



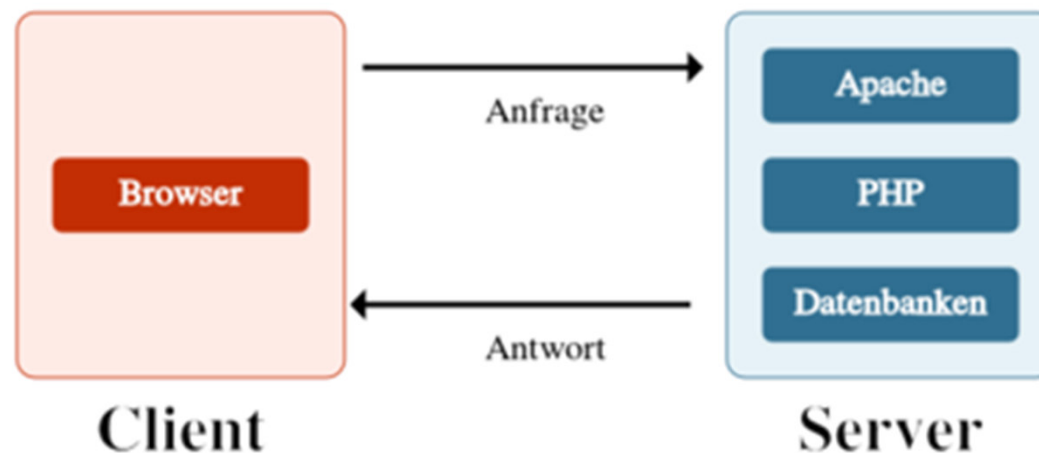
PHP

Hypertext Preprocessor

Grundlagen

Was ist PHP

- Serverseitige Skript-Sprache: kein kompilieren
- Erstellt dynamischen Webseiten: HTML-Seite wird beim Aufrufen erzeugt.
- Auswertung von Benutzer-Eingaben (z.B. Formulare)
- Anbindung der Datenbank



Welche Programmiersprache ist am populärsten?

StackOverflow Developer Survey

Frequency: Annually.

Methodology: 56,033 coders in 173 countries surveyed for 2016.

Rankings:

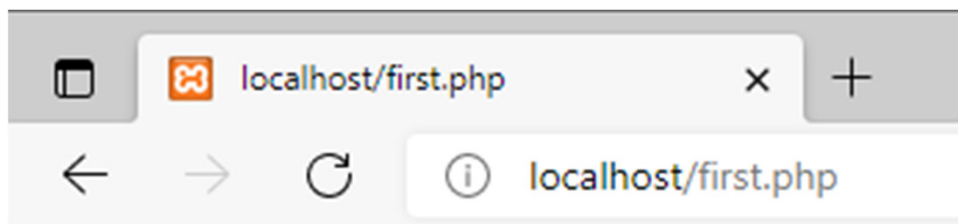
1. JavaScript
2. SQL (eliminated because it's not a conventional PL)
3. Java
4. C#
5. PHP
6. Python
7. C++



Erste Beispiele

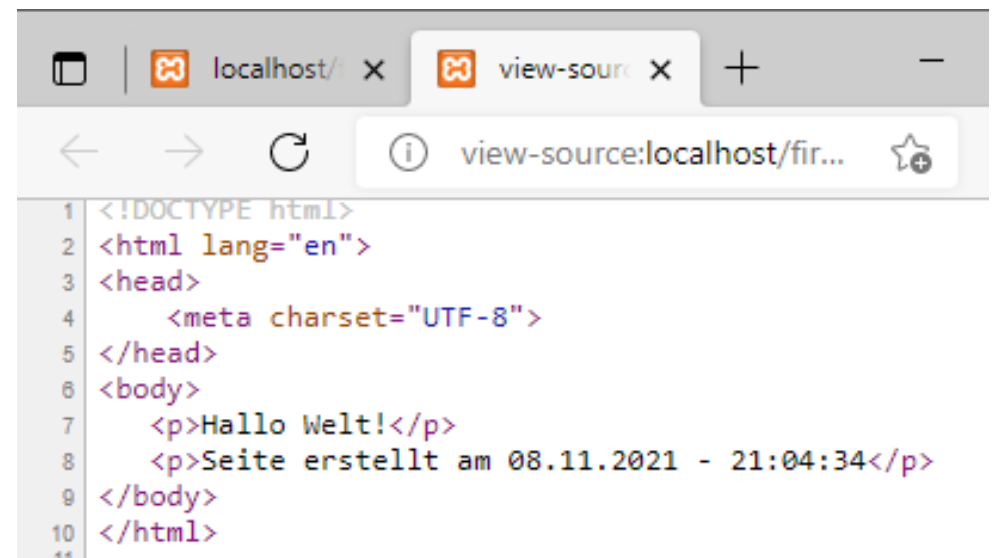
Beispiel first.php

```
<!doctype HTML>
<head>
  <meta charset="utf-8">
</head>
<body>
<p>Hallo Welt!</p>
<p>Seite erstellt am <?php echo date("d.m.Y - H:i:s"); ?></p>
</body>
</html>
```



Hallo Welt!

