

Strukturierte Programmierung

- [Einführung](#)
- [VS Code und XAMPP](#)
- [Variablen und Rechenoperationen](#)
- [Datentypen](#)
- [Operatoren](#)
- [Strings](#)
- [Übungen](#)
- [Arrays](#)
- [Übungen zu Arrays](#)
- [Mehrdimensionales Arrays](#)
- [Übungen zu mehrdimensionalen Arrays](#)
- [Bedingung](#)
- [Übungen zu Kontrollstrukturen](#)
- [Struktogramme](#)
- [Schleifen](#)
- [Übungen zu Strings](#)
- [Datenkompression](#)
- [Barcode-Generator](#)
- [Funktionen](#)
- [Übungen zu Funktionen](#)
- [Arbeiten mit Dateien](#)



Hinweis zur verwendeten Literatur:

Die nachfolgend niedergeschriebene Theorie sowie die daran angelehnten Übungsaufgaben basieren auf unterschiedlichen Werken dritter Autoren, die je nach Themengebiet im unterschiedlichen Umfang als Quelle dieses Skriptes dienen. Welche Bücher konkret verwendet wurden, können Sie dem im Anhang befindlichen Literaturverzeichnis entnehmen.

Weithin könnten folgende Internetquellen für das schnelle digitale Recherchieren hilfreich sein:

- Dokumentation von PHP: <http://de2.php.net/>
- Forum inkl. Tutorials: <http://www.selfphp.de/>
- Tutorialseite: <https://www.w3schools.com/pHP/>

Aus <<https://steinam.rigel.uberspace.de/php/strukturiert/toctree.html>>

1.3 Übersetzungsarten

Arbeitsauftrag:

Schauen Sie sich nachfolgende Lernvideos an und beantworten Sie dabei folgende Fragen:

Danke Herr Miller

Quora

<https://de.quora.com/Was-sind-die-Unterschiede-zwischen-einem-Compiler-einem-Interpreter-und-einem-Assembler>

<https://www.youtube.com/embed/F1GLYZ7fhvw>
interpreter vs compiler



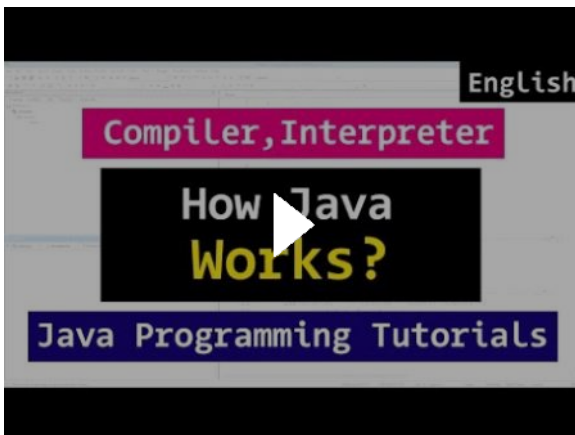
- Um welche Fragestellung handelt es sich hierbei überhaupt? Um was geht es?
- Beschreiben Sie die Eigenschaften beider Programmierarten.
- Finden Sie nun je Übersetzungsart zwei Programmiersprachen, die diese Übersetzungsart verwenden.
- Finden Sie nun auch noch mindestens eine Programmiersprache, die einen zweifachen Übersetzungsprozess hat.

Aus

<https://steinam.rigel.uberspace.de/php/strukturiert/einfuehrung_uebersetzungsarten.html>

<https://www.youtube.com/embed/fhfVvkPplwik>

[How Java Program Works, Compiler, Interpreter | Java Tutorial](#)



Variablen und Rechenoperationen



Wir reisen in das Jahr 1994 zurück. Im Rahmen des sich sukzessive entwickelnden World Wide Webs hat ein junger, vielversprechender Hoffnungsträger folgende scheinbar geniale Idee:

```
<?php
$booktitle= "1984";
$author="George Orwell";
$rate = 9.5;
echo $booktitle;
echo $author;
echo $rate;
```

Gerne zeigen wir Ihnen die zur Verfügung stehenden Bücher. Um Sie persönlich ansprechen zu können, bitte ich Sie jedoch zuerst Ihren Namen anzugeben.

<input type="text"/>	Vorname
<input type="text"/>	Nachname
<input type="button" value="Senden"/>	<input type="button" value="Zurücksetzen"/>



← → ↻ ⓘ localhost/Webg..
Guten Tag, Christoph Zobel
Buch des Tages:
1984

XSS(Cross-site scripting)-Gefahr: Immer dann, wenn Daten vom Client zum Server übertragen werden, sollten diese hinsichtlich ihrer Existenz sowie ihrer möglichen Manipulation hin überprüft werden. Aus didaktischen Gründen verzichten wir *an dieser Stelle* auf einen entsprechenden Exkurs.

