

# Anleitungen Schritt-Für-Schritt

Softwareentwicklung

Modulname:	<b>Dynamische Webprogrammierung in PHP</b>
------------	--

## **HINWEIS:**

Dokumentenname = Schritt\_fuer\_Schritt\_Anleitungen\_PHP.odt

Stand: 3. Mai 2015



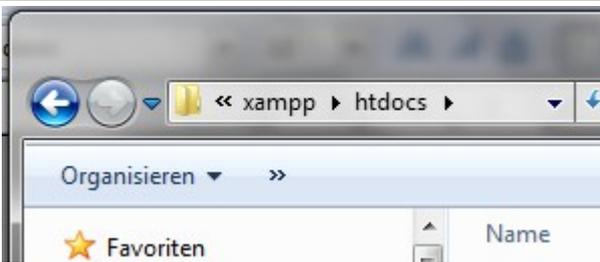
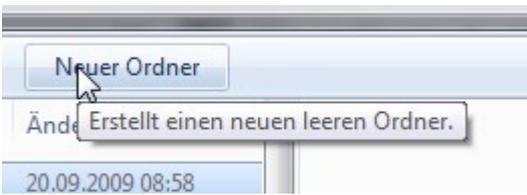
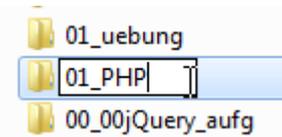
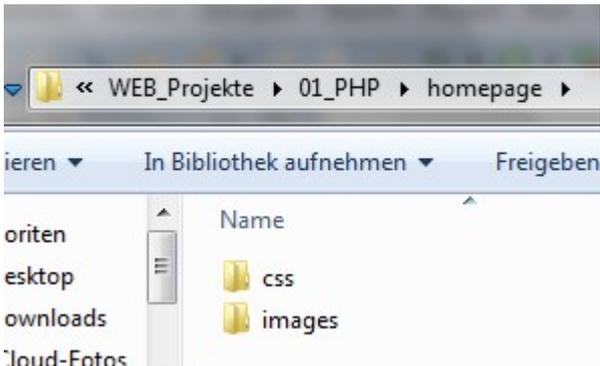
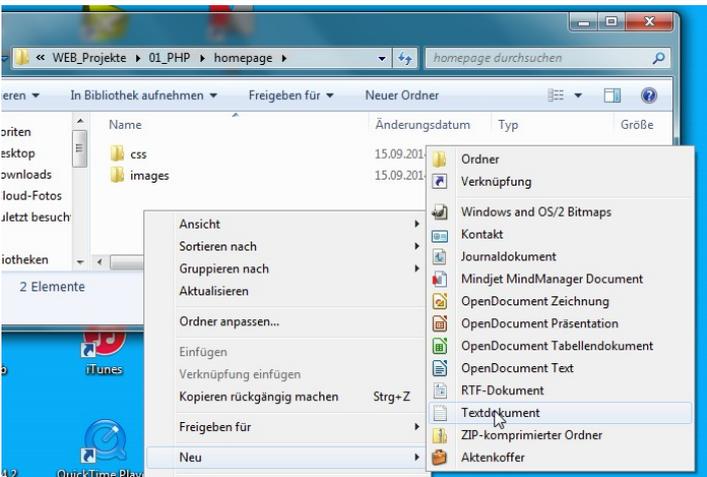
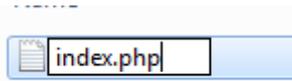
© Christine Janischek

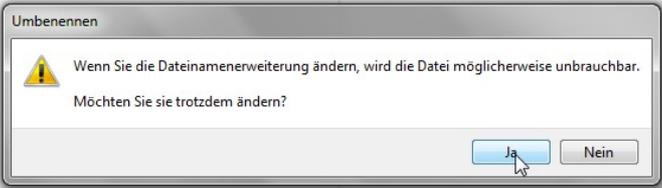
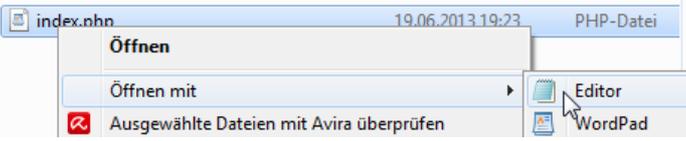
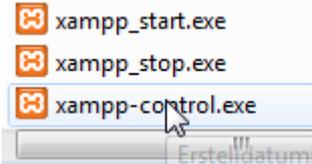
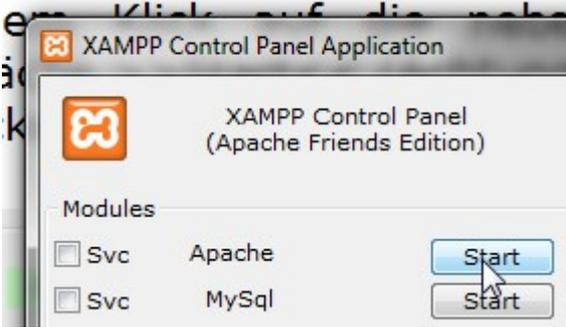
## Inhaltsverzeichnis

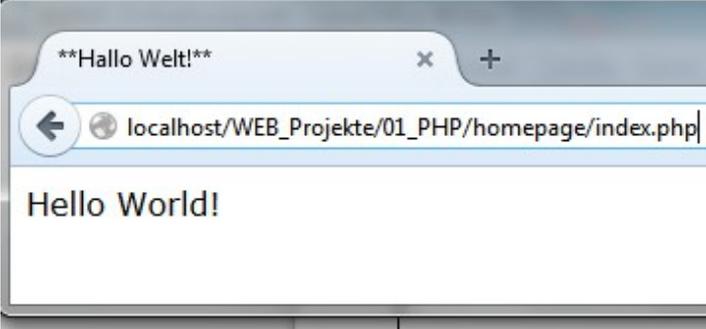
1 Einführung in PHP.....	3
2 Erstellen eines PHP-Grundgerüst.....	6
3 Formulare mit PHP erstellen und auswerten.....	10
4 Umsetzung informatischer Prinzipien im PHP-Quellcode.....	16
5 Praktische Konzepte der objektorientierten Programmierung.....	23
6 Kontrollstrukturen.....	30

## 1 Einführung in PHP

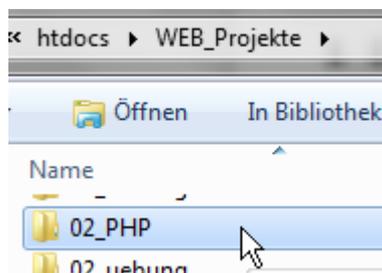
Voraussetzung für die Entwicklung ist das Verzeichnis xampp auf dem Wechseldatenträger (USB-Stick im Wurzelverzeichnis, der ersten Ebene).

	<p>Wechseln Sie in das htdocs-Verzeichnis.</p> <p>Öffnen Sie dazu Ihren USB-Stick und klicken Sie erst auf das Verzeichnis &gt;&gt;xampp&lt;&lt; und dann auf das darin liegende Verzeichnis &gt;&gt;htdocs&lt;&lt;.</p> <p><code>\xampp\htdocs\WEB_Projekte</code></p>
	<p>Klicken Sie dann auf die Schaltfläche &gt;&gt;Neuer Ordner&lt;&lt; und geben Sie als Verzeichnisname &gt;&gt;01_PHP&lt;&lt; an.</p> 
	<p>Öffnen Sie das Verzeichnis mit einem Doppelklick. Erzeugen Sie in diesem Verzeichnis den homepage-Ordner.</p> <p>Kopieren Sie dann aus dem letzten HTML-Projekt das Verzeichnis css und images und fügen Sie es hier ein. Wir benötigen diese Verzeichnisse erst zu einem späteren Zeitpunkt. Wir nehmen Sie aber trotzdem gleich mit!</p>
	<p>Öffnen Sie das Verzeichnis mit einem Doppelklick.</p> <p>Wählen Sie im Kontextmenü (rechte Maustaste) die Option <i>Neu &gt;&gt; Textdokument</i> und ändern Sie den Dateinamen in &gt;&gt;index.php&lt;&lt; ab.</p> 

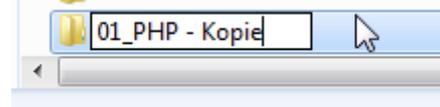
	<p>Bestätigen Sie die Nachfrage nach der Änderung der Dateierweiterung mit einem Klick auf die Schaltfläche &gt;&gt;ja&lt;&lt;.</p>
	<p>Öffnen Sie die Datei mit dem Kontextmenü (rechte Maustaste) mit dem einfachen Texteditor durch einen Klick auf die Option <i>Öffnen mit</i> &gt;&gt; <i>Editor</i>.</p>
	<p>Die Datei ist wie erwartet noch leer.</p> <p>Ergänzen Sie nun den angezeigten Quellcode und Speichern Sie die Veränderung (STRG + S).</p>
	<p>Um das Ergebnis zu Test sind nun einige weitere Schritte notwendig.</p> <p>Öffnen Sie ihr xampp-Verzeichnis. Um das xampp-Control-Panel zu starten klicken Sie die darin enthaltene Datei &gt;&gt;xampp-control.exe&lt;&lt; doppelt an.</p>
	<p>Starten Sie den Webserver &gt;&gt;Apache&lt;&lt; mit einem Klick auf die nebenliegende Schaltfläche &gt;&gt;Start&lt;&lt; (Achtung <b>nur einmal</b> klicken!).</p>  <p>Im Ergebnis muss das grüne Feld dauerhaft sichtbar sein.</p> <p>Hinweis: Falls das grüne Feld nur kurz, bzw. gar nicht erscheint liegt der Fehler meist darin, dass das</p>

	<p>xampp-Verzeichnis nicht auf der obersten Ebene Ihres Datenträgers liegt, oder aber ihre Firewall blockiert den Zugriff.</p> <p>Im ersten Fall müssen Sie das xampp-Verzeichnis verschieben. Im zweiten Fall fragt Windows 7/8 meistens ob der Zugriff erlaubt werden soll bestätigen Sie diese Nachfrage ggf. mit einem Klick auf die Schaltfläche &gt;&gt;zulassen&lt;&lt;.</p>
 A screenshot of a web browser window. The title bar shows a single tab with the text "**Hallo Welt!**". The address bar contains the URL "localhost/WEB_Projekte/01_PHP/homepage/index.php". The main content area of the browser displays the text "Hello World!".	<p>Wenn der Webserver gestartet ist können wir das Ergebnis im Internetbrowser testen.</p> <p>Öffnen Sie dazu den Internetbrowser &gt;&gt;Firefox&lt;&lt; und geben Sie den Pfad zur gerade erstellen Datei an. Geben Sie dazu den in der Grafik angezeigten Pfad ein.</p> <p><b>Herzlichen Glückwunsch Sie haben ihre erste PHP-Seite erstellt.</b></p> <p>Folgen Sie nun der Anleitung zum PHP-Grundgerüst.</p> <p>Hinweis: &gt;&gt;localhost&lt;&lt; steht für die Adresse des lokalen Webservers (APACHE). In der Regel ist das die IP-Adresse 197.0.0.1</p>

## 2 Erstellen eines PHP-Grundgerüst



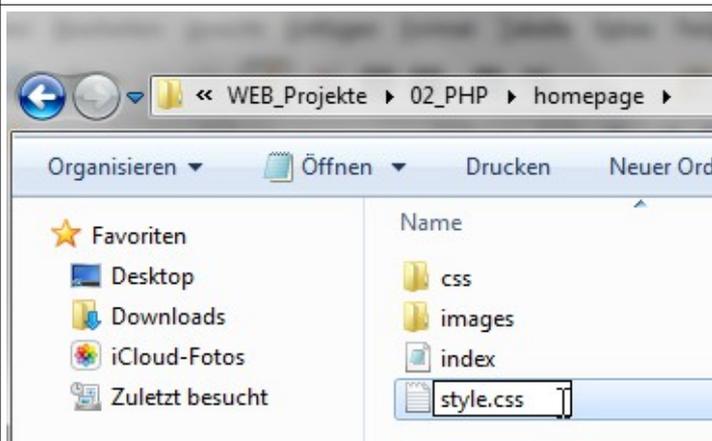
Wechseln Sie wie zuvor in das htdocs-Verzeichnis. Kopieren Sie das gerade fertiggestellte Verzeichnis >>01\_PHP<<:



Und geben Sie dem Verzeichnis den Namen >>02\_PHP<<

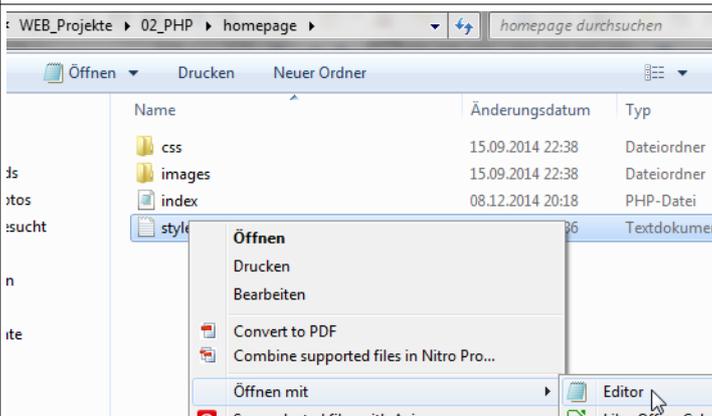
Hinweis:

Falls Sie in Eclipse arbeiten kopieren Sie das Projektverzeichnis 01\_PHP mit STRG + C. Fügen Sie diese mit STRG + V ein und benennen Sie das Verzeichnis wie oben angezeigt. Nehmen Sie nun alle geschilderten Veränderungen in Ihrem Projekt vor.