Seite erstellt am 08.11.2021 - 21:01:54



Entwicklungsumgebung XAMPP



XAMPP Apache + MariaDB + PHP + Perl

Was ist XAMPP?

XAMPP ist die beliebteste PHP-Entwicklungsumgebung


XAMPP ist eine vollständig kostenlose, leicht zu installierende Apache-Distribution, die MariaDB, PHP und Perl enthält. Das XAMPP Open-Source-Paket wurde für eine extrem einfache Installation und Nutzung eingerichtet.




Herunterladen

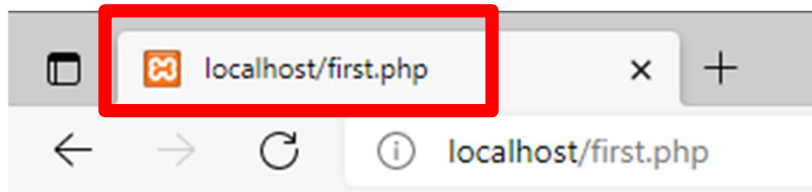
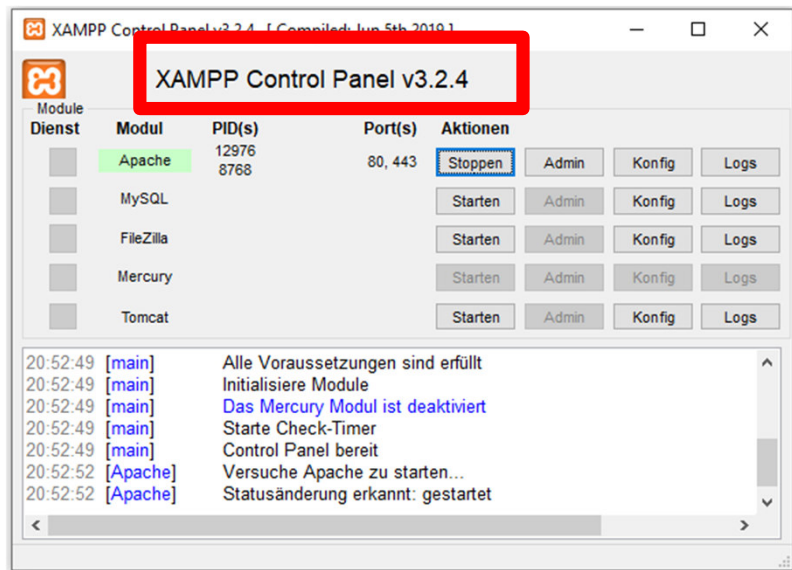
Klicken Sie hier für weitere Versionen

 XAMPP für **Windows**
8.0.12 (PHP 8.0.12)

 XAMPP für **Linux**
8.0.12 (PHP 8.0.12)

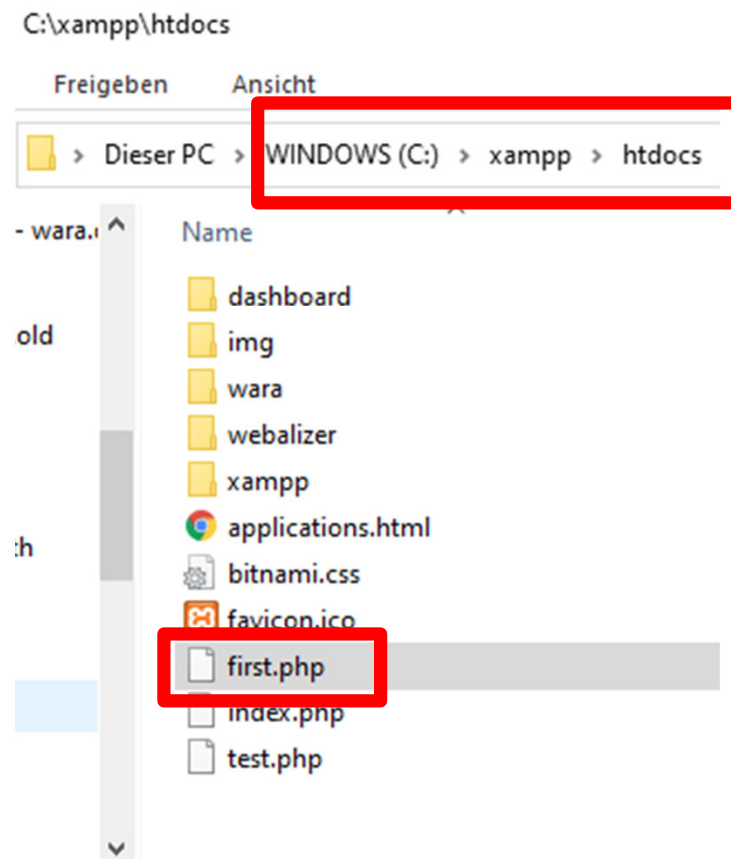
 XAMPP für **OS X**
8.0.12 (PHP 8.0.12)