Weitere Informationen zu den vielfältigen Gestaltungsmöglichkeiten von Formularen htmlseitig entnehmen Sie bitte nachfolgenden Link.

Möglichkeiten von
Formularen

https://wiki.selfhtml.org/wiki/HTML/Tutorials/Formulare_erstellen_und_gestalten

Informationstext zu Variablen:

Bitte lesen Sie sich nun nachfolgenden Informationstext durch und *markieren* Sie sich die Ihrer Meinung nach wichtigen Stellen.

Info

PHP-Variablen beginnen mit einem \$-Zeichen, gefolgt von einem Variablennamen. Ein gültiger Variablenname beginnt mit einem Buchstaben oder einem Unterstrich, gefolgt von einer Kombination aus Buchstaben, Ziffern und oder Unterstrichen. Variablennamen sind *case-sensitive*, d.h. Groß- und Kleinschreibung machen einen Unterschied. Alles nach dem // ist ein Kommentar und wird ignoriert.

Q-1: Welche Variablennamen sind korrekt?

- ☐ A. \$_some_value = ,abc'
- ☐ B. \$1number = 12.3
- ☐ C. \$some\$signs% = ,&^%'
- ☐ D. \$go_2_home = „ok“
- ☐ E. \$go_2_Home = „ok“
- ☐ F. \$isThisCamelCase = true

Check Me

Datentypen

PHP ist schwach typisierend. Das heißt, dass der Datentyp einer Variablen automatisch von PHP zugewiesen wird. Hierzu „untersucht“ PHP den Wert, welcher der Variablen zugewiesen wird und vergibt entsprechend automatisch den vermeintlich passenden Datentyp.

PHP hat 8 primitive Datentypen, aber zurzeit konzentrieren wir uns auf die vier wichtigsten:

- Boolean: Sie besitzen entweder true oder false als Wert.
- Integer: Dies sind numerische Werte ohne Dezimalstellen, z.B. 2 oder 5
- Float: Fließkommazahlen mit einer Dezimalstelle, z.B. 2.3
- Strings: aneinandergefügte Zeichen, die entweder von einem “ oder einem ‘, umschlossen sind, z.B. ‘dies’ oder „das“.

Auch wenn PHP diese Datentypen besitzt, erlaubt es dennoch die Zuweisung verschiedener Typen an die gleiche Variable. Betrachten Sie den folgenden Code:

Quellcode

Ergebnis

Erklärung

```
<?php
$number = 123;
var_dump($number);
$number = 'abc';
var_dump($number);
```

Aufgabe:

Geben Sie nachfolgenden Code in der Konsole ein und finden Sie mit Ihrem Banknachbarn zusammen eine Erklärung für das Ergebnis:

Quellcode

Ergebnis

Erklärung

```
<?php  
  
$a = "1";  
$b = 2;  
var_dump($a + $b);  
var_dump($a . $b);
```

Operatoren

Damit Variablen miteinander agieren können, benötigen Sie sogenannte Operatoren. Nachfolgend eine kleine Auswahl.

Arithmetische Operatoren

Arithmetische Operatoren sind leicht verständlich. Addition, Subtraktion, Multiplikation und Division (+, -, */) funktionieren wie ihr Name es sagt. Modulo (%) gibt den Restwert einer Division zurück; der Exponent nimmt den ersten Operanden hoch den 2. Operanden. Die Negation versetzt den Operanden in das Negative; dies ist auch der einzige Operator, der nur einen Operanden besitzt.

```
<?php

$a = 10;
$b = 3;
var_dump($a + $b); // 13
var_dump($a - $b); // 7
var_dump($a * $b); // 30
var_dump($a / $b); // 3.333333...
var_dump($a % $b); // 1
var_dump($a ** $b); // 1000
var_dump(-$a); // -10
```

Vergleichsoperatoren

Vergleichsoperatoren werden am häufigsten genutzt. Sie nehmen zwei Operanden und vergleichen diese. Das Ergebnis des Vergleichs wird üblicherweise als boolean zurückgegeben. Ein Beispiel:

```
<?php

var_dump(2 < 3); // true
var_dump(3 < 3); // false
var_dump(3 <= 3); // true
var_dump(4 <= 3); // false
var_dump(2 > 3); // false
var_dump(3 >= 3); // true
var_dump(3 > 3); // false
```

Zuweisungsoperatoren

Dieser wurden bereits vorher genutzt; er weist das Ergebnis eines Ausdrucks einer Variablen zu.

```
<?php

$a = 3 + 4 + 5 - 2;
var_dump($a); // 10
```