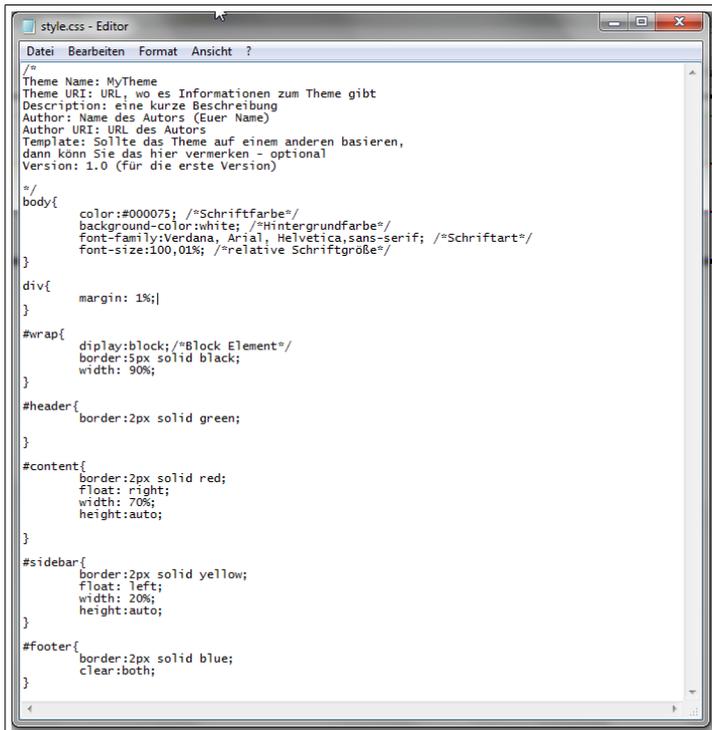


Wir erzeugen in diesem Verzeichnis das Box-Modell für unser Layout (Theme).

Erzeugen Sie dazu die Datei >> style.css<<



Öffnen Sie die Datei >>style.css<< mit einem einfachen Editor. Die Datei ist wie erwartet noch leer!



```

/*
Theme Name: MyTheme
Theme URI: URL, wo es Informationen zum Theme gibt
Description: eine kurze Beschreibung
Author: Name des Autors (Euer Name)
Author URI: URL des Autors
Template: Sollte das Theme auf einem anderen basieren,
dann kann Sie das hier vermerken - optional
Version: 1.0 (Für die erste Version)
*/

body{
    color:#000075; /*Schriftfarbe*/
    background-color:white; /*Hintergrundfarbe*/
    font-family:Verdana, Arial, Helvetica,sans-serif; /*Schriftart*/
    font-size:100,01%; /*relative Schriftgröße*/
}

div{
    margin: 1%;
}

#wrap{
    display:block; /*Block Element*/
    border:5px solid black;
    width: 90%;
}

#header{
    border:2px solid green;
}

#content{
    border:2px solid red;
    float: right;
    width: 70%;
    height:auto;
}

#sidebar{
    border:2px solid yellow;
    float: left;
    width: 20%;
    height:auto;
}

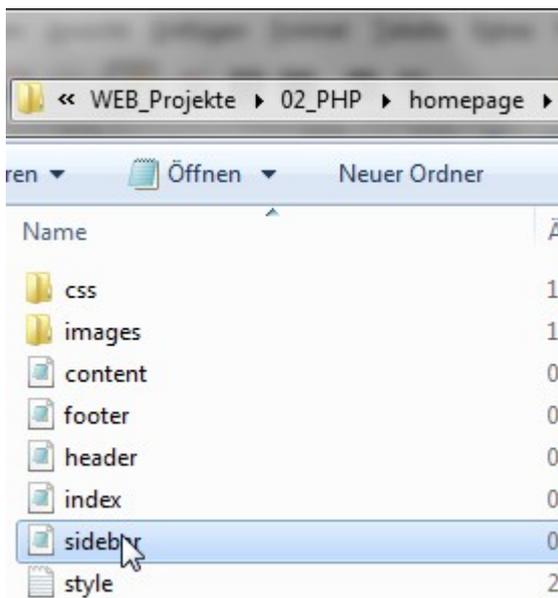
#footer{
    border:2px solid blue;
    clear:both;
}

```

Ergänzen Sie den angezeigten Quellcode für das Box-Modell und speichern Sie die Änderungen.(STRG + S)

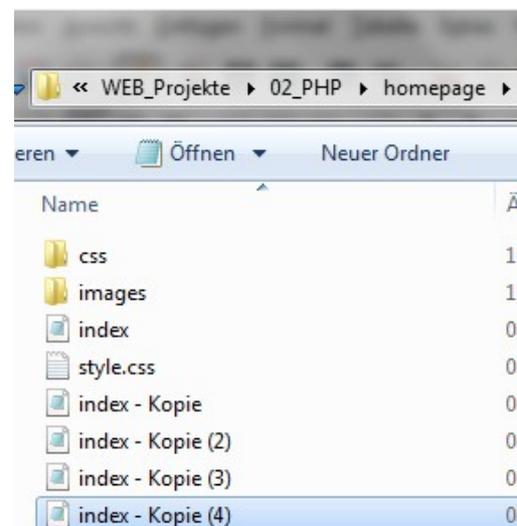
Hinweis:

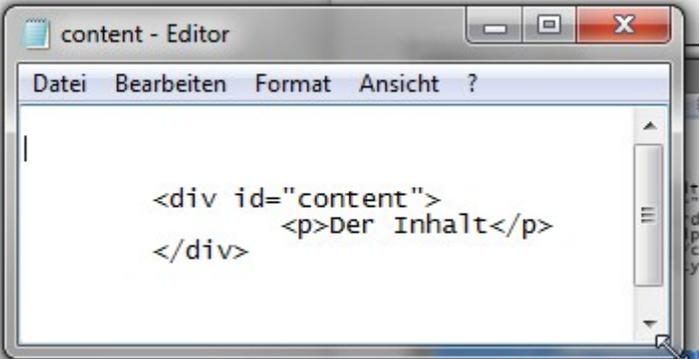
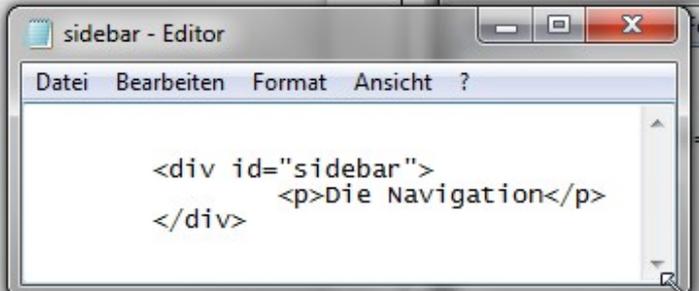
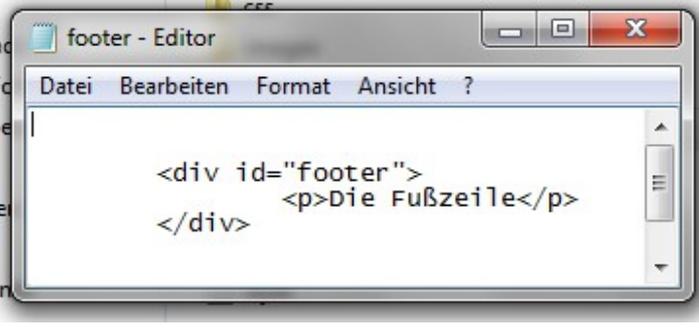
Die wrap-Box wird später den Container bilden der die ganze Seite im Internetbrowser zusammenhält und ausrichtet. Weitere Details folgen zu einem späteren Zeitpunkt.

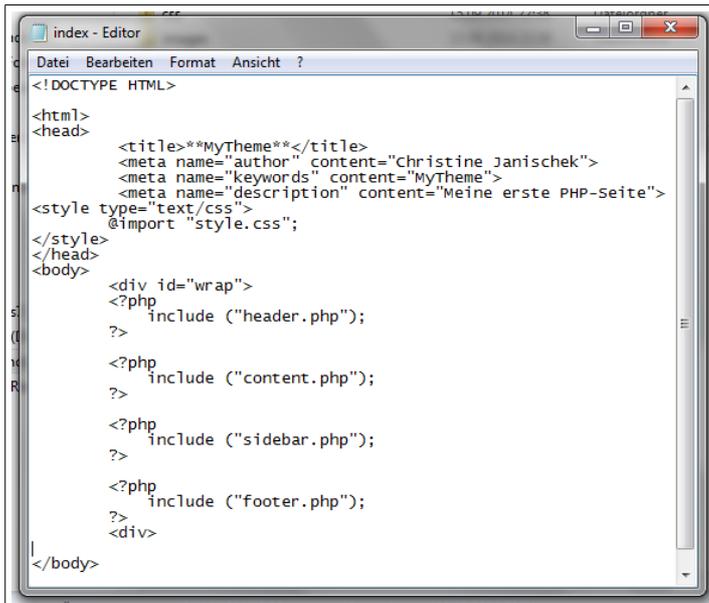


Um das Box-Modell zu nutzen benötigen wir weitere PHP-Dateien für die Kopfzeile (header), den Inhalt (content), die Navigation (sidebar) und die Fußzeile (footer).

Kopieren Sie dazu die Datei index.php vier mal und benennen Sie die Dateien wie in der nebenstehenden Grafik angezeigt um.