PHP-Datei in Verzeichnis htdocs ablegen Server starten, localhost/first.php aufrufen

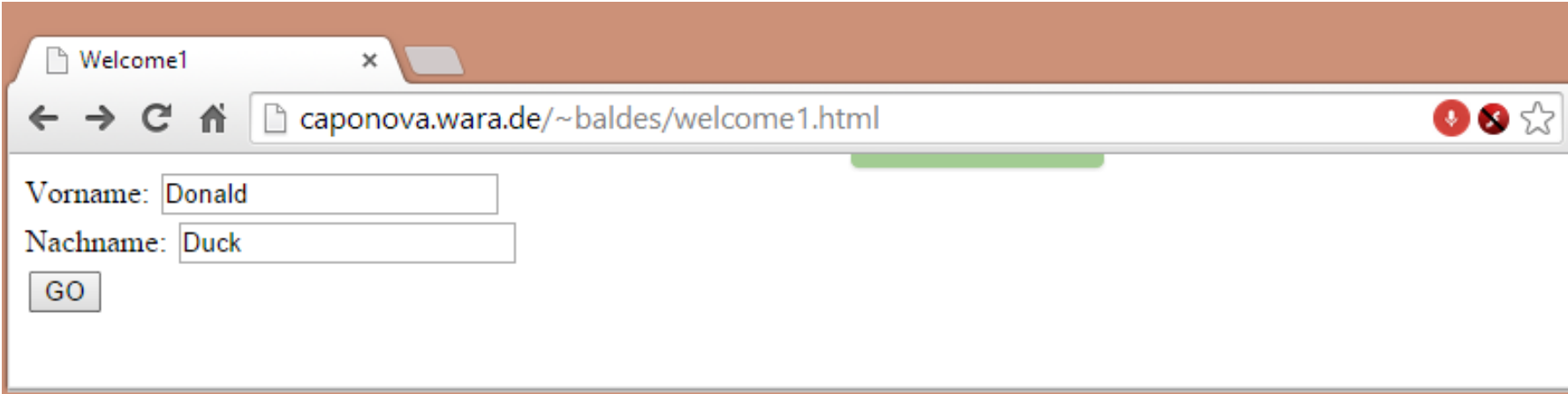


Hallo Welt!

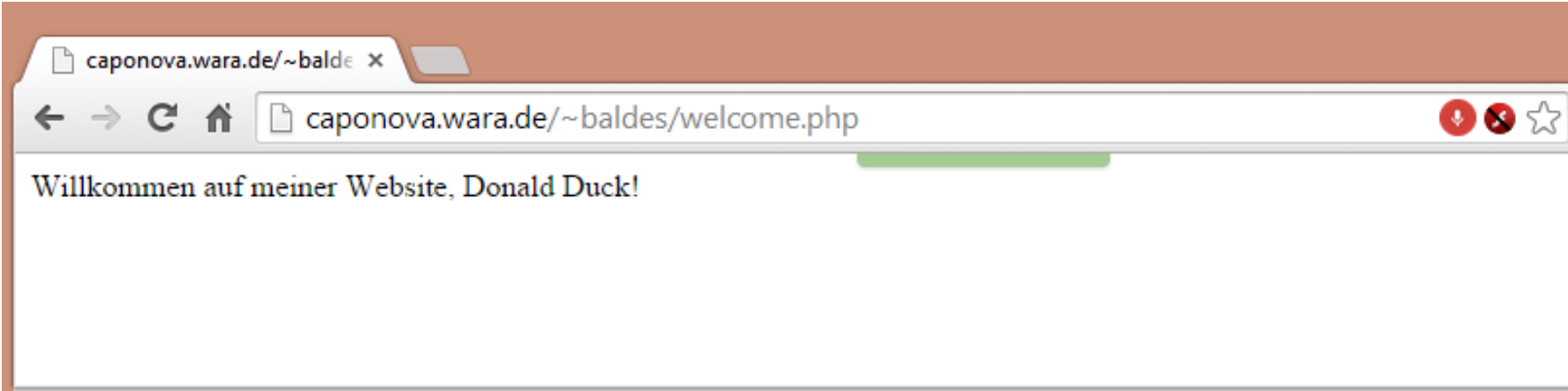
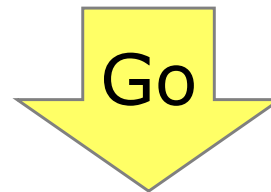
Seite erstellt am 08.11.2021 - 21:01:54