Es gibt eine Reihe von verkürzten Zuweisungsoperatoren. Man erstellt sie durch die Kombination eines arithmetischen mit einem Zuweisungsoperator. Einige Beispiele:

```
<?php

$a = 13;
$a += 14; // same as $a = $a + 14;
var_dump($a);

$a -= 2; // same as $a = $a - 2;
var_dump($a);

$a *= 4; // same as $a = $a * 4;
var_dump($a);
```

Logische Operatoren

Diese vergleichen logische Operationen, also Aussagen wie True oder False miteinander; der Rückgabewert ist ein boolescher Wert. Die meistverwendeten Operatoren sind " (not), && (and) und || (oder)

```
<?php

var_dump(true && true);
var_dump(true && false);
var_dump(true || false);
var_dump(false || false);
var_dump(!false);
```

Show

Increment- / Decrement- Operatoren

Dies sind abkürzende Schreibweisen wie += bzw. -=. sie werden nur mit Variablen verwendet. Sie benötigen besondere Beachtung, weil sie sich nicht immer intuitiv verhalten:

++: Links von der Variable wird der Wert der Variable um eins erhöht und dann als Ergebnis weiterverarbeitet. Rechts von der Variable eingesetzt, wird zunächst der Wert der Variable zurückgegeben und dann um eins erhöht.

–: Wie bei ++, nur wird der Wert der Variable jeweils um eins vermindert.

Beispiel:

```
<?php
$a = 3;

$b = $a++; // $b is 3, $a is 4

var_dump($a, $b);

$b = ++$a; // $a and $b are 5

var_dump($a, $b);

var_dump($g);
```

Q-1: In welcher Reihenfolge werden die verketteten Operatoren abgearbeitet. Informieren Sie sich zum Thema

Operatorenrangfolge (operator precedence)

- ☒ A. ((5*3) > 10) and ((4+6) == 11)
- ☐ B. (5*(3 > 10)) and (4 + (6 == 11))
- ☐ C. (((((5*3) > 10) and 4)+6) == 11
- ☐ D. ((5*3) > (10 and (4+6))) == 11

Check Me

Strings

Strings sind alphanumerische Zeichenketten, d.h. sie enthalten entweder Ziffern, Buchstaben bzw. Sonderzeichen. Der Programmierer hat relativ häufig mit Strings zu tun; PHP stellt dafür eine ganze Menge an Funktionen zur Verfügung, die man unter <http://php.net/manual/en/ref.strings.php> betrachten kann. Wir werden nur die am häufigsten verwendeten Funktionen ansprechen.

```
<?php
$text = ' How can a clam cram in a clean cream can? ';
echo strlen($text); // 45
$text = trim($text); echo $text; // How can a clam cram
in a clean cream can?
echo strtoupper($text); // HOW CAN A CLAM CRAM IN A CLE
AN CREAM CAN?
echo strtolower($text); // how can a clam cram in a cle
an cream can?

$text = str_replace('can', 'could', $text);
echo $text; // How could a clam cram in a clean cream c
ould?
echo substr($text, 2, 6); // w coul

var_dump(strpos($text, 'can')); // false
var_dump(strpos($text, 'could')); // 4
```

Im obigen Beispiel wird ein String mit verschiedenen Funktionen bearbeitet:

- **strlen**: Diese Funktion gibt die Anzahl der Zeichen zurück, die ein String enthält.
- **trim**: Diese Funktion nimmt alle Leerzeichen links und rechts vom String weg und gibt ihn anschließend zurück
- **strtoupper** / **strtolower**: Diese Funktionen wandeln alle Zeichen eines Strings in Groß- oder Kleinbuchstaben um
- **str_replace**: Diese Funktion ersetzt alle Vorkommen eines Teilsstrings im betrachteten String durch einen anderen String (Suchen/Ersetzen)
- **substr**: Diese Funktion holt aus einem String eine Zeichenkette heraus, die durch die übergebenen Positionen definiert sind.
- **strpos**: Diese Funktion zeigt die Position der ersten Fundstelle einer zu suchenden Zeichenkette innerhalb eines Strings.

Zusätzlich gibt es einen String-Operator (.), der zwei Strings zusammenfügt.

Strings können weiterhin durch einzelne oder doppelte Anführungszeichen umschlossen werden. Der Unterschied besteht darin, dass mit einfachen Anführungszeichen ein String so dargestellt wird, wie er ist. Mit doppelten Anführungszeichen hingegen werden Escape-Zeichen sowie Variablen innerhalb des Strings unterschiedlich behandelt.

Fragen an Dr. Simpson (jun.)