 <pre>&lt;div id="header"&gt;   &lt;p&gt;Die Kopfzeile&lt;/p&gt; &lt;/div&gt;</pre>	<p>Öffnen Sie zuerst die Datei &gt;&gt;header.php&lt;&lt; im Editor und reduzieren Sie den Quellcode auf die folgende div-Box:</p> <pre>&lt;div id="header"&gt;   &lt;p&gt;Die Kopfzeile&lt;/p&gt; &lt;/div&gt;</pre> <p>Speichern Sie die Änderungen. (STRG + S)</p>
 <pre>&lt;div id="content"&gt;   &lt;p&gt;Der Inhalt&lt;/p&gt; &lt;/div&gt;</pre>	<p>Ändern Sie auf die gleiche Weise den Quellcode für die übrigen Dateien.</p> <p>Ändern Sie die Datei &gt;&gt;content.php&lt;&lt; wie folgt.</p> <p>Speichern Sie die Änderungen. (STRG + S)</p>
 <pre>&lt;div id="sidebar"&gt;   &lt;p&gt;Die Navigation&lt;/p&gt; &lt;/div&gt;</pre>	<p>Ändern Sie die Datei &gt;&gt;sidebar.php&lt;&lt; wie folgt.</p> <p>Speichern Sie die Änderungen. (STRG + S)</p>
 <pre>&lt;div id="footer"&gt;   &lt;p&gt;Die Fußzeile&lt;/p&gt; &lt;/div&gt;</pre>	<p>Ändern Sie die Datei &gt;&gt;footer.php&lt;&lt; wie folgt.</p> <p>Speichern Sie die Änderungen. (STRG + S)</p>



```

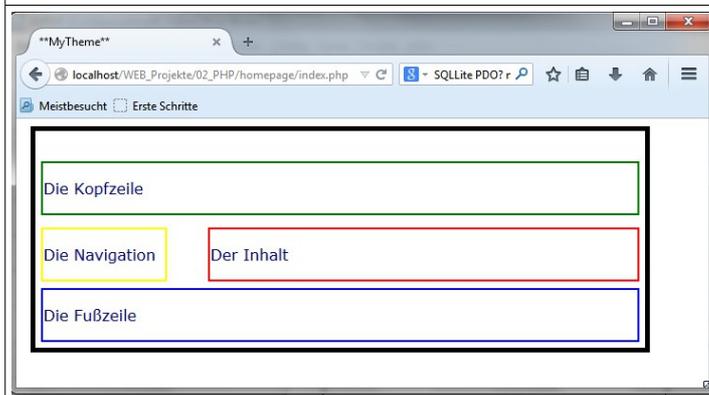
<!DOCTYPE HTML>
<html>
<head>
  <title>**MyTheme**</title>
  <meta name="author" content="Christine Janischek">
  <meta name="keywords" content="MyTheme">
  <meta name="description" content="Meine erste PHP-Seite">
</head>
<style type="text/css">
  @import "style.css";
</style>
</head>
<body>
  <div id="wrap">
    <?php
      include ("header.php");
    ?>
    <?php
      include ("content.php");
    ?>
    <?php
      include ("sidebar.php");
    ?>
    <?php
      include ("footer.php");
    ?>
  </div>
</body>

```

Ein wesentliches Prinzip bei der Erzeugung eines PHP-Layouts ist, dass die Seite beim Aufruf der Startseite (index.php) dynamisch erzeugt wird. Das bedeutet, die Seite wird aus ihren einzelnen Bestandteilen zusammengesetzt.

Damit das geschieht müssen wir die gerade erzeugten und veränderten Dateien in der Datei >>index.php<< vorerst mit Hilfe eines PHP-includes einbetten.

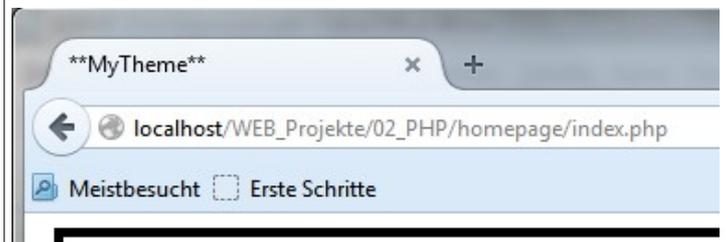
Verändern Sie dazu den Quellcode in dieser Datei wie nebenstehend angezeigt ab.



Wenn der Webserver gestartet ist können wir das Ergebnis im Internetbrowser testen.

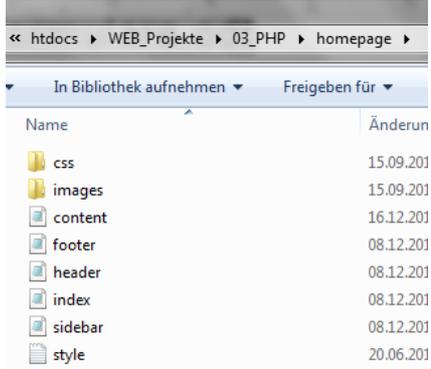
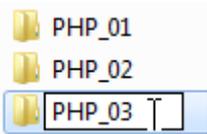
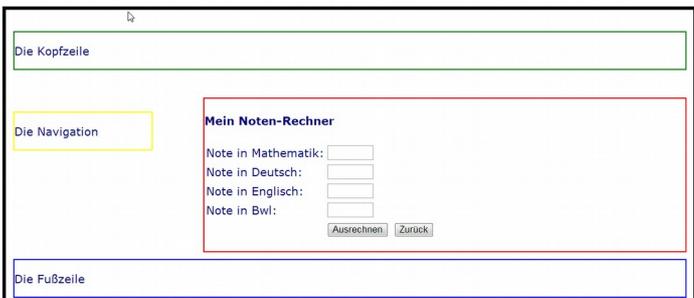
Öffnen Sie dazu den Internetbrowser >>Firefox<< und geben Sie den Pfad zur gerade erstellen Datei an. Geben Sie dazu den in der Grafik angezeigten Pfad ein.

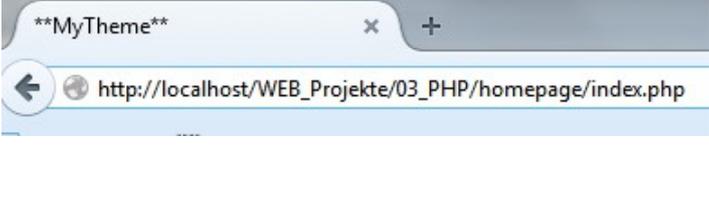
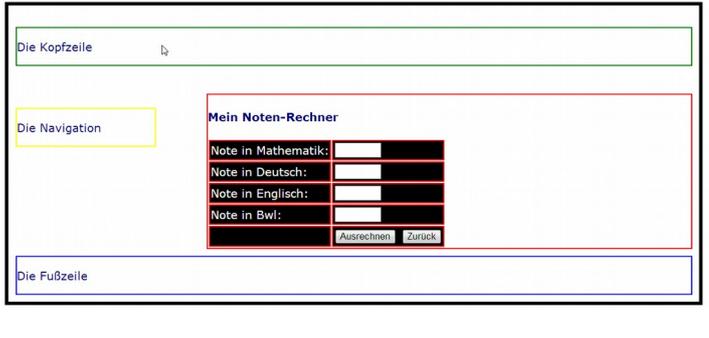
[http://localhost/WEB\\_Projekte/02\\_PHP/homepage/index.php](http://localhost/WEB_Projekte/02_PHP/homepage/index.php)

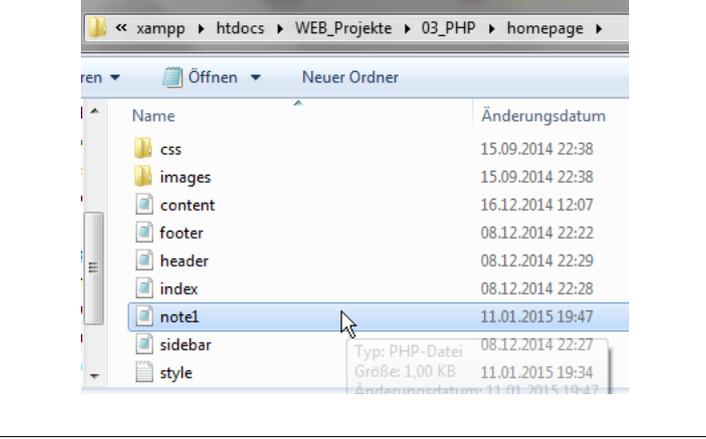
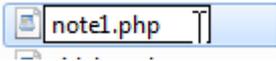
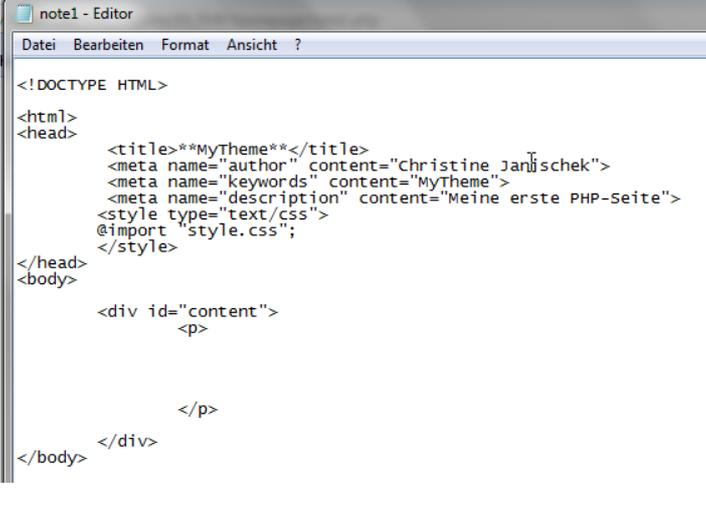


**Herzlichen Glückwunsch Sie haben ihre erstes dynamisches PHP-Layout erstellt.**

### 3 Formulare mit PHP erstellen und auswerten

 <table border="1"> <thead> <tr> <th>Name</th> <th>Änderun</th> </tr> </thead> <tbody> <tr><td>css</td><td>15.09.201</td></tr> <tr><td>images</td><td>15.09.201</td></tr> <tr><td>content</td><td>16.12.201</td></tr> <tr><td>footer</td><td>08.12.201</td></tr> <tr><td>header</td><td>08.12.201</td></tr> <tr><td>index</td><td>08.12.201</td></tr> <tr><td>sidebar</td><td>08.12.201</td></tr> <tr><td>style</td><td>20.06.201</td></tr> </tbody> </table>	Name	Änderun	css	15.09.201	images	15.09.201	content	16.12.201	footer	08.12.201	header	08.12.201	index	08.12.201	sidebar	08.12.201	style	20.06.201	<p>Wechseln Sie wie zuvor in das htdocs-Verzeichnis. Kopieren Sie das gerade fertiggestellte Verzeichnis &gt;&gt; 02_PHP&lt;&lt;:</p>  <p>Und geben Sie dem Verzeichnis den Namen &gt;&gt;03_PHP&lt;&lt;</p> <p>Hinweis: Falls Sie in Eclipse arbeiten kopieren Sie das Projektverzeichnis 01_PHP mit STRG + C. Fügen Sie diese mit STRG + V ein und benennen Sie das Verzeichnis wie oben angezeigt. Nehmen Sie nun alle geschilderten Veränderungen in Ihrem Projekt vor.</p>
Name	Änderun																		
css	15.09.201																		
images	15.09.201																		
content	16.12.201																		
footer	08.12.201																		
header	08.12.201																		
index	08.12.201																		
sidebar	08.12.201																		
style	20.06.201																		
<pre> &lt;div id="content"&gt;   &lt;form action="notel.php" method="post"&gt;     &lt;h4&gt;Mein Noten-Rechner&lt;/h4&gt;     &lt;table&gt;       &lt;tr&gt;         &lt;td&gt;Note in Mathematik:&lt;/td&gt;         &lt;td&gt;&lt;input type="text" name="tfMathe" size="5" maxlength="5" /&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td&gt;Note in Deutsch:&lt;/td&gt;         &lt;td&gt;&lt;input type="text" name="tfDeutsch" size="5" maxlength="5" /&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td&gt;Note in Englisch:&lt;/td&gt;         &lt;td&gt;&lt;input type="text" name="tfEnglisch" size="5" maxlength="5" /&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td&gt;Note in Bwl:&lt;/td&gt;         &lt;td&gt;&lt;input type="text" name="tfBwl" size="5" maxlength="5" /&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td&gt;&lt;input type="submit" value="Ausrechnen" name="durchschnittAusrechnen" /&gt;         &lt;td&gt;&lt;input type="button" value="Zurück" onclick="history.back()" /&gt;&lt;/td&gt;       &lt;/tr&gt;     &lt;/table&gt;   &lt;/form&gt; &lt;/div&gt; </pre>	<p>Der Inhaltsbereich &gt;&gt;content.php&lt;&lt; soll nun um ein einfaches Formular erweitert werden.</p> <p>Öffnen Sie dazu die Datei &gt;&gt;content.php&lt;&lt; mit einem einfachen Editor und ergänzen Sie in der div-Box &gt;&gt;content&lt;&lt; den folgenden Quellcode für das Formular wie nebenstehend angezeigt.</p> <p>Prüfen Sie ihre Eingaben auf Fehler und speichern Sie das Ergebnis (STRG +S).</p> <p>Hinweis: Formulare bestehen aus Komponenten, wie beispielsweise Texteingabefeldern und Schaltflächen. In SELFHTML finden Sie dazu jede Menge weitere Beispiele (<a href="http://de.selfhtml.org/html/formulare/">http://de.selfhtml.org/html/formulare/</a>).</p>																		
	<p>Testen Sie nun ob die &gt;&gt;View&lt;&lt; des Formulars korrekt implementiert wurde. Mit &gt;&gt;View&lt;&lt; ist das Darstellung des Formulars gemeint.</p> <p>Um die Ereignisse, &gt;&gt;Die Auswertung des Formulars&lt;&lt; kümmern wir uns etwas später.</p> <p>Wenn der Webserver gestartet ist können wir</p>																		