Beispiel: Benutzer begrüßen



Screenshot of a web browser window titled "Welcome1". The address bar shows the URL `caponova.wara.de/~baldes/welcome1.html`. The page content includes two input fields: "Vorname: Donald" and "Nachname: Duck". Below the input fields is a button labeled "GO".



Screenshot of a web browser window titled "caponova.wara.de/~balde". The address bar shows the URL `caponova.wara.de/~baldes/welcome.php`. The page content displays the message: "Willkommen auf meiner Website, Donald Duck!".

Beispiel: Benutzer begrüßen

welcome.html

...

```
<form action="welcome.php" method="post">  
  <label>Vorname: </label><input type="text" name="vorname" /><br/>  
  <label>Nachname: </label><input type="text" name="nachname" /><br/>  
  <input type="submit" value="GO" />  
</form>
```

...

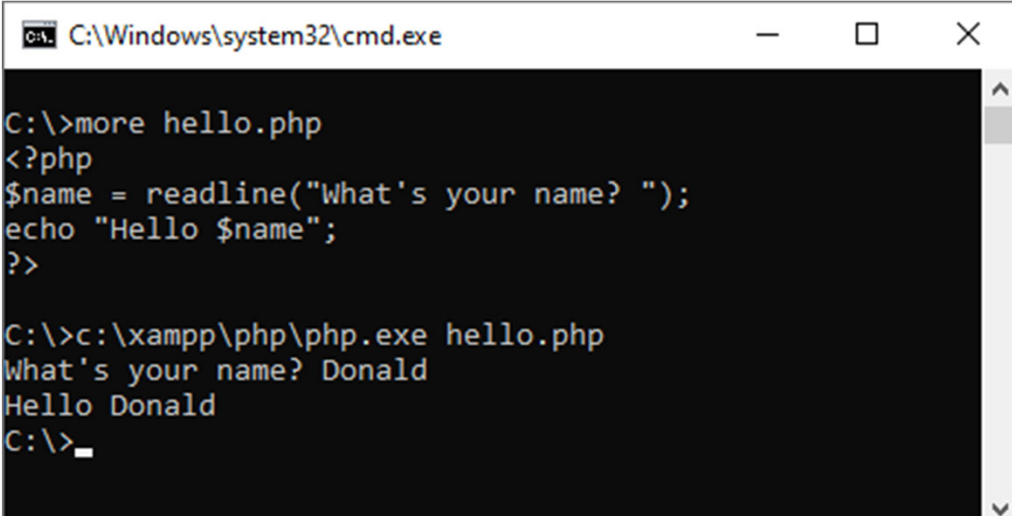
welcome.php

```
<body>  
  <p><?php  
    $vorname = $_REQUEST['vorname'];  
    $nachname = $_REQUEST['nachname'];  
    echo "Willkommen auf meiner Website, $vorname $nachname!";  
  ?></p>  
</body>
```

PHP in der Konsole

- Datei ausführen

```
php.exe hello.php
```



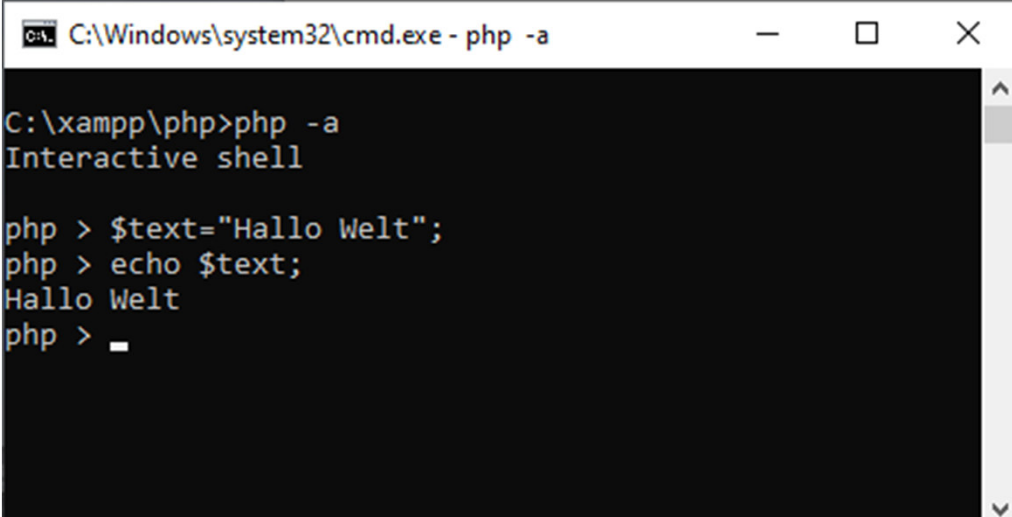
```
C:\Windows\system32\cmd.exe

C:\>more hello.php
<?php
$name = readline("What's your name? ");
echo "Hello $name";
?>

C:\>c:\xampp\php\php.exe hello.php
What's your name? Donald
Hello Donald
C:\>_
```

