

```
[crayon-67678a86af333697058474/]
```

dazu verwenden die Funktion zu anonymisieren.

Variablentypen

PHP hat weit weniger Typen als z.B. noch C#.

Während der boolean, integer, string und null gleich sind, haben folgende Typen so ihre Besonderheiten.

Eine numerische Zahl mit Nachkommastellen ist zwar ein **float**,

fragt man aber mit

[crayon-67678a86af336118710446/]

den Variablentypen ab, sagt PHP, dass es ein **double** ist.

Der **array** in PHP darf alle Möglichen Typen enthalten und erfüllt sogar die Funktion einer aus C# bekannten List oder Diktionary mit Key and Value Prinzip.

Betrachtet man das manuel von PHP.net, so findet man auch eigenartige Funktionen um sogar die im array enthaltenen Werte zu summieren, zu mischen usw.

Man kann in PHP Funktionen und sogar Klassen in eine Variable oder als Variable? speichern. Der Typ ist dann ein object.

[crayon-67678a86af338396290907/]

Wie auch in C# kann man jeden anderen Typ in ein object konvertieren, nennt sich hier aber nicht casten... Dies geschieht aber im Prinzip genau so:

[crayon-67678a86af33a785786561/]

Als letzter Type ist die Resource, welche quasi einen Zeiger auf externe Ressourcen zeigt. Diese beinhaltet z.B. das handeln mit Datenbanken, Pdf Dateien usw. Die folgende Liste zeigt einen Auszug davon:

Ressource-List