	<p>das Ergebnis im Internetbrowser testen.</p> <p>Öffnen Sie dazu den Internetbrowser &gt;&gt;Firefox&lt;&lt; und geben Sie den Pfad zur gerade erstellen Datei an. Geben Sie dazu den in der Grafik angezeigten Pfad ein.</p>
<pre>#content{     border:2px solid red;     float: right;     width: 70%;     height:auto; }  #content p{     text-align:center; }  #content form td{     background-color:black;     color:white; }</pre>	<p>Wir wollen das Formular hübscher gestalten und müssen dazu das Stylesheet &gt;&gt;style.css&lt;&lt; erweitern.</p> <p>Öffnen Sie dazu mit einem einfachen Editor die Datei &gt;&gt;style.css&lt;&lt;.</p> <p>Ergänzen Sie das Stylesheet an der Stelle wo die div-Box &gt;&gt;content&lt;&lt; deklariert ist (darunter) wie angezeigt.</p>
	<p>Testen Sie wie zuvor erneut die &gt;&gt;View&lt;&lt;.</p> <p>Das Ergebnis sollte sein, dass die Zellen in der Formular-Tabelle rot umrandet sind, die Hintergrundfarbe schwarz und die Schriftfarbe weiß dargestellt wird.</p>
	<p>In den nächsten Schritten kümmern wir uns um die Funktionalität &gt;&gt;Ausrechnen&lt;&lt;. Das Ereignis, das heißt die Auswertung der Eingaben durch den Benutzer sollen dann erfolgen, wenn die Schaltfläche &gt;&gt;Ausrechnen&lt;&lt; angeklickt wird.</p> <p>Diese Auswertung kann eine Seitenbeschreibungssprache HTML nicht leisten. Wir nutzen deshalb die Programmiersprache PHP dafür.</p>

	<p>Im Moment rufen wir mit dem Klick auf die Schaltfläche &gt;&gt;Ausrechnen&lt;&lt; eine Datei &gt;&gt;note1.php&lt;&lt; auf (siehe Form-Action in der Datei &gt;&gt;content.php&lt;&lt;):</p> <pre data-bbox="817 387 1497 517">&lt;form action="note1.php" method="post"&gt;   &lt;h4&gt;Mein Noten-Rechner&lt;/h4&gt;   &lt;table&gt;     &lt;tr&gt;</pre> <p>Diese Datei existiert aber noch nicht, deshalb erscheint auch die nebenstehende Fehlermeldung.</p>
	<p>Wir werden diese Datei nun erstellen und das EVA-Prinzip (<b>E</b>ingaben lesen, <b>v</b>erarbeiten und <b>a</b>usgeben) mit Hilfe von PHP anwenden.</p> <p>Erzeugen Sie dazu die Datei &gt;&gt;note1.php&lt;&lt;.</p> 
 <pre data-bbox="82 1189 788 1585">&lt;!DOCTYPE HTML&gt; &lt;html&gt; &lt;head&gt;   &lt;title&gt;*&lt;MyTheme*&lt;/title&gt;   &lt;meta name="author" content="Christine Janischek"&gt;   &lt;meta name="keywords" content="MyTheme"&gt;   &lt;meta name="description" content="Meine erste PHP-Seite"&gt;   &lt;style type="text/css"&gt;     @import "style.css";   &lt;/style&gt; &lt;/head&gt; &lt;body&gt;   &lt;div id="content"&gt;     &lt;p&gt;     &lt;/p&gt;   &lt;/div&gt; &lt;/body&gt;</pre>	<p>Öffnen Sie die Datei &gt;&gt;note1.php&lt;&lt; mit einem einfachen Editor. Die Datei ist wie erwartet noch leer!</p> <p>Ergänzen Sie das HTML-Grundgerüst.</p>

<pre> &lt;div id="content"&gt;   &lt;p&gt;     &lt;?php     //Eingaben lesen     \$pMathe = \$_POST['tfMathe'];     \$pDeutsch = \$_POST['tfDeutsch'];     \$pEnglisch = \$_POST['tfEnglisch'];     \$pBwl = \$_POST['tfBwl'];     ?&gt;   &lt;/p&gt; &lt;/div&gt; </pre>	<p>Wir lesen nun im Ersten Schritt die Eingabewerte aus den Texteingabefeldern (tfMathe, tfEnglisch, tfDeutsch, tfBWL) und übernehmen diese Werte in die PHP-Parameterattribute (pMathe, pEnglisch, pDeutsch, pBWL).</p> <p>Ergänzen Sie dazu im Content-Tag den angezeigten Quellcode.</p>
<div style="text-align: center; border: 1px solid black; padding: 5px; background-color: #ffffcc; margin-bottom: 5px;"> <b>Berechnung</b> </div> <div style="border: 1px solid black; padding: 5px; background-color: #ffffcc; margin-bottom: 5px;">     durchschnitt_berechnen (pMathe,pEnglisch,pDeutsch,pBwl)   </div> <p style="text-align: center;">UML-Klasse: Berechnung</p>	<p>Egal in welcher Programmiersprache wir heutzutage programmieren lautet das Prinzip &gt;&gt;Objektorientierung&lt;&lt;. Im Großen und Ganzen geht es dabei darum, Ordnung zu halten, d. h. den Quellcode auf eine bestimmte Art und Weise zu strukturieren.</p> <p>Das Prinzip stellt u. a. sicher, dass wir die Anwendung später problemlos erweitern können und Quellcodebestandteile wiederverwenden können.</p> <p>Wir halten uns von Anfang an dieses Prinzip und werden es in den nächsten Schritten erstmals anwenden.</p>
<pre> &lt;body&gt;   &lt;div id="content"&gt;     &lt;p&gt;       &lt;?php       //Eingaben lesen       \$pMathe = \$_POST['tfMathe'];       \$pDeutsch = \$_POST['tfDeutsch'];       \$pEnglisch = \$_POST['tfEnglisch'];       \$pBwl = \$_POST['tfBwl'];       //Für die Objektorientierung       class Berechnung{       public function berechne_durchschnitt(\$pMathe,\$pDeutsch,\$pEnglisch,\$pBwl){           \$pErgebnis = (\$pMathe+\$pDeutsch+\$pEnglisch+\$pBwl)/4;           return \$pErgebnis;       }       }     &lt;/p&gt;   &lt;/div&gt; &lt;/body&gt; </pre>	<p>Erweitern Sie dazu den gerade eingefügten Quellcode um den noch fehlenden Teil.</p>
<pre> &lt;body&gt;   &lt;div id="content"&gt;     &lt;p&gt;       &lt;?php       //Eingaben lesen       \$pMathe = \$_POST['tfMathe'];       \$pDeutsch = \$_POST['tfDeutsch'];       \$pEnglisch = \$_POST['tfEnglisch'];       \$pBwl = \$_POST['tfBwl'];       //Für die Objektorientierung       class Berechnung{       public function berechne_durchschnitt(\$pMathe,\$pDeutsch,\$pEnglisch,\$pBwl){           \$pErgebnis = (\$pMathe+\$pDeutsch+\$pEnglisch+\$pBwl)/4;           return \$pErgebnis;       }       }       //Verarbeitung       \$durchschnitt = new Berechnung();       \$ausgabe = \$durchschnitt -&gt; berechne_durchschnitt(\$pMathe,\$pDeutsch,\$pEnglisch,\$pBwl);     &lt;/p&gt;   &lt;/div&gt; &lt;/body&gt; </pre>	<p>Wir nutzen nun den für die Strukturierung notwendigen Quellcode.</p> <p>Ergänzen Sie dazu den Quellcode für die Verarbeitung der Formulare Daten.</p>

```

<body>
  <div id="content">
    <p>
      <?php
      //Eingaben Lesen
      $pMathe = $_POST['tfMathe'];
      $pDeutsch = $_POST['tfDeutsch'];
      $pEnglisch = $_POST['tfEnglisch'];
      $pBwl = $_POST['tfBwl'];

      //Verarbeitung
      $durchschnitt = new berechnung();
      $ausgabe = $durchschnitt -> berechne_durchschnitt($pMathe,$pDeutsch,$pEnglisch,$pBwl);

      //Ausgabe
      echo $ausgabe;
    >
  </p>
</div>
</body>

```

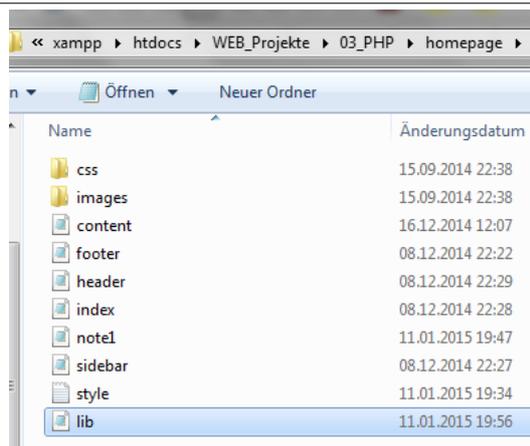
Abschließend möchten Wir das Ergebnis auch ausgeben.

Ergänzen Sie dazu den Quellcode für die Ausgabe des Ergebnisses.

Testen Sie das Formular. Geben Sie Noten in das Formular ein und klicken Sie auf die Schaltfläche >>Ausrechnen<<.

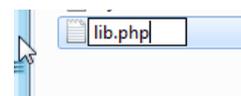
Im Ergebnis sollte die Ausgabe so aussehen:

3.25



Weil wir künftig unsere Seite um weitere Berechnungen (BMI, Umfang, Flächeninhalt,...) erweitern möchten und diese auf unterschiedlichen Unterseiten nutzen möchten, lagern wir den Quellcode der Klasse in eine extra Datei aus.

Erzeugen Sie dazu eine neue Datei >>lib.php<<



```

//Für die Objektorientierung
class berechnung{
    public function berechne_durchschnitt($pMathe,$pDeutsch,$pEnglisch,$pBwl) {
        $pErgebnis = ($pMathe + $pDeutsch + $pEnglisch + $pBwl)/4;
        return $pErgebnis;
    }
}

```

Schneiden Sie den für die Objektorientierung notwendigen Quellcode aus.

(STRG + X)

```

lib.php - Editor
Datei Bearbeiten Format Ansicht ?
<?php
//Für die Objektorientierung
class berechnung{
    public function berechne_durchschnitt($pMathe,$pDeutsch,$pEnglisch,$pBwl) {
        $pErgebnis = ($pMathe + $pDeutsch + $pEnglisch + $pBwl)/4;
        return $pErgebnis;
    }
}
?>

```

Öffnen Sie die noch leere Datei >>lib.php<< und fügen Sie den ausgeschnittenen Quellcode ein und ummanteln Sie die Klasse mit dem <?php ... ?>-Tag wie angezeigt.

Speichern Sie die Änderungen in beiden Dateien (note1.php und lib.php).

```

note1 - Editor
Datei Bearbeiten Format Ansicht ?
<head>
  <title>**MyTheme**</title>
  <meta name="author" content="Christine Janischek">
  <meta name="keywords" content="MyTheme">
  <meta name="description" content="Meine erste PHP-Seite">
  <style type="text/css">
  @import "style.css";
  </style>
  <?php include('lib.php'); ?>
</head>

```

Damit wir die berechnende Methode in der Datei >>note1.php<< trotz Auslagerung nutzen können, müssen wir ähnlich wie beim Stylesheet darauf verweisen.