- Interaktiv:

```
php.exe -a
```



```
C:\Windows\system32\cmd.exe - php -a

C:\xampp\php>php -a
Interactive shell

php > $text="Hallo Welt";
php > echo $text;
Hallo Welt
php > _
```



Die Sprache

Besonderheiten

- `<?php ... ?>` Code auf Webseite ausführen
- `<?=$wert ?>` Wert in Webseite einfügen

- Variablen

- Beginnen mit einem `$`: `$zahl = 1;`
- Haben **keine Datentypen**, werden nicht definiert:

```
$zahl1 = 50; $zahl2 = 120; $summe=$zahl1+$zahl2;
```

- Werte im String eingesetzt:

```
echo "Das Ergebnis lautet $summe"
```

- oder mit `'.'` verknüpfen

```
echo 'Das Erg von $summe lautet' . $summe .  
"<br>\n";
```

Ausgabe und Variablen

- Ausgabe mit `echo` oder `print`.
- In HTML ist `
` notwendig für eine neue Zeile (Im Quellcode `\n`):
`echo "Das ist eine Zeile
\n"`
- `print_r`, `var_dump` gibt Wert und Datentyp aus
- Datentypen: String, Integer, Float, Boolean, Array, Object, NULL, Resource

```
$bool = true; // a boolean
$str  = "foo"; // a string
$int  = 12;   // an integer
```

```
echo var_dump($int); // int(12)
echo is_int($int);  // true
echo is_numeric($int); // true
echo gettype($bool); // prints out "boolean"
$int = (int)"1234"; // $int=1234
echo isset($int0); // false
```

```
$x == $y      Returns true if $x is equal to $y
$x === $y     Returns true if $x is equal to $y, and they are of the same type
```

```
$a_x == $a_y  Returns true if array $a_x and $a_y have the same key/value pairs
$a_x === $a_y Returns true if array $a_x and $a_y have the same key/value pairs
              in the same order and of the same types
```

Kontrollstrukturen und Schleifen

- If, else, elseif, switch
- While, do ... while, for, foreach (\$colors as \$value)

Arrays / Listen / Maps

```
$personArray = array();  
$personArray[0]="Peter"; $zahlArray[1]="Monika"; //...  
$personArray = ["Peter", "Monika", "Hans"];  
$personArray[] = "Tina"; // Array / Liste erweitern  
$tabArray = [[22.3, 18.5],[21.6, 19.7], [24.6, 20.1]];
```

```
$dict=array("house"=>"Haus", "cat"=>„Katze");  
//$dict = []; $dict["house"]="Haus"; $dict["cat"]="Katze";  
// array( [house] => Haus [cat] => Katze )
```

● Besonderheiten

- Keine Größenangaben des Arrays (Arrays = Listen = Maps).
- `for($x = 0; $x < count($personArray); $x++) {...}`
- `foreach($personArray as $person) {echo $person;}`
`foreach($dict as $key => $value) {echo "$key = $value";}`
- Funktionen: `sort`, `rsort`, `array_values`, ...

