***FORMULARE - THEORIE***

PHP-Entwicklungsumgebung für den eigenen PC einrichten

> XAMPP noch auf mein Surface installiert.

> GIT für Windows mit Übungen und Tutorials war Auftrag für PSE vergangene Woche.

HTTP

***> HTTP - Status Messages***

- Der Server gibt bei jeder Anfrage immer eine Nachricht zurück, sogenannte http-Statusnachrichten. Tritt ein Fehler auf, wenn diese üblicherweise HTML-Fehlermeldungen genannt.

- kleiner Auszug aus der Liste von Statusmeldungen:

> 1xx: Informationen: zB.: 100 Continue

> 2xx: Erfolgreich: zB.: 200 OK

> 3xx: Weiterleitung: zB.: 302 Found

> 4xx: Client-Fehler: zB.: 401 Unauthorized

> 5xx: Serverfehler: zB.: 502 Bad Gateway

***> HTTP – Request Methods***

- HTTP funktioniert als Anforderungs-Antwort-Protokoll zwischen Client und Server

- HTTP Methoden :

> GET wird verwendet, um Daten von einer angegebenen Ressource anzufordern. Abfragezeichenfolge (Name/Wert-Paare) werden in der URL einer GET-Anfrage gesendet.

> HEAD fast identisch mit GET, jedoch ohne Antworttext ( zum Prüfen vor einer großen Abfrage )

> POST wird verwendet, um Daten an einen Server zu senden, um eine Ressource zu erstellen bzw. zu aktualisieren. Die an den Server gesendeten Daten werden im Request-Body des http-Request gespeichert.

> PUT. Unterschied zu POST ist, dass mehrfache Aufrufe derselben PUT-Anforderung zum gleichen Ergebnis führt.

> DELETE löscht die angegebene Ressource.

> OPTIONS beschreibt die Kommunikationsoptionen für Zielressource.

> Unterschiede zwischen GET und POST

|  |  |
| --- | --- |
| GET | POST |
|  |  |
| können zwischengespeichert werden | werden nie zwischengespeichert |
| bleiben im Browserverlauf | bleiben nicht im Browserverlauf |
| können mit Lesezeichen versehen sein | können nicht mit einem Lesezeichen versehen sein |
| Längenbeschränkungen | keine Längenbeschränkungen |

HTML

***> HTML Forms***

- HTML-Formulare werden verwendet um Benutzereingaben zu sammeln. Diese werden meistens zur Verarbeitung an einen Server gesendet.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> ein Container für verschiedene Arten von Eingabeelementen

***> Attribute***

> action-Attribut -> definiert Aktion die beim Absenden des Formulars ausgeführt werden soll. Normalerweise werden Daten an eine Datei auf dem Server gesendet. Wird action-Attribut weggelassen, wird die aktion auf die aktuelle Seite gesetzt.

> target-Attribut -> gibt an, wo die empfangene Antwort nach dem Senden angezeigt werden soll ( zB in einem neuen Fenster oder Tab ). Standard ist aktuelles Fenster.

> method-Attribut -> gibt die http-Methode an beim Senden an. GET oder POST. Standard ist GET.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> autocomplete-Attribut -> für automatische Vervollständigung vom Benutzer im Browser

***> Elemente***

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> input-Element -> wird je nach Eingabetyp (type) auf verschiedene Arten angezeigt.

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

-> Attribute für input-Element:

- value -> definiert Anfangswert für Eingabefeld

- readonly -> schreibgeschützt

- disabled -> deaktiviertes Eingabefeld, wird beim Absenden nicht gesendet.

- size -> sichtbare Breite des Eingabefeldes, Standard sind 20 Zeichen

- maxlength -> maximal zulässige Zeichenanzahl für Eingabe

- min und max -> minimaler und maximaler Eingabewert

> label-Element -> für Screenreader-User, hilft Usern beim Klicken auf kleine Bereiche. Das for-Attribut sollte identisch mit dem id-Attribut des input-Elements sein um sie miteinander zu verbinden

-> select-Element -> definiert eine Dropdown Liste

BOOTSTRAP

> w3schools wurde im letzten Semester schon besprochen, gut zum Nachlesen.

> Video angeschaut

PHP-Tutorial

> hab ein eigenes Tutorial auf UDEMY.

**Grundlegendes**

> PHP ( Hypertext Preprocessor ) ist eine Server-Skriptsprache zum Erstellen dynamischer und interaktiver Webseiten.

> PHP-Dateien können Text, HTML, CSS, JS und PHP-Code enthalten.

> PHP-Code wird auf dem Server ausgeführt und das Ergebnis wird als reines HTML an den Browser zurückgegeben.