Ergänzen und speichern Sie dazu im <head>...</head> den notwendigen PHP-include-Befehl.

Die Kopfzeile

Die Navigation

Mein Noten-Rechner

Note in Mathematik:	<input type="text" value="3"/>
Note in Deutsch:	<input type="text" value="2"/>
Note in Englisch:	<input type="text" value="1"/>
Note in Bwl:	<input type="text" value="5"/>

Die Fußzeile

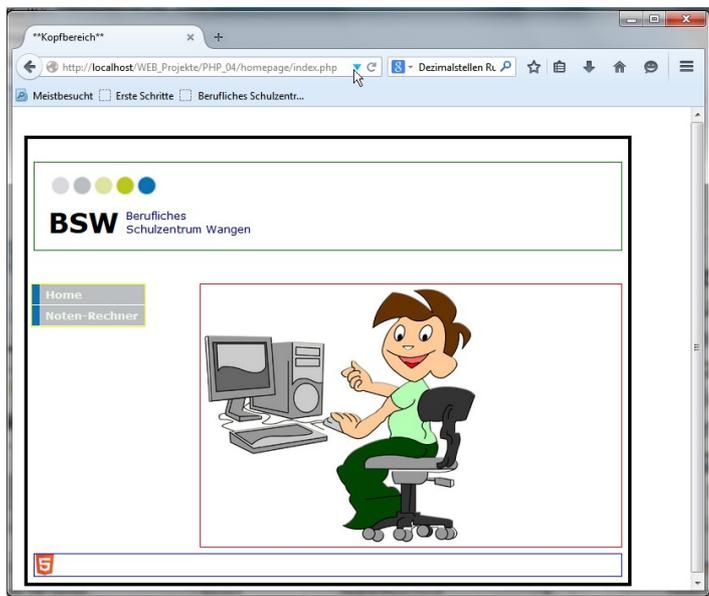
2.75

Testen Sie die Anwendung erneut. An der Funktionalität sollte sich nichts geändert haben.

Mit der vorgenommenen Auslagerung können wir nun die Berechnende Methode >>berechne\_durchschnitt()<<, immer wieder und an unterschiedlichen Stellen, nutzen.

**Herzlichen Glückwunsch Sie haben ihre erstes interaktives Formular erstellt.**

## 4 Umsetzung informatischer Prinzipien im PHP-Quellcode

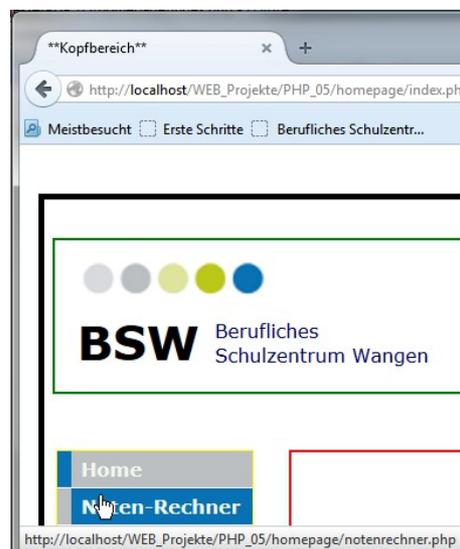


Informatiker sind immer bestrebt Ihre Arbeit effizient zu verrichten. In der Softwareentwicklung wird daher viel Wert darauf gelegt auf Wiederholungen von Quellcode zu verzichten. Im Ergebnis sollen es möglich sein Bestandteile des Quellcodes wiederzuverwenden und gleichzeitig soll sichergestellt werden, dass bei Änderungen nur nur eine „Baustelle“ aufgetan wird.

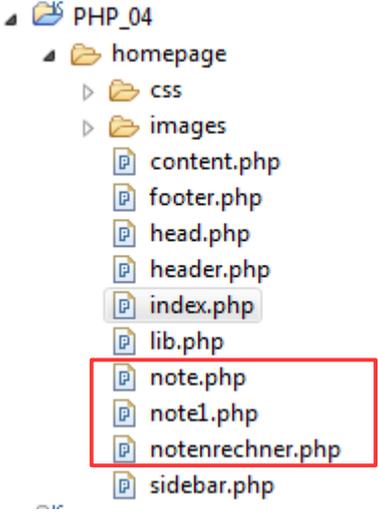
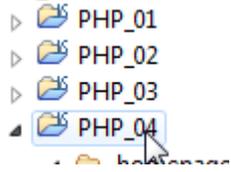
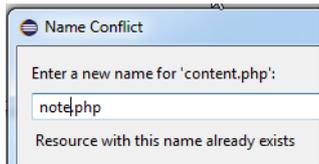
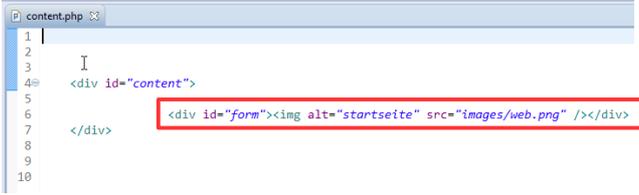
### Wichtige informatische Prinzipien:

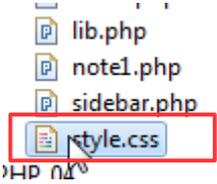
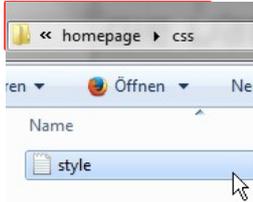
1. Wiederverwendung → keine Redundanzen (keine Widersprüche, deshalb keine Wiederholungen)
2. Zerlegung

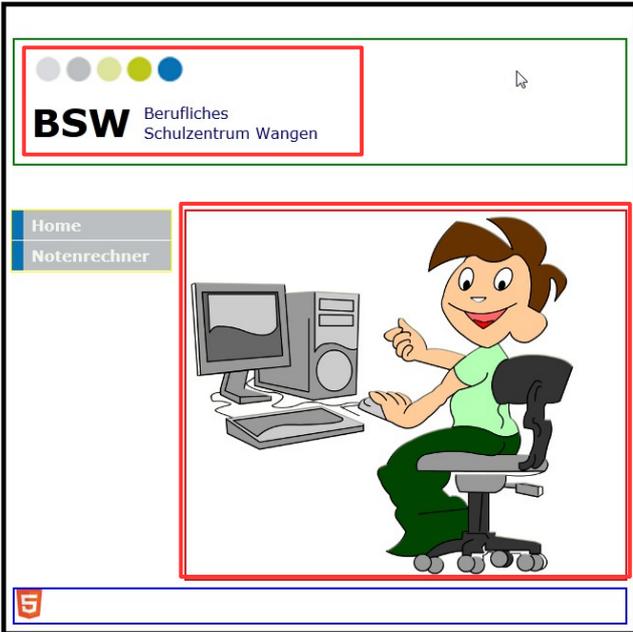
Projekt: Noten-Rechner



Noten-Rechner (notenrechner.php)

	<p>Erzeugen Sie dazu ein neues Dynamisches Webprojekt-Verzeichnis (Workspace im ht-docs-Verzeichnis) „04_PHP“. Kopieren Sie den Inhalt des gerade fertiggestellte Verzeichnis &gt;&gt; 03_PHP&lt;&lt;:</p>  <p>Und geben Sie dem Verzeichnis den Namen &gt;&gt;04_PHP&lt;&lt;.</p>
	<p>Kopieren Sie Datei content.php. Fügen Sie die Datei ein und benennen Sie die Datei um. Nennen Sie diese Datei dazu note.php.</p> 
	<p>Öffnen Sie die Datei content.php und ändern Sie den Quellcode wie nebenstehend angezeigt.</p> <p>Kopieren Sie dann die Bild-Datei web.png in das Verzeichnis images.</p>

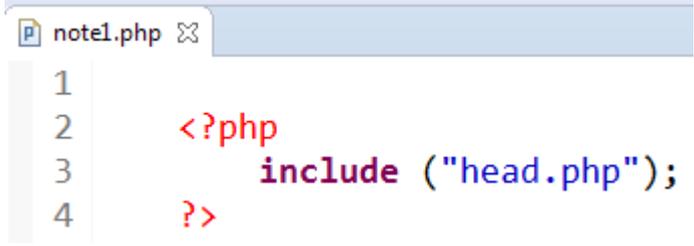
 <pre> 1 &lt;!DOCTYPE HTML&gt; 2 3 &lt;html&gt; 4 &lt;head&gt; 5   &lt;title&gt;**MyTheme**&lt;/title&gt; 6   &lt;meta name="author" content="Christine Janischek"&gt; 7   &lt;meta name="keywords" content="MyTheme"&gt; 8   &lt;meta name="description" content="Meine erste PHP-Seite"&gt; 9 &lt;style type="text/css"&gt; 10   @import "style.css"; 11 &lt;/style&gt; 12 &lt;/head&gt; </pre>	<p>Erzeugen Sie eine neue, noch leere PHP-Datei mit dem Namen head.php.</p> <p>Öffnen Sie die dann die Datei index.php. Schneiden Sie die Quellcodezeilen für den &lt;head&gt;&lt;/head&gt;, das DOCTYPE- und html-Tag aus und fügen Sie sie in die Datei head.php ein.</p> <p>Speichern Sie die Veränderung in der Datei index.php ab.</p>
	 <p>Verlagern Sie nun das bestehende Stylesheet in das css-Verzeichnis.</p>
 <pre> 1 &lt;!DOCTYPE HTML&gt; 2 3 &lt;html&gt; 4 &lt;head&gt; 5 &lt;title&gt;**Kopfbereich**&lt;/title&gt; 6 &lt;meta name="author" content="Christine Janischek" /&gt; 7 &lt;meta name="keywords" content="Kontaktseite" /&gt; 8 &lt;meta name="date" content="2015-01-06T12:40:00+01:00" /&gt; 9 &lt;meta name="description" content="Kontaktseite von Christine Janischek" /&gt; 10 &lt;style type="text/css"&gt; 11   @import "css/style.css"; 12 &lt;/style&gt; </pre>	<p>Öffnen Sie erneut die Datei head.php und passen Sie die Quellangabe für das Stylesheet entsprechend an.</p> <p>Testen Sie ob die Formatangaben aus dem Stylesheet für die index.php angezeigt werden.</p>
 <pre> 10 &lt;style type="text/css"&gt; 11   @import "css/style.css"; 12 &lt;/style&gt; 13 &lt;?php 14   include("lib.php"); 15 ?&gt; </pre>	<p>Ergänzen Sie darunter den Verweis auf unsere künftige, eigene PHP-Bibliothek:</p> <pre> &lt;?php     include("lib.php"); ?&gt; </pre> <p>In der lib.php werden unseren eigenen Quellcode strukturiert ablegen, sodass wir ihn künftig flexibel, auch in anderen Pro-</p>

	<p>jekten, <b>wiederverwenden</b> können.</p> <p>In der header.php soll das neue Logo platziert werden. Außerdem soll danach auf der Startseite die Grafik „web.png“ eingebettet werden.</p> <p>Dazu wird im Ersten Schritt die Grafiken „logo.png“ eingebunden und mit Hilfe des CSS platziert und ausgerichtet. Kopieren Sie die Datei in Ihr Verzeichnis „images“.</p> 
--	---

<pre> 1 header.php 2 index.html 2 &lt;div id="header"&gt; 3   &lt;div id="ci"&gt; 4     &lt;div id="Logo"&gt;&lt;/div&gt; 5     I   &lt;div id="schriftzug"&gt; 6       &lt;div id="bsw_Logo"&gt;BSW&lt;/div&gt; 7       &lt;div id="bsw"&gt;Berufliches &lt;br&gt;Schulzentrum Wangen&lt;/div&gt; 8     &lt;/div&gt; 9   &lt;/div&gt; 10 &lt;/div&gt; </pre>	<p>Wir möchten die gesamte Ausrichtung und Einbettung des Logos mit CSS umsetzen. Dazu werden wir jede Menge CSS-Boxen definieren und schachteln.</p> <p>Setzen Sie dazu zunächst die geschachtelten Boxen in HTML um, übernehmen Sie dazu den nebenstehenden Quellcode in die Datei „header.php“.</p>
---	--