Funktionen

```
function addiere ($a, $b, $c=6) {  
    $sum = $a + $b + $c;  
    return $sum; }  
  
$erg = addiere(1,2,3);
```

- Funktionen mit Datentypen definieren

```
declare(strict_types=1); // strict requirement  
function addNumbers(float $a, float $b) : int {  
    return (int)($a + $b);  
}
```

- Pass by reference

```
function add_five(&$value) {  
    $value += 5;  
}  
  
$num = 2; add_five($num);
```


Klassen und Objekte

```
class Car {
  public $color;
  public $model;

  public function __construct($color, $model) {
    $this->color = $color;
    $this->model = $model;}

  public function message() {
    return "My car is a " .
      $this->color . " " . $this->model . "!";
  }
}

$myCar = new Car("black", "Volvo");
echo $myCar -> message();
```



Lokale und globale Variablen

Toplevel Variablen sind global

```
$x = 5; // global scope

function myTest() {
    // using x inside this function will generate an error
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();

echo "<p>Variable x outside function is: $x</p>";
```

Keyword global: Globale Variable in einer Funktion nutzen

```
$x = 5;  
$y = 10;  
  
function myTest() {  
    global $x, $y;  
    $y = $x + $y;  
}  
  
myTest();  
echo $y; // outputs 15
```

Keyword Static: Lokale Variablen in Funktionen als global erklären

```
function myTest() {  
    static $x = 0;  
    echo $x;  
    $x++;  
}
```

```
myTest();  
myTest();  
myTest();
```



Dynamische Webseiten

PHP-Anweisung vs PHP-Code und HTML mischen

```
<?php
for ($i=0;$i<10;$i++) {
    echo "<b>";
    echo "Zahl $i";
    echo "</b>\n";
}
?>
```

HTML mit echo erzeugen

```
<?php
for ($i=0;$i<10;$i++) {
    ?>
    <b>Zahl <?= $i ?></b>
<?php
}
?>
```

HTML und php mischen

The screenshot shows a web browser with two panes. The left pane displays the rendered output: a list of numbers from 0 to 9, each on a new line, with the number 0 being bolded. The right pane shows the source code, which is a mix of HTML and PHP. It includes a meta charset declaration, a body tag, and a series of PHP echo statements that output the numbers 0 through 8, with the first echo statement outputting a bolded number 0.

Zähler bis 100

Beide Methoden ausprobieren



A screenshot of a web browser window. The address bar shows 'localhost/zaehler1.php'. The page content displays a counter from 0 to 99, with the last number '9' partially cut off. Below the counter, it shows the current date: 'Heutiges Datum:09.11.2021'.

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20  
28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44  
52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68  
76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92  
Heutiges Datum:09.11.2021
```

A screenshot of a web browser window showing the source code of the PHP counter script. The address bar shows 'view-source:localhost/za...'. The code is displayed in a monospaced font with syntax highlighting. It includes HTML meta tags for charset, viewport, and http-equiv, and a title 'Zähler mit PHP erzeugt'. The body contains an h1 tag with the same text and a paragraph with numbers 0 through 5.

```
1 <!DOCTYPE html>  
2 <html lang="en">  
3 <head>  
4     <meta charset="UTF-8">  
5     <meta name="viewport" content="width=device-width, initial-  
6     <meta http-equiv="X-UA-Compatible" content="ie=edge">  
7     <title>Zähler mit PHP erzeugt</title>  
8 </head>  
9 <body>  
10 <h1>Zähler mit PHP erzeugt</h1>  
11 <p>  
12 0  
13 1  
14 2  
15 3  
16 4  
17 5
```


Formularauswertung

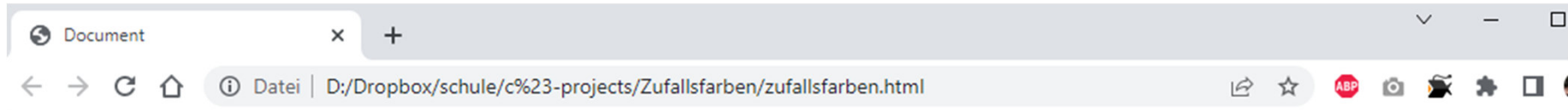
- formular.html

```
<form action="auswertung.php" method="post">
  <input type="text" name="name">
  ...
</form>
```

- auswertung.php