F: Wie kann ich erkennen, welchen Datentyp php einer Variablen zugewiesen hat?

Show

F: Ist es auch möglich, die Typzuweisung an eine Variable zu erzwingen?

Show

F: Im Text stand, dass PHP schwach typisierend ist. Gibt es folglich auch stark typisierende Sprachen?

Show

F: Was ist denn der Unterschied bei einem Vergleich zwischen == und ===?

Show

Übungen

1. Informieren Sie sich in Zweiergruppen über die unterschiedlichen Programmiersprachenkonzepte **Interpreter / Compiler / Bytecompiler**.
 - Stellen Sie die jeweiligen Vor- und Nachteile sowie die Einsatzgebiete mit Hilfe weniger Folien dar.
 - Fügen sie weiterführende Links hinzu.
 - Speichern Sie ihre Ergebnisse auf dem T-Laufwerk ab
2. Erstellen Sie eine Anwendung, bei der eine von Nutzer in einen Formular eingegebene Ganzzahl mit sich selbst multipliziert und anschließend am Bildschirm ausgibt.
3. In einem Programm müssen die Inhalte der Variablen \$v1 und \$v2 getauscht werden. Ordnen Sie die folgenden Anweisungen in die richtige Reihenfolge, in dem Sie 1,2,3 passend vor der Anweisung notieren.

Q-1:

Drag from here Drop blocks here

1 \$temp=\$v2

2 \$v2=\$v1

3 \$v1=\$temp

Check Me Reset

4. Simulieren Sie nun die Anmeldemaske einer Kartenreservierung für ein Kino. Neben einem Begrüßungstext soll der Nutzer aufgefordert werden, seinen Vornamen, Nachnamen sowie die Anzahl der gewünschten Karten nacheinander einzugeben. Alle diese Informationen werden auf der Serverseite mit PHP anschließend übersichtlich noch einmal ausgegeben werden. Geben Sie zusätzlich den Gesamtpreis der Reservierung aus, wenn eine Karte 8 € kostet.
5. Erstellen Sie mit folgenden Angaben ein PHP-Skript.

Variable	Bezeichnung	Variable	Preis (netto)
\$bezeichnung_tisch	Schreibtisch	\$preis_tisch	1999.00 €
\$bezeichnung_stuhl	Bürostuhl	\$preis_stuhl	589.00 €
\$bezeichnung_lampe	Lampe	\$preis_lampe	29.00 €
\$bezeichnung_pctisch	Computertisch	\$preis_pctisch	999.00 €

6. Berechnen Sie den Gesamtpreis (\$netto_gesamt) der eingekauften Artikel, indem Sie die zuvor von Ihnen angelegten Variablenwerte addieren lassen.
7. Berechnen Sie für den gerade berechneten Gesamtpreis den Bruttopreis (\$brutto_gesamt) mithilfe einer Konstanten namens MWST. Recherchieren Sie hierzu, wie man eine Konstante in PHP erzeugt. Der Mehrwertsteuersatz, der zur Berechnung verwendet wird, beträgt 19 %. Die verwendete Zeichenkette für die Währung Euro stellen Sie bitte ebenfalls über eine Konstante (EURO) bereit.
8. Berechnen Sie zusätzlich die Bruttopreise aller Artikel.
9. Lassen Sie alle errechneten Werte in verständlicher Form mit Beschriftungen anzeigen.



Lösungsvorschlag „bueno.php“

10. Mit der Widmark-Formel können wir die Blutalkoholkonzentration abschätzen. Sie wird mit folgender Formel berechnet:

$$c = \frac{A}{m * r}$$

mit

$$A = V * \epsilon * \rho$$

wobei

- c : Alkoholkonzentration im Blut in [g/kg]
- A : Aufgenommene Masse des Alkohols in [g]
- r : Verteilungsfaktor im Körper – Männer: $r \approx 0,7$ – Frauen: $r \approx 0,6$ – Kinder: $r \approx 0,8$
- m : Masse der Person in [kg]
- V : Volumen des Getränks in [ml]
- ϵ : Alkoholvolumenanteil in [%] (z.B. Bier $\approx 0,05$)
- ρ : Dichte von Alkohol [g/ml] $\rightarrow \rho \approx 0,8$ g/ml

Aufgabenstellung:

Schreiben Sie ein Programm, das die Blutalkoholkonzentration mit der Widmark-Formel berechnet. Die zur Berechnung notwendigen Daten sollen dabei vom Nutzer im Rahmen einer Formulareingabe ermittelt werden können. Das Ergebnis soll mit einen von Ihnen frei wählbaren situativen Hinweis versehen werden.

Aus <https://steinam.rigel.uberspace.de/php/strukturiert/uebung_var_strings.html>