Erweitern Sie dann die dazu benötigten Formatierungsangaben für die Boxen in der CSS-Datei (style.css):

Zugehöriges Box-Modell im CSS	
#header	Höhe der Box in 130 Pixel Textausrichtung links Rahmen 2 Pixel breit durchgehend Grün
#header #ci	Ausrichtung des Elements links Breite der Box in 320 Pixel Rahmen 1 Pixel breit durchgehend Grün
#header #Logo	Ausrichtung des Elements links Breite der Box in 175 Pixel Höhe der Box in 42 Pixel Relative Angabe der Bildquelle für den Hintergrund der Box(..../images/logo.png)
#header #schriftzug	Ausrichtung des Elements links

	Breite der Box in 310 Pixel Rahmen 1 Pixel breit durchgehend Grün
<code>#header #schriftzug #bsw_Logo</code>	Ausrichtung des Elements links Innenabstand Links 5 Pixel Relative Angabe der Schriftgröße 2.5em Schriftfarbe schwarz Schriftgewicht fett
<code>#header #schriftzug #bsw</code>	Ausrichtung des Elements rechts Innenabstand Oben 5 Pixel Schriftgewicht normal
 <pre> 1 2 &lt;?php 3 include ('head.php'); 4 ?&gt; </pre>	<p>Da wir künftig darauf achten müssen Wiederholungen zu vermeiden, sollten wir in der Datei note1.php eine Optimierung des Quellcodes vornehmen.</p> <p>Ersetzen Sie deshalb in dieser Datei das &lt;head&gt;&lt;/head&gt;-Tag durch den Verweis auf die Datei head.php.</p>
<pre> &lt;?php include ('head.php'); ?&gt; &lt;body&gt; &lt;div id="wrap"&gt; &lt;?php include ('header.php'); ?&gt;  &lt;?php include ('note.php'); ?&gt;  &lt;?php include ('sidebar.php'); ?&gt;  &lt;?php include ('footer.php'); ?&gt; &lt;/div&gt; &lt;/body&gt; </pre>	<p>Nun benötigen wir eine Datei die alle Bestandteile des Systems enthält (zusammenführt).</p> <p>Erzeugen Sie eine neu Datei, noch leere Datei, mit dem Namen notenrechner.php.</p> <p>Mit dem Aufruf aller benötigten php-Includes:</p> <ul style="list-style-type: none"> <li>• head.php → Kopf</li> <li>• header.php → Top-Frame (Kopfzeile)</li> <li>• note.php → Main-Frame (Formular)</li> <li>• sidebar.php → Navigation-Frame (Navi)</li> <li>• footer.php → Bottom-Frame (Fußzeile)</li> </ul> <p>Verweisen wir auf die benötigten Dateien.</p> <p><b>Hinweis:</b> Die Wrap-Box umfasst alle Seitenteile die unabhängig von der Größe und Auflösung des Bildschirms in einer Einheit dargestellt werden sollen. Die Angabe der Breite für die Wrap-Box kann je nach Seitenanforderung (Corporate Identity) eine relative oder absolute Angabe enthalten.</p>

Testen Sie das Ergebnis.



index.php

Den Noten-Rechner in der Navigation ergänzen:

```
<div id="sidebar">
  <ul>
    <li><a href="index.php">Home</a></li>
    <li><a href="notenrechner.php" target="_parent">Noten-Rechner</a></li>
  </ul>
</div>
```

Öffnen Sie dazu die Datei sidebar.php und verändern Sie die Verweisangabe für den Menü-Punkt „Noten-Rechner“.

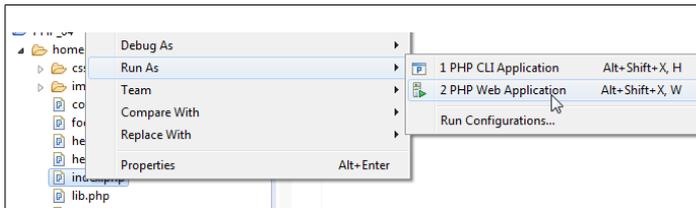
Die Höhe des Footers festlegen:

```
211 #footer{
212     clear:both;
213     height: 33px;|
214     border:2px solid blue;
215
216 }
217
```

Ergänzen Sie im CSS abschließend die Angabe der Höhe mit 33px.

Testen Sie das Projekt

Stellen Sie sicher dass mindestens der Dienst (xampp → Webserver: Apache) läuft.



Klicken Sie die Datei index.php an und wählen Sie im Kontextmenü die Option RUN AS → PHP Web Application →

Pfadangabe:

[http://localhost/WEB Projekte/PHP\\_04/homepage/index.php](http://localhost/WEB_Projekte/PHP_04/homepage/index.php)

Hinweis:

Sie können alternativ die Seite auch im externen Browser testen.



Testen Sie nun den Notenrechner.

**Mein Noten-Rechner**

Note in Mathematik:	<input type="text" value="2.5"/>
Note in Deutsch:	<input type="text" value="2.0"/>
Note in Englisch:	<input type="text" value="3.0"/>
Note in Bwl:	<input type="text" value="2.0"/>

Das Ergebnis solle so aussehen:



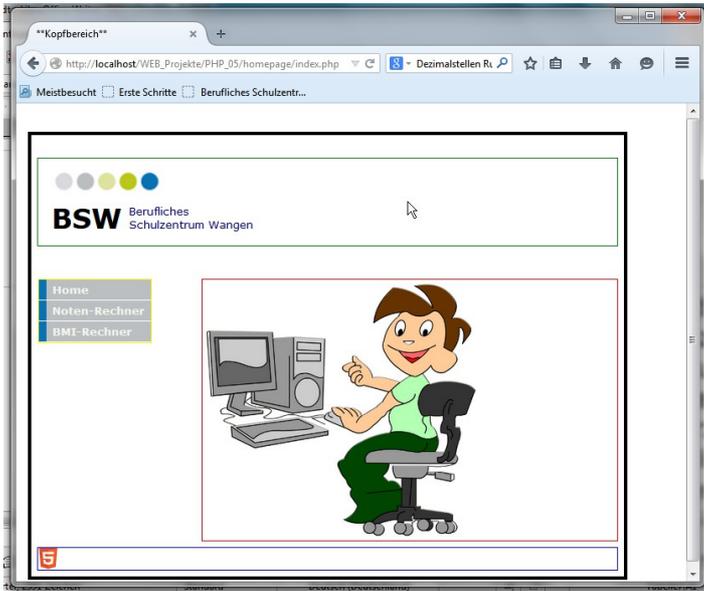
Prima nun hat Ihre Anwendung einiges an Qualität gewonnen.

Wenn wir unser Projekt erweitern müssen wir diese Aspekte (Prinzipien) künftig immer mit einbeziehen.

**Wichtige informatische Prinzipien:**

1. Wiederverwendung → keine Redundanzen (keine Widersprüche, deshalb keine Wiederholungen)
2. Zerlegung

## 5 Praktische Konzepte der objektorientierten Programmierung

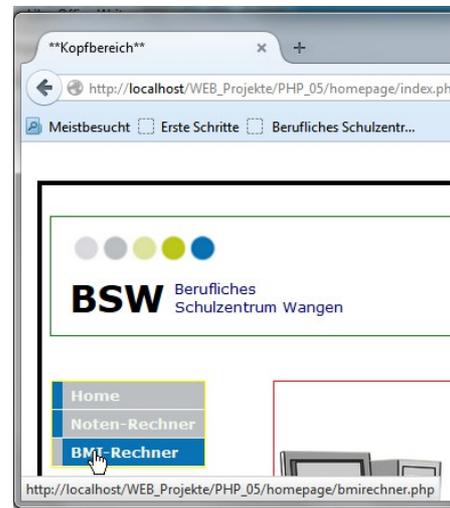


In der Informatik händeln wir Strukturen. Diese Strukturen machen die Sprache aus.

Zwischenzeitlich ähneln sich die Sprachen, da die Objektorientierung Einzug erhalten hat. → Java ↔ PHP

- Wichtige Begriffe:**
1. Klassen, Objekte, Attribute, Methoden und Konstruktoren
  2. Set- und Get-Methoden
  3. EVA-Prinzip
  4. Kontrollstrukturen → Fallunterscheidungen
  5. UML → Klassendiagramme erstellen
  6. Objektorientierung → Begriffsklärung

Projekt: BMI-Rechner

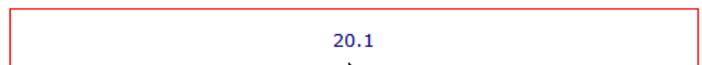


BMI-Rechner (bmirechner.php)

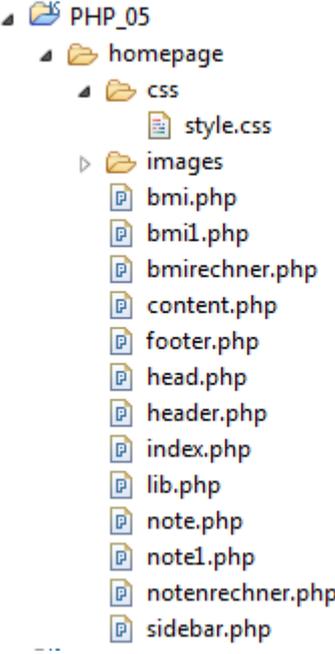
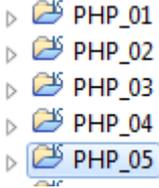
**Mein BMI-Rechner**

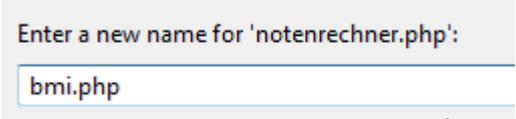
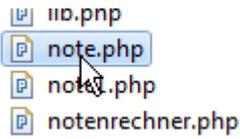
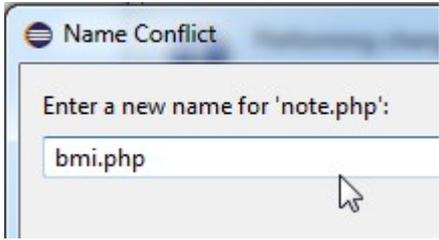
Körpergewicht in kg:	<input type="text"/>
Körpergröße im m (#.#):	<input type="text"/>
<input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>	

BMI-Formular (bmi.php)



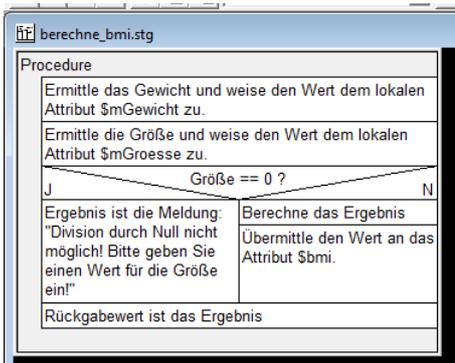
BMI-Auswertung (bmi1.php)

	<p>Erzeugen Sie dazu ein neues Dynamisches Webprojekt-Verzeichnis (Workspace im htdocs-Verzeichnis) „05_PHP“. Kopieren Sie den Inhalt des gerade fertiggestellte Verzeichnis &gt;&gt; 04_PHP&lt;&lt;:</p>  <p>Und geben Sie dem Verzeichnis den Namen &gt;&gt;05_PHP&lt;&lt;.</p>
<p>Erweitern Sie die Navigationsleiste:</p> <pre data-bbox="92 1003 778 1146">&lt;div id="sidebar"&gt; &lt;ul&gt; &lt;li&gt;&lt;a href="index.php"&gt;Home&lt;/a&gt;&lt;/li&gt; &lt;li&gt;&lt;a href="notenrechner.php" target="_parent"&gt;Noten-Rechner&lt;/a&gt;&lt;/li&gt; &lt;li&gt;&lt;a href="bmirechner.php" target="_parent"&gt;BMI-Rechner&lt;/a&gt;&lt;/li&gt; &lt;/ul&gt; &lt;/div&gt;</pre>	<p>Öffnen Sie dazu die Datei „sidebar.php“ und ergänzen Sie den neuen Menüpunkt für das BMI-Rechner-Projekt.</p>

	<p>Kopieren Sie dazu die Datei „noterechner.php“. Fügen Sie die Datei ein und benennen Sie die Datei um. Nennen Sie diese Datei dazu „bmi-rechner.php“.</p>  <p>Öffnen Sie die Datei „bmi.php“. Verändern Sie den Quellcode an genau einer Stelle.</p> <p>Verweisen Sie dazu auf die Datei „bmi.php“:</p> <pre> &lt;?php     include ("bmi.php"); ?&gt; </pre>
	<p>Kopieren Sie Datei „note.php“. Fügen Sie die Datei ein und benennen Sie die Datei um. Nennen Sie diese Datei dazu „bmi.php“.</p> 



Kontrollstruktur/Fallunterscheidung:  
IF-ELSE, das Struktogramm:



Alternativen bereitzustellen.