```
<?php
  if (isset($_REQUEST["name"]) &&
      $_REQUEST["name"]!="") {
    $name = $_REQUEST["name"];
    echo "Hallo $name!";}
?>
```

Tabelle mit Zufallsfarben generieren



Zufallszahlen

179	45	200	209	218	27	145	80	249	202	187	121	154	190	162	207	179	181	189	79	35	190	182	108	156	22	212	33	65	199								
129	148	102	11	195	174	83	80	203	198	184	230	11	236	226	140	201	59	34	245	213	240	23	197	127	3	148	80	29									
92	223	196	100	49	176	250	154	241	205	206	199	116	249	237	216	166	221	187	35	232	170	116	65	112	6	161	85	220	239								
223	80	151	7	175	12	185	190	199	139	254	97	62	29	75	131	94	171	219	132	113	234	139	249	188	245	201	209	166	121								
136	134	211	164	90	55	165	114	161	234	208	133	146	156	12	17	84	201	144	126	46	27	70	194	178	39	133	104	220									
74	33	26	194	218	123	118	142	48	53	56	93	233	75	32	153	48	56	125	119	195	222	128	148	245	176	139	134										
174	1	77	179	244	44	43	141	17	252	89	146	250	54	23	83	92	185	82	3	54	248	135	253	173	176	106	97	244	196								
37	198	137	180	25	98	213	0	113	132	96	219	157	201	221	138	154	54	228	250	38	88	64	123	223	71	55	93	241	218	78	193	65	79	135	104	48	88
139	176	165	39	215	168	178	180	100	225	185	193	39	80	154	100	208	79	88	236	69	143	127	184	215	152	194	1	230	63	42	189	232	250	186	243	24	24
89	52	116	199	7	9	183	127	49	61	75	150	190	134	110	174	49	186	17	4	49	143	171	119	109	150	152	177	201	172	138	14	62	123	58	60	65	66
144	95	63	43	2	97	215	80	254	255	29	232	178	219	80	150	216	167	226	68	7	34	122	132	85	230	254	203	226	111	41	49	27	108	166	63		
142	186	35	211	76	224	16	93	25	127	120	63	78	139	160	79	15	210	215	233	10	18	71	161	244	246	206	141	239	234	76	163	44	192	51	224	199	147
204	161	193	112	241	89	167	215	252	231	113	193	64	252	183	69	73	49	238	28	226	0	195	22	233	58	113	62	224	86	112	124	68	123	242	91	36	109
191	27	217	171	117	164	154	49	41	255	58	152	22	54	23	108	185	58	197	60	123	26	210	30	116	31	98	196	31	250	219	184	98	134	121	225	153	128
96	12	143	90	177	240	196	79	186	177	150	16	156	220	87	15	231	53	221	181	165	26	99	83	49	66	1	151	68	73	112	110	47	70	142	210	84	
84	162	194	119	41	215	180	240	118	43	54	223	250	25	53	97	200	71	128	92	20	12	17	191	97	230	28	15	58	37	3	110	228	205	80	226	199	74
142	192	157	7	255	234	231	22	204	98	0	199	244	106	57	24	82	219	73	37	224	148	57	96	13	22	45	107	187	222	214	72	20	62	225	22	103	240
6	73	147	5	227	88	44	122	43	91	224	216	90	83	229	99	108	2	137	110	80	221	107	78	181	78	25	203	41	150	160	195	245	13	83	249	20	207
83	11	118	213	218	212	49	157	178	38	1	231	72	46	92	108	132	121	214	167	124	89	56	174	97	22	50	76	112	208	207	144	106	220	247	222	214	15
33	78	62	122	215	215	145	118	143	223	132	33	33	129	28	53	254	226	217	144	191	205	35	71	213	42	60	13	168	196	93	215	3	214	131	45	100	208

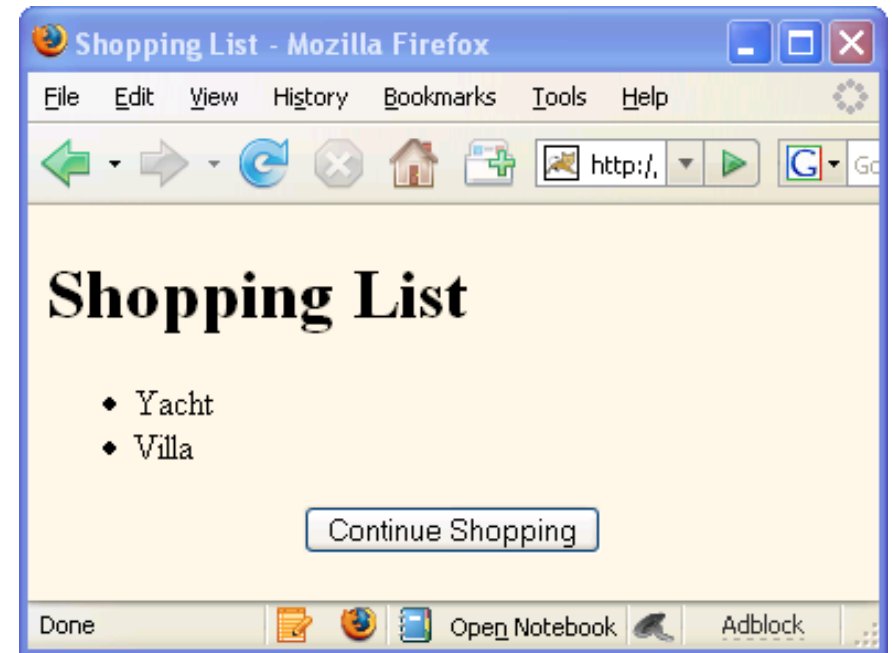
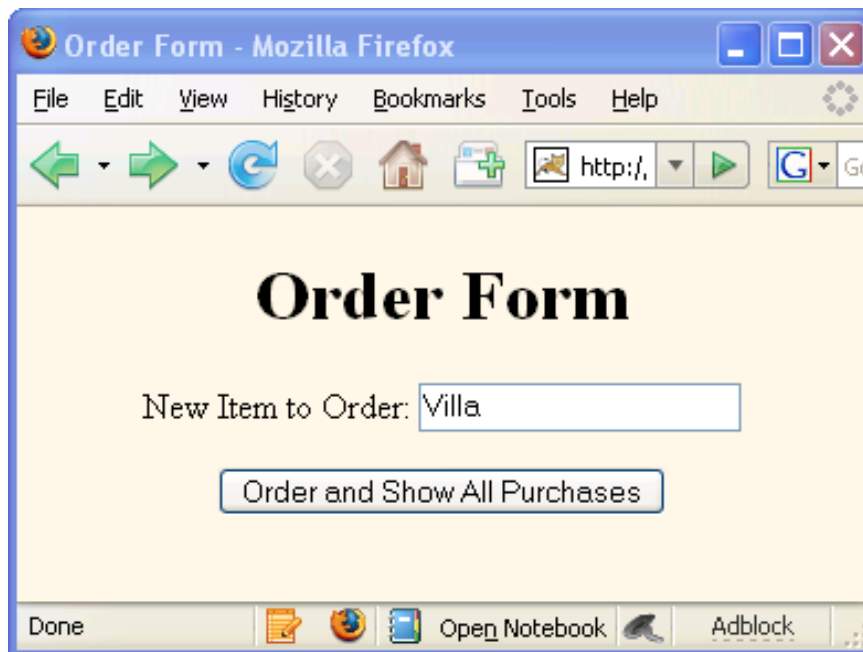


Globale Variablen Session

Session-Variable definieren

