

# PHP (Syntaxregeln – Zusammenfassung)

## ***Kommentare***

Syntax:

*<?php*

*// Variablendefinition*

*/\**

*Mehrzeiliger  
Kommentar...*

*\*/*

*?>*

Anmerkung:

- Nach dem Schrägstrich von einzeiligen Kommentaren können Leerzeichen und Tabulatoren gesetzt werden.
- Innerhalb von mehrzeiligen Kommentaren können sowohl Leerzeichen, Tabulatoren als auch Zeilenumbrüche gesetzt werden.
- Innerhalb von Kommentaren sind Sonderzeichen und Umlaute zulässig!

## Operatoren

Syntax:

`<?php`

```
// Operatoren
$spruch = "Hey" . "du!";

// Ausgabe der Variablen
echo $spruch;

// Operatoren - syntaktisch korrekt
// Jedoch nicht zu empfehlen
$spruch = "Hey"
    .
    "du!";

// Ausgabe der Variablen
echo $spruch;
```

`?>`

Anmerkung:

- Vor und Nach Operatoren, wie z.B. +, -, /, \*, %, =, ==, !=, <, >, <=, >=, etc. sind Leerzeichen, Tabulatoren und Zeilenumbrüche zulässig! Diese Aussage gilt für sämtliche Operatoren.

## Variablen

Syntax:

`<?php`

```
// Variablendefinition  
// variablenname = variablenwert;
```

```
// Beispiel  
$vorname = "Matthias";
```

```
// Beispiel - syntaktisch Korrekt  
// Jedoch nicht zu empfehlen!  
$vorname  
=  
"Matthias";
```

```
// Beispiel - syntaktisch Korrekt  
// Jedoch nicht zu empfehlen!  
$vorname = "  
Matthias  
";
```

`?>`

Anmerkung:

- Vor und Nach dem *variablennamen* können Leerzeichen, Tabulatoren und Zeilenumbrüche gesetzt werden!
- Vor und Nach dem Gleichheitszeichen = (Zuweisungsoperator) können Leerzeichen, Tabulatoren und Zeilenumbrüche gesetzt werden. Diese können zur Lesbarkeit des Quelltext beitragen.
- Nach dem Semikolon (Punktstrich) können Leerzeichen, Tabulatoren und Zeilenumbrüche gesetzt werden.
- Zeilenumbrüche innerhalb von Zeichenketten (strings) sind zulässig!

## **Aufruf von Benutzerdefinierten- bzw. Vordefinierten Methoden und Funktionen**

Syntax:

`<?php`

```
// Methode (print())  
print("Hallo Welt!");  
print ("Hallo Welt!");  
print ( "Hallo Welt!" );
```

```
// Methode (printf())  
printf("Hallo Welt!");  
printf ("Hallo Welt!");  
printf ( "Hallo Welt!" );
```

```
// Methode (print()) - syntaktisch Korrekt  
// Jedoch nicht zu empfehlen  
print  
    (  
        "Hallo Welt!"  
    )  
    ;
```

```
// Methode (printf()) - syntaktisch Korrekt  
// Jedoch nicht zu empfehlen  
printf  
    (  
        "Hallo Welt!"  
    )  
    ;
```

`?>`

Anmerkung:

- Beim Aufruf (Auslösen) von Methoden und Funktionen kann man kaum etwas falsch machen. Sowohl Leerzeichen, Tabulatoren und Zeilenumbrüche sind zulässig – nach dem Methoden-/Funktionsnamen und vor und nach den Funktionsklammern. (siehe oben - Syntax-Beispiel)

## **Funktionsdefinition**

Syntax:

`<?php`

```
// Funktionsdefinition
/*
function funktionsname(p1,...,pN)
{
    // Anweisung/Anweisungen
}
*/
```

```
// Funktionsdefinition
function addieren($wert1, $wert2)
{
    return $wert1 + $wert2;
}
```

```
// Funktionsaufruf
print(addieren(10,10));
```

`?>`

Anmerkung:

- Vor und Nach dem Schlüsselwort *function* sind Leerzeichen, Tabulatoren und Zeilenumbrüche zulässig.
- Vor und Nach den Funktionsklammern (...) sind Leerzeichen, Tabulatoren und Zeilenumbrüche zulässig.
- Vor und Nach den Funktionsparametern *p1, p2, ..., pN* sind Leerzeichen, Tabulatoren und Zeilenumbrüche zulässig.
- Vor und Nach den Anweisungsblockklammern {...} sind Leerzeichen, Tabulatoren und Zeilenumbrüche zulässig.

Extrem Beispiel – Funktionsdefinition:

```
// Funktionsdefinition - syntaktisch korrekt
// Jedoch nicht zu empfehlen
function
addieren
(
    $wert1
    ,
    $wert2
)
{
    return $wert1
    +
    $wert2;
}
```

## **Kontrollstrukturen (if-Abfragen) – Aussagen sind auch für – switch-case, for, do-while, while und foreach gültig**

Syntax:

`<?php`

```
// Variable
$vorname = "Matthias";

// Kontrollstruktur (if-Abfrage)
if ($vorname == "Matthias")
{
    print("Hallo, " . $vorname);
}
else
{
    print("Kenn ich nicht!");
}
```

`?>`

Anmerkung:

- Vor und Nach dem Schlüsselwort *if* sind Leerzeichen, Tabulatoren und Zeilenumbrüche zulässig.
- Vor und Nach den Bedingungsklammern (...) sind Leerzeichen, Tabulatoren und Zeilenumbrüche zulässig.
- Vor und Nach den Anweisungsblockklammern {...} sind Leerzeichen, Tabulatoren und Zeilenumbrüche zulässig.
- Diese drei Aussagen sind für alle von JavaScript zur Verfügung gestellten Kontrollstrukturen gültig!

Extrem Beispiel – if-Abfrage:

```
// Kontrollstruktur (if-Abfrage) - syntaktisch korrekt
// Jedoch nicht zu empfehlen
if
    (
        $vorname == "Matthias"
    )
{
    print("Hallo, " . vorname);
}
    else
    {
        print("Kenn ich nicht!");
    }
}
```