Da im vorliegenden Fall eine Division durch Null nicht berechenbar wäre, wird der Nutzer bei einer fehlenden Eingabe für die Größe (Ja-Fall / IF-Zweig) mit einer Meldung darauf aufmerksam gemacht. Ansonsten (Nein-Fall / ELSE-Zweig) wird der BMI berechnet und zurückgegeben.

```

bmi.php  bmi1.php  lib.php
2 // Für die Objektorientierung
3 class Person {
4     //Attribute
5     private $gewicht;
6     private $groesse;
7     private $bmi;
8
9     //Konstruktor
10    function __construct() {
11    }
12
13    //Get-und Set-Methoden
14    public function set_gewicht($pGewicht) {
15        $this->gewicht = $pGewicht;
16    }
17    public function get_gewicht() {
18        return $this->gewicht;
19    }
20    public function set_groesse($pGroesse) {
21        $this->groesse = $pGroesse;
22    }
23    public function get_groesse() {
24        return $this->groesse;
25    }
26    public function set_bmi($pBmi) {
27        $this->bmi = $pBmi;
28    }
29    public function get_bmi() {
30        return $this->bmi;
31    }
32 }
33

```

Ergänzen Sie den Quellcode für die Klasse Person.

Grundgerüst einer Klasse:

```

class Klassenname {
    //Attribute
    private $attributname;

    //Konstruktor
    function __construct() {
    }

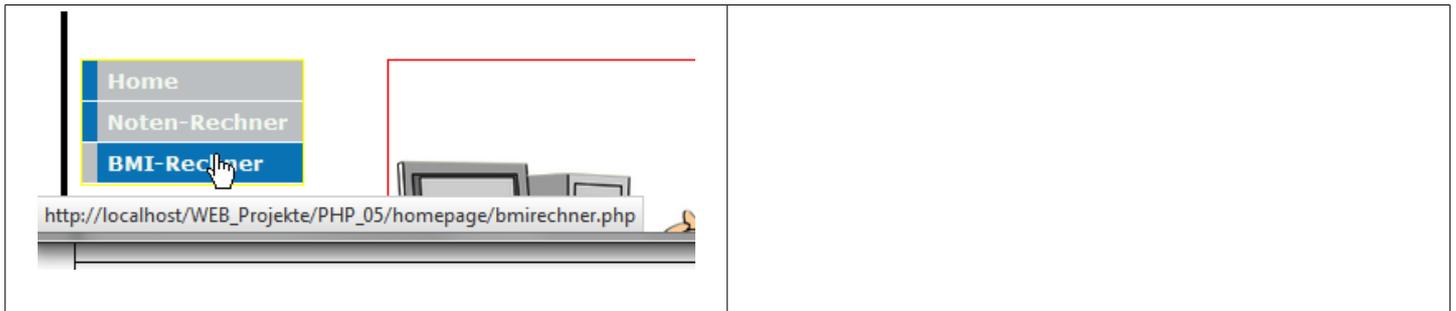
    //Get-und Set-Methoden
    public function
set_attributname($pAttributname) {
        $this->attributname = $pAttributname;
    }
    public function get_attributname() {
        return $this->attributname;
    }

    //Höhere Methoden
}

```

Testen Sie den BMI-Rechner:

Öffnen Sie die Anwendung im Browser.



**Mein BMI-Rechner**

Körpergewicht in kg:

Körpergröße im m (##.##):

Geben Sie die Werte für das Körpergewicht und die Körpergröße ein und klicken Sie auf die Schaltfläche „Ausrechnen“.

20.1

Das berechnete Ergebnis wird berechnet und auf eine Kommastelle gerundet ausgegeben.

Testen Sie folgende Anwendungsfälle:

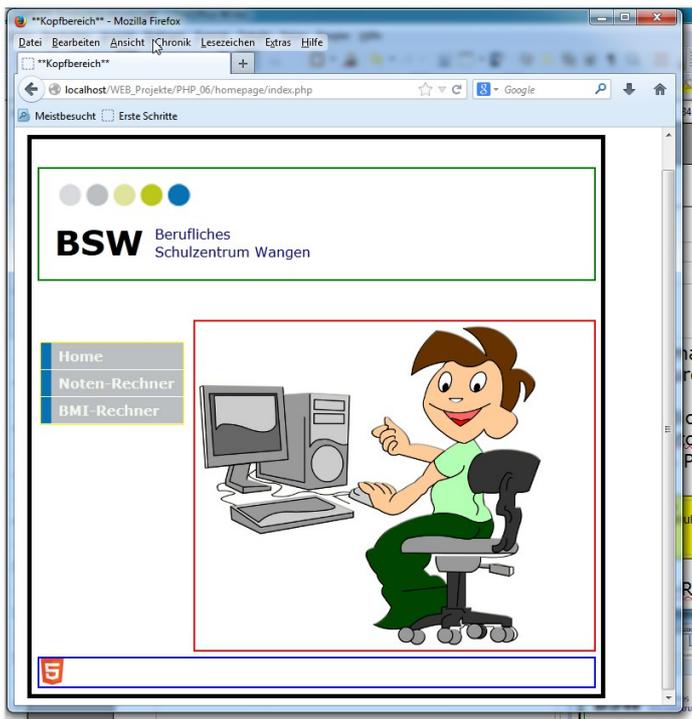
**Anwendungsfälle:**

<input type="text" value="66"/> <input type="text" value="1.7"/> <input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>	<input type="text" value="92"/> <input type="text" value="1.86"/> <input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>	<input type="text" value="30"/> <input type="text" value="1.1"/> <input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>	<input type="text" value="24"/> <input type="text" value="0.98"/> <input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>
<p><i>Myra Bellamy ist eine Frau (54).</i></p> <p>Sie gibt für ihr Gewicht den Wert „66“ und ihre</p>	<p><i>Red Barklay ist ein Mann (42).</i></p> <p>Er gibt für sein Gewicht den Wert „92“ und seine Körpergröße „1.86“ ein und erhält</p>	<p><i>Bart Simpson ist ein Junge (6).</i></p> <p>Er gibt für sein Gewicht den Wert „30“ und seine Körpergröße „1.10“ ein</p>	<p><i>Maggie Simpson ist ein Mädchen (4).</i></p> <p>Sie gibt für ihr Gewicht den Wert „24“ und ihre Körpergröße „0.98“ ein</p>

Körpergröße „1.7“ ein und erhält das folgende Ergebnis angezeigt:	das folgende Ergebnis ange- zeigt:	und erhält das folgende Ergebnis angezeigt:	und erhält das folgende Ergebnis angezeigt:
<hr/> 22.8 <hr/>	<hr/> 26.6 <hr/>	<hr/> 24.8 <hr/>	<hr/> 25 <hr/>

**Welcher Ausgabe erfolgt auf eine fehlende der Größe?**

## 6 Kontrollstrukturen



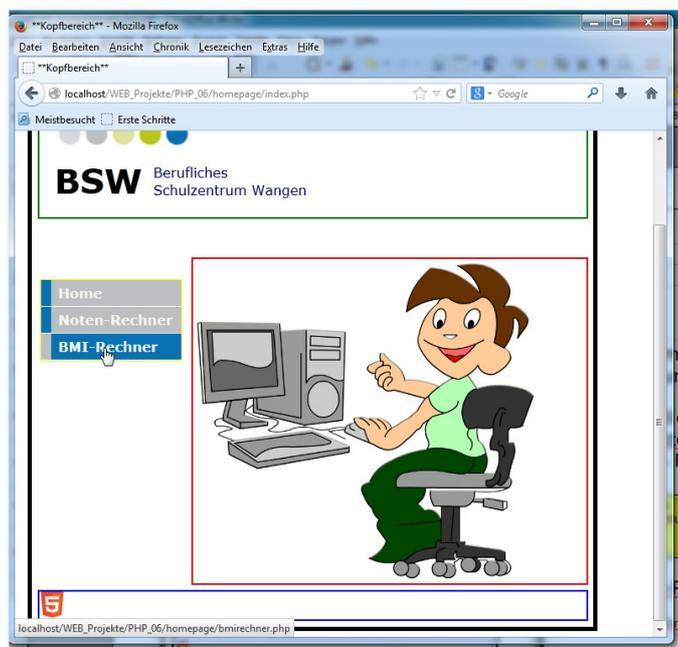
In der Informatik händeln wir Strukturen. Diese Strukturen machen die Sprache aus.

Zwischenzeitlich ähneln sich die Sprachen, da die Objektorientierung Einzug erhalten hat. → Java ↔ PHP

### Wichtige Begriffe:

1. Kontrollstrukturen (Fallunterscheidungen, Schleifen)
2. Im speziellen die Fallunterscheidung:
  - IF ELSE
  - ELSEIF
  - SWITCH CASE
3. Vererbung

### Projekt: BMI-Rechner



BMI-Rechner-Erweiterung (bmirechner.php)

**Mein BMI-Rechner**

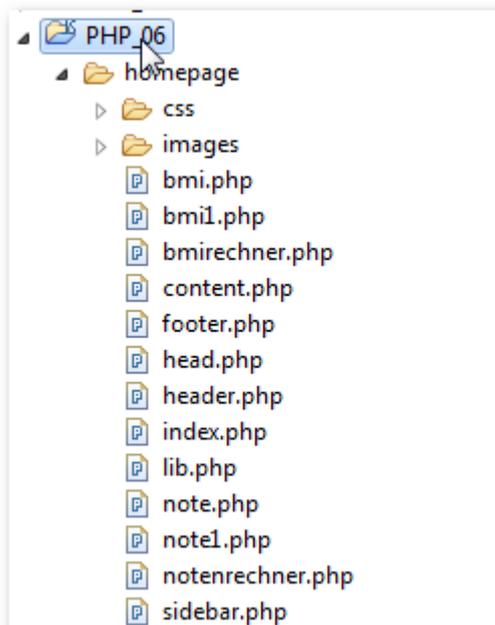
Körpergewicht in kg:	<input type="text"/>
Körpergröße in m (#.#):	<input type="text"/>
Alter in Jahren:	<input type="text"/>
	<input type="radio"/> weiblich <input type="radio"/> männlich
	<input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>

BMI-Formular (bmi.php)

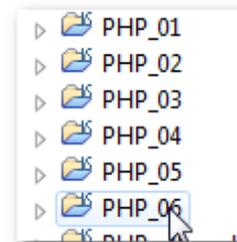
Gewicht: 66  
Größe: 1.7  
BMI: 22.8

Geschlecht: 66  
Klassifikation: Normalgewicht  
Min: 22  
Max: 27

BMI-Auswertung (bmi1.php)



Erzeugen Sie dazu ein neues Dynamisches Webprojekt-Verzeichnis (Workspace im htdocs-Verzeichnis) „06\_PHP“. Kopieren Sie den Inhalt des gerade fertiggestellte Verzeichnis >> 05\_PHP<<:



Und geben Sie dem Verzeichnis den Namen >>06\_PHP<<.

```

1
2     <?php
3         include ("head.php");
4     ?>
5 <body>
6     <div id="wrap">
7         <?php
8             include ("header.php");
9         ?>
10
11         <?php
12             include ("bmi.php");
13         ?>
14
15         <?php
16             include ("sidebar.php");
17         ?>
18
19         <?php
20             include ("footer.php");
21         ?>
22     </div>
23 </body>

```

Die Hauptseite des BMI-Rechners „bmi-rechner.php“, bleibt unverändert.