> PHP kann

- Dateien auf dem Server erstellen, öffnen, lesen, schreiben, löschen und schließen

- Formulardaten sammeln

- Cookies senden und empfangen

- Daten in Datenbank hinzufügen, löschen und ändern

- Benutzerzugriffe steuern

- Daten verschlüsseln

***Syntax***

> PHP-Skript beginnt mit <?php und endet mit ?>

> Groß- und Kleinschreibung wird nicht beachtet ( ausser bei festgelegten Variablennamen )

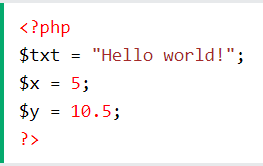
***Kommentare***

> Es gibt wie in Java zB einzelne Kommentare ( // ) und mehrzeilige Kommentare ( /\* \*/ )

***Variablen***

> PHP ist eine lose typisierte Sprache. PHP ordnet der Variablen abhängig vom Wert automatisch einen Datentyp zu. In PHP 7 wurden Typdeklarationen hinzugefügt.

> Eine Variable beginnt mit einem $-Zeichen gefolgt vom gewünschten Namen. Sie muss mit einem Buchstaben oder dem Unterstrich beginnen. Sie darf nicht mit einer Zahl beginnen und darf nur alphanumerische Zeichen und Unterstriche enthalten.



> PHP hat 3 Gültigkeitsbereiche für Variablen:

- local -> Wird eine Variable innerhalb einer Funktion deklariert hat sie einen LOCAL SCOPE -> kann nur innerhalb aufgerufen werden

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

- global -> Wird eine Variable ausserhalb einer Funktion deklariert hat sie einen GLOBAL SCOPE -> kann nur ausserhalb aufgerufen werden

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> mit den Schlüsselwort global kann innerhalb einer Funktion auf eine globale Variable zugegriffen werden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> In PHP werden auch alle globalen Variablen in dem Array $GLOBALS gespeichert und auch so innerhalb einer Funktion aufgerufen werden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

- static -> Da beim Abschließen/Ausführen einer Funktion alle Variablen wieder gelöscht werden gibt es noch das Schlüsselwort static. Damit ist es möglich die Informationen der Variable beim nochmaligen Aufruf einer Funktion beizubehalten. Die Variable bleibt lokal für die Funktion.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

***Echo und Print***

> sind beide mehr oder weniger gleich und werden verwendet um Daten auf dem Bildschirm anzuzeigen.

> print kann noch in Ausdrücken verwendet werden da es einen Rückgabewert von 1 hat.

***Datentypen***

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Die Funktion var\_dump() gibt den Datentyp und den Wert zurück.

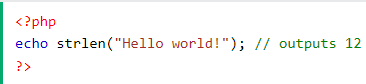
> Objekt

Ein Bild, das Text enthält.

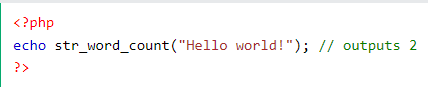
Automatisch generierte Beschreibung

***String-Funktionen***

> strlen() – Gibt die Länge eines Strings zurück



> str\_word\_count() – Zählt die Wörter in einem String

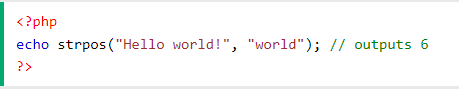


> strrev() – Kehrt einen String um

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> strpos() – Sucht man nach einem Text innerhalb eines Strings; (erste Position ist 0 nicht 1)



> str\_replace – Ersetzt Text innerhalb eines Strings



> vollständige String-Referenz : <https://www.w3schools.com/php/php_ref_string.asp>

***Numbers***

> Weist man einer Variablen eine ganze Zahl zu ist dieser Typ automatisch ein Integer. Wird derselben Variablen eine Zeichenfolge zugeordnet, ändert sich der Typ in String -> automatische Konvertierung.

> Integer

- muss eine Ziffer haben, darf keinen Dezimalpunkt haben, kann + oder – sein, kann in 3 Formaten angegeben werden: dezimal, hexadezimal, oder oktal

- vordefinierte Konstanten:

- PHP\_INT\_MAX – größter unterstützter Integer

- PHP\_INT\_MIN – kleinster unterstützter Integer

- PHP\_INT\_SIZE – die Größe eines Integers in Bytes

- Funktionen zum Prüfen ob Typ der Variable ein Integer ist:

- is\_int()

- is\_integer()

- is\_long()

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

***MATH***

> mit einer Reihe von mathematischen Funktionen können Aufgaben mit Zahlen ausgeführt werden.

> pi() – Gibt den Wert von PI zurück

> abs() – Gibt absolut (positiven) Wert zurück

> squrt() – Gibt die Quadratwurzel einer Zahl zurück

> round() – Rundet eine Gleitkommazahl auf ihre nächste Ganzzahl um

> rand() – Generiert zufällige Zahlen

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> vollständige Math-Referenz: <https://www.w3schools.com/php/php_ref_math.asp>

***Konstanten***

> Konstanten sind wie Variablen. Nach dem Definieren können sie allerdings nicht mehr geändert oder undefiniert werden.

> um eine Konstante zu erstellen wird die define() Funktion verwendet.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

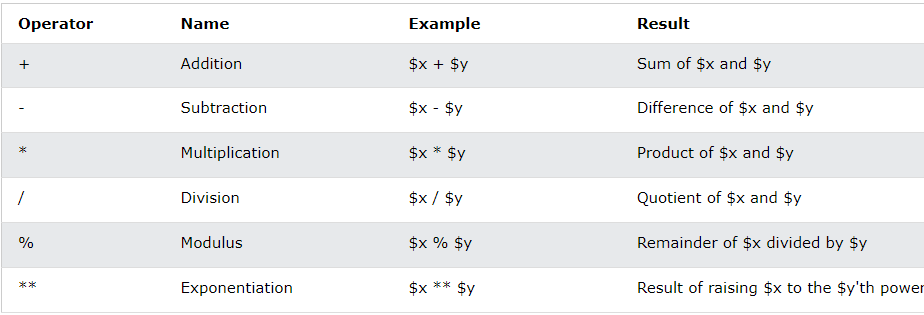
***OPERATOREN***

> werden verwendet, um Operationen mit Variablen und Werten auszuführen.

> sie werden in folgende Gruppen unterteilt.

ARITHMETIC OPERATORS

> werden mit numerischen Werten verwendet, um gängige arithmetische Operationen wie Addition, Subtraktion usw durchzuführen



ASSIGNMENT OPERATORS

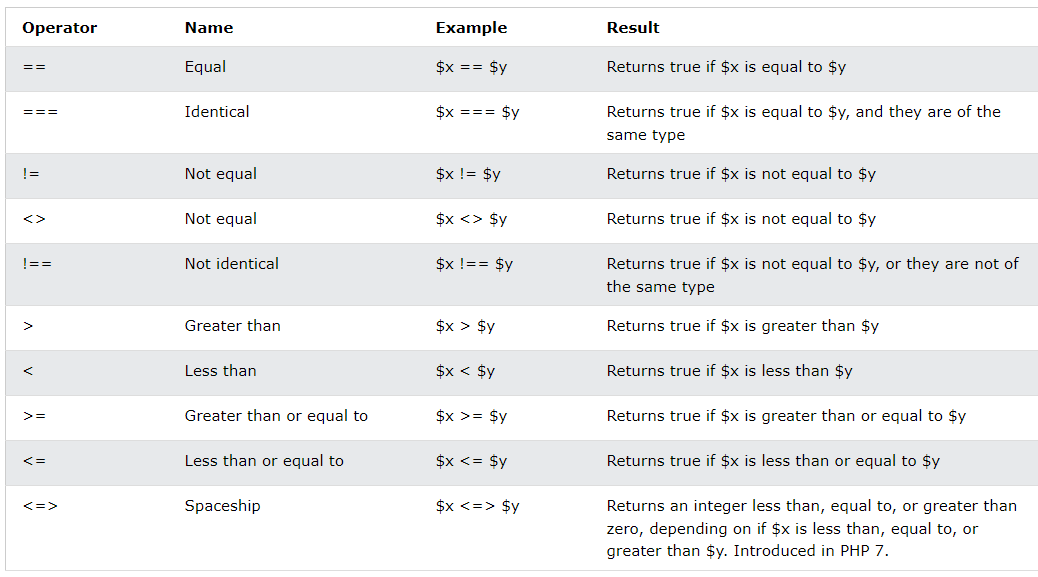
> werden mit numerischen Werten verwendet, um einen Wert in eine Variable zu schreiben.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

COMPARISON OPERATORS

> werden verwendet, um zwei Werte ( Zahl oder Zeichenfolge ) zu vergleichen



INCREMENT / DECREMENT OPERATORS

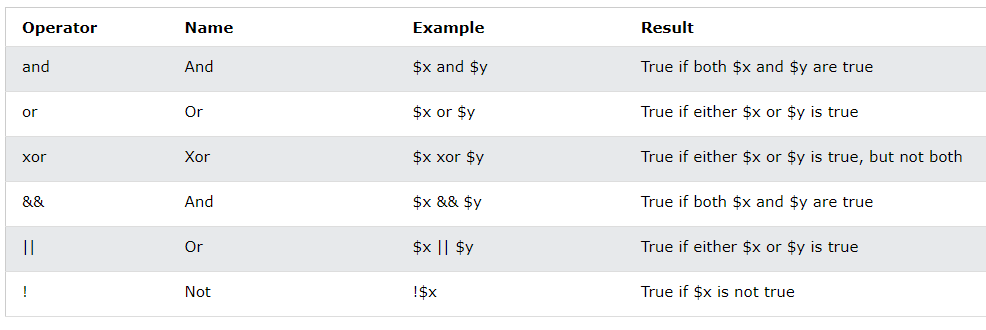
> werden verwendet, um den Wert einer Variable zu erhöhen (increment) oder zu verringern (decrement)

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

LOGICAL OPERATORS

> werden verwendet, um bedingte Anweisungen zu kombinieren.



STRING OPERATORS

> hat 2 Operatoren die speziell für Strings entwickelt wurden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

ARRAY OPERATORS

> um Arrays zu vergleichen

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

CONDITIONAL ASSIGNMENT OPERATORS

> um einen Wert abhängig von Bedingungen festzulegen

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

***If…else…elseif und switch – Anweisungen***

> werden verwendet, um verschieden Aktionen basierend auf verschiedenen Bedingungen auszuführen.

> sind mehr oder weniger ident mit JAVA.

> Syntax

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

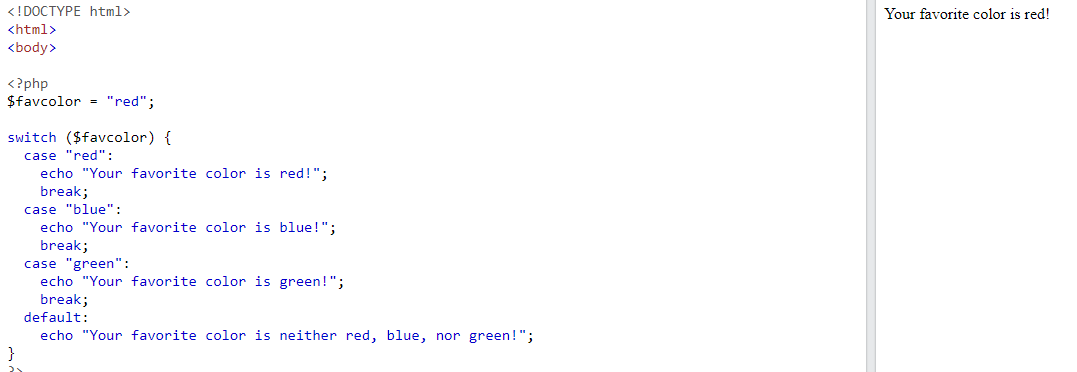
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung





***LOOPS***

> werden verwendet, um den gleichen Codeblock immer wieder auszuführen, solange eine bestimmte Bedingung zutrifft.

> sind mehr oder weniger ident mit JAVA

> Syntax

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung



Ein Bild, das Text enthält.

Automatisch generierte Beschreibung



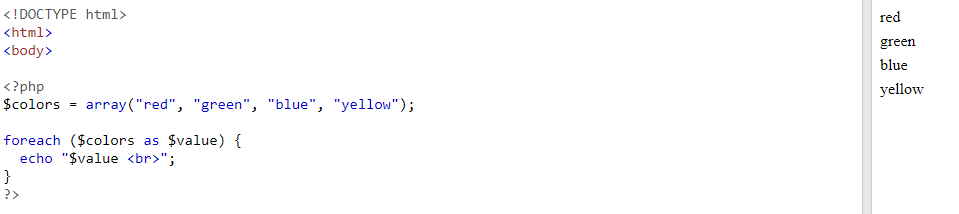
Ein Bild, das Text enthält.

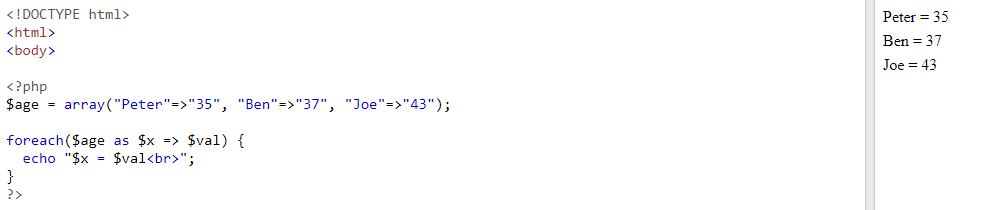
Automatisch generierte Beschreibung



Ein Bild, das Text enthält.

Automatisch generierte Beschreibung





***FUNKTIONEN***

> PHP hat über 1000 eingebaute Funktionen. Zusätzlich kann man eigene benutzerdefinierte Funktionen erstellen.

> Vollständige Funktionen-Referenzen: <https://www.w3schools.com/php/php_ref_overview.asp>

> Eine Funktion ist ein Block von Anweisungen, der in einem Programm wiederholt angewendet werden kann. Sie wird beim Laden einer Seite nicht automatisch ausgeführt, sondern muss direkt aufgerufen werden.

> Syntax: eine Funktion muss mit einem Buchstaben oder einem Unterstrich beginnen. Case-Insensitive.