```
<h1>Zähler mit Session Variable</h1>
<p>
<?php
// Session starten
session_start();
// Variable beim ersten Mal anlegen
if (!isset($_SESSION['counter'])) $_SESSION['counter']=0;
// Variable nutzen
$counter=$_SESSION['counter'];
?>
<?php
    $counter++;
    echo($counter . "<br>\n");
    // Neuer Wert wieder in Session-Variable speichern
    $_SESSION['counter']=$counter;
?>
<p>F5 drücken zum Hochzählen!</p>
```

Einkaufs-Liste



Einkaufsliste

Alles auf einer Seite

A screenshot of a web browser window. The browser's address bar shows the URL "Datei | D:/Drop...". The page content is on a light yellow background and includes the following elements:

- A heading "Shopping List" in a large, bold, serif font.
- A bulleted list with two items: "Villa" and "Yacht".
- A heading "Order Form" in a large, bold, serif font.
- A text label "New Item to Order:" followed by a text input field containing the word "Yacht".
- A button labeled "Order and Show All Purchases".



Dateizugriff

Dateien lesen / schreiben / löschen

- Datei einlesen: **file_get_contents**

```
error_reporting(0); //Oder if(file_exists(...))
$content = file_get_contents("datei.txt");
```

- Datei zeilenweise ausgeben

```
if (isset($content)) {
    //Zeilenweise aufteilen
    //In c#: string[] a = c.Split("\n");
    $lines= explode("\n", $content);
    foreach($lines as $line) {
        echo "$line<br>\n";}}}
```

- Dateien schreiben: **file_put_contents**

```
file_put_contents("datei2.txt", "Mein Inhalt");
file_put_contents("datei2.txt", "Neue Zeile",
    FILE_APPEND);
```

- Datei löschen: **unlink**

```
unlink("todo.txt");
```


csv-Datei als Tabelle ausgeben

```
<?php
$file="adressen.txt";
if (file_exists($file))
    $content = file_get_contents($file);
if (isset($content)) {
    echo "\n";
    echo "<table border=1>\n";
    $lines = explode("\n",$content);
    foreach($lines as $line) {
        $spalten=explode(";", $line);
        //echo "$line <br>\n";
        echo "<tr>\n";
        for($i=0;$i<count($spalten);$i++) {
            echo "<td>$spalten[$i]</td>";
        }
        echo "</tr>\n";
    }
    echo "</table>\n";
}
?>
```

Thomas;Müller;E3FS1T
Tina;Maier;E3FS2T
Ivan;Kasparov;E3IT1T

php

Thomas	Müller	E3FS1T
Tina	Maier	E3FS2T
Ivan	Kasparov	E3IT1T