Alter in Jahren:	<input type="text"/>
	<input type="radio"/> weiblich <input type="radio"/> männlich

Änderungen im BMI-Formular:

Optionsfelder-Komponente (Radio-Buttons):

```

<tr>
<td><input type="radio" name="rbOperator" value="1" />weiblich</td>
<td><input type="radio" name="rbOperator" value="2" />männlich</td>
</tr>

```

Erweitern Sie das bestehende BMI-Formular „bmi.php“, um die noch fehlenden Komponenten (Testeingabefeld für das Alter und die Radio-Buttons für das Geschlecht).

Hinweis:

jeder Radio-Button benötigt einen eindeutigen Wert (value)! Um die Farbgebung des Formulars kümmern wir uns später.

```

//Eingaben lesen
$pGroesse = $_POST['tfGroesse'];
$pGewicht = $_POST['tfGewicht'];

```

1. A.:

Erweiterung der Aktions-Datei „bmi1.php“. *Eingaben lesen!*

Öffnen Sie für die nächsten Schritte diese Datei und führen Sie alle Anpassungen aus.

Fügen Sie die Anweisung zum Lesen des Alters (pAlter → tfAlter) aus dem Texteingabefeld hinzu.

```

18
19     // Verarbeitung
20     $bmi = new Berechnung ();
21     $person = new Person ();
22

```

2. A.:

Erweiterung der Aktions-Datei „bmi1.php“. *Objekte erzeugen!*

Wir beginnen im Anschluss daran mit der Verarbeitung.

Erzeugen Sie dazu zwei neue Objekte:

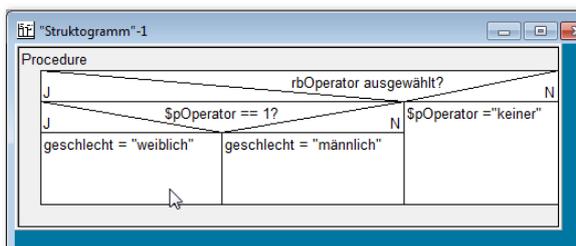
1. Ein Objekt der Klasse Berechnung mit der Bezeichnung „\$bmi“.
2. Ein Objekt der Klasse Person mit der Bezeichnung „\$person“.

```

23 // Übernahme des Operators und Bestimmung des Geschlechts
24 if (isset ( $_POST ['rbOperator'] )) {
25     $pOperator = $_POST ['rbOperator'];
26     if ($pOperator == 1) {
27         $person->set_geschlecht ( "weiblich" );
28     } else {
29         $person->set_geschlecht ( "männlich" );
30     }
31 } else {
32     $pOperator = "keiner";
33 };

```

Struktogramm:



3. A.:

Erweiterung der Aktions-Datei „bmi1.php“. *Fallunterscheidung IF ELSE!*

Im Rahmen der Verarbeitung müssen wir sicherstellen, dass das Optionsfeld (Radio-Button) geprüft wird. Falls der Operator aktiviert ist wird weiter geprüft, ob der Wert des Attributs \$pOperator gleich 1 ist. Falls \$pOperator gleich 1 ist, erhält das Attribut „geschlecht“ des aktuellen Personenobjektes (\$person) den Wert „weiblich“. Im Anderen Fall wird der Wert „männlich“ gesetzt.

Für den Fall dass kein Optionsfeld angeklickt wurde soll \$pOperator den Wert „keiner“ erhalten.

```

35 // Datentyp sicherstellen --> Kommazahl
36 settype ( $pGroesse, "float" );
37 settype ( $pGewicht, "float" );
38

```

4. A.:

Erweiterung der Aktions-Datei „bmi1.php“. *Datentypen sicherstellen!*

Im Anschluss an die Prüfung stellen wir, wie zuvor auch, sicher dass die Größe und das Gewicht als numerische Werte mit Nachkommastellen behandelt werden (→ Datentypen: float, double, integer, String, boolean).

```

39 // Berechnung über das Objekt $bmi der Klasse Berechnung abwickeln
40 $ausgabe = $bmi->berechne_bmi ( $person );

```

4. A.:

Erweiterung der Aktions-Datei „bmi1.php“. *Implementierte Methode zur Berechnung nutzen!*



1. B.:

Erweiterung unserer Klassenbibliothek „lib.php“. *Die Klasse Person!*

Die Klasse Person definiert alle Eigenschaften und Verhaltensweisen die für Personen des Systems benötigt werden.

Deklaration eines Attributs:

```
private $gewicht;
```

Deklaration der Get-Methode:

→ Eigenschaftswerte ermitteln

<p>UML-Klasse: Person</p>	<pre>public function get_gewicht() {     return \$this-&gt;gewicht; }</pre> <p>Deklaration der Set-Methode: → Eigenschaftswerte übermitteln</p> <pre>public function set_gewicht(\$pGewicht) {     \$this-&gt;gewicht = \$pGewicht; }</pre> <p>Deklaration einer höheren Methode:</p> <pre>public function klassifiziere(){     /*hier fehlt noch die Implementie-     rung*/ }</pre> <p>Erweitern Sie die Klasse Person, um die noch fehlenden Attribute und Methoden.</p>
	<p>2. B.: Erweiterung unserer Klassenbibliothek „lib.php“. <i>Spezielle Personen!</i> Männer im System sollen anders behandelt werden als Frauen. Deshalb müssen wir die speziellen Eigenschaften und Verhaltensweisen auslagern in extra Klassen.</p> <p>Erzeugen Sie die Klasse Frau und die Klasse Mann als Spezialisierung der Klasse Person: → Vererbung</p> <p>Deklaration einer spezialisierten Klasse:</p> <p>Frauen sind spezielle Personen:</p> <pre>class Frau extends Person{</pre>

## Klassifizierung:

	BMI männlich	BMI weiblich
<u>Untergewicht</u>	unter 20	unter 19
Normalgewicht	20-25	19-24
<u>Übergewicht</u>	26-30	25-30
Adipositas	31-40	31-40
<u>starke Adipositas</u>	größer 40	größer 40

## 3. B.:

Erweiterung unserer Klassenbibliothek „lib.php“. *Spezielle Verhaltensweisen!*

Die Implementierung der Klassifizierung für den Mann ist lückenhaft:

```
public function klassifiziere($bmi){
    if($bmi < 20){
        $this->
        set_klassifikation("Untergewicht");
    }elseif($bmi>=20 && $bmi <= 25 ){

    }elseif(..... ){

    }elseif(..... ){

    }else{

    }
}
```

Implementieren Sie die Methode klassifiziere(\$bmi) für den Mann und auch für die Frau.

## Optimaler BMI:

Alter	optimaler BMI
19-24	19-24
25-34	20-25
35-44	21-26
45-54	22-27
55-64	23-28
älter als 65	24-29

## 3. B.:

Erweiterung unserer Klassenbibliothek „lib.php“. *Implementieren einer weiteren Verhaltensweisen!*

Implementieren Sie die Methode bestimme\_opt(). Entscheiden Sie selbst an welcher Stelle im System und welche Kontrollstruktur für die Implementierung sinnvoll erscheint.

```
42 // Werte an das Objekt "spezielle Person" übermitteln
43 switch ($pOperator) {
44     case 1 :
45         $frau = new Frau();
46         $frau->set_gewicht( $pGewicht );
47         $frau->set_groesse( $pGroesse );
48         $frau->set_Alter( $pAlter );
49         $ausgabe = $bmi->berechne_bmi( $frau );
50
51         $frau->klassifiziere( $ausgabe );
52         $frau->bestimme_opt();
53         $klassifikation = $frau->get_klassifikation();
54         $min = $frau->get_min();
55         $max = $frau->get_max();
56         break;
57     case 2 :
```

## 5. A.:

Erweiterung der Aktions-Datei „bmi1.php“. *Implementierte Verhaltensweisen nutzen!*

→ Wir nutzen die implementierten Methoden aus der Bibliothek

Vervollständigen Sie die SWITCH-CASE Kontrollstruktur um den Fall „Mann“ (case 2 :).

```
switch ($pOperator) {
    case 1 :
```

```

...
    break;
case 2 :
...
    break;
default:
    echo "Bitte wählen Sie ein Ge-
schlecht!";
}

```

```

74 // Ausgabe (des Ergebnisses gerundet auf eine Kommastelle)
75 echo "<br />Gewicht: " . $pGewicht;
76 echo "<br />Größe: " . $pGroesse;
77 echo "<br />BMI: " .round ( $ausgabe, 1 );
78 echo "<br />";

```

Idealgewicht:

#### Idealgewicht Formel: (Größe in m)

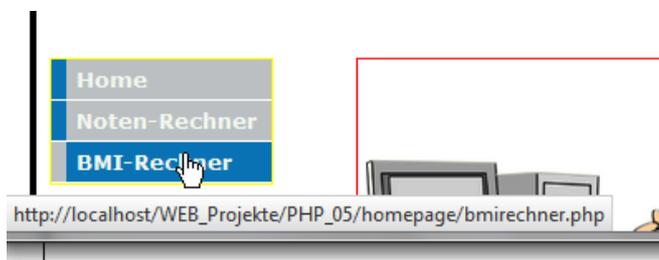
Idealgewicht Männer (unterer Bereich) =  $GRÖ\ddot{O}BE^2 \times 20$   
 Idealgewicht Männer (oberer Bereich) =  $GRÖ\ddot{O}BE^2 \times 25$

Idealgewicht Frauen (unterer Bereich) =  $GRÖ\ddot{O}BE^2 \times 19$   
 Idealgewicht Frauen (oberer Bereich) =  $GRÖ\ddot{O}BE^2 \times 24$

Hausaufgabe: Erweitern Sie die Anwendung

Implementieren Sie an geeigneter Stelle die Berechnungen des Idealgewichts für Männer und Frauen.

Testen Sie den BMI-Rechner:



Öffnen Sie die Anwendung im Browser.

#### Mein BMI-Rechner

Körpergewicht in kg:	<input type="text" value="56"/>
Körpergröße in m (.#.#):	<input type="text" value="1.70"/>
Alter in Jahren:	<input type="text" value="41"/>
<input checked="" type="radio"/> weiblich	<input type="radio"/> männlich
<input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>	

Geben Sie die Werte für das Körpergewicht und die Körpergröße, das Alter ein und wählen Sie das Geschlecht aus. Klicken Sie dann auf die Schaltfläche „Ausrechnen“.

20.1

Das berechnete Ergebnis wird berechnet und auf eine Kommastelle gerundet ausgegeben.

```
74 // Ausgabe (des Ergebnisses gerundet auf eine Kommastelle)
75 echo "<br />Gewicht: " . $pGewicht;
76 echo "<br />Größe: " . $pGroesse;
77 echo "<br />BMI: " .round ( $ausgabe, 1 );
78 echo "<br />";
```

5. A.:  
Erweiterung der Aktions-Datei „bmi1.php“.  
*Ausgabe optimieren!*

Verändern Sie die Ausgabe und stellen Sie sicher, dass alle Eingabe und Ergebniswerte, wie unten in den Anwendungsfällen, angezeigt werden.

Testen Sie folgende Anwendungsfälle:

### Anwendungsfälle:



66
1.70
54
<input checked="" type="radio"/> weiblich <input type="radio"/> männlich
<input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>

Myra Bellamy ist eine Frau (54).

Sie gibt für ihr Gewicht den Wert „66“ und ihre Körpergröße „1.7“ ein. Legen Sie dann und erhält das folgende Ergebnis angezeigt:

93
1.86
42
<input type="radio"/> weiblich <input checked="" type="radio"/> männlich
<input type="button" value="Ausrechnen"/> <input type="button" value="Zurück"/>

Red Barklay ist ein Mann (42).

Er gibt für sein Gewicht den Wert „92“ und seine Körpergröße „1.86“ ein und erhält das folgende Ergebnis angezeigt:

<p>Gewicht: 66 Größe: 1.7 BMI: 22.8</p> <p>Geschlecht: 66 Klassifikation: Normalgewicht Min: 22 Max: 27</p>	<p>Gewicht: 93 Größe: 1.86 BMI: 26.9</p> <p>Geschlecht: 93 Klassifikation: Übergewicht Min: 21 Max: 26</p>
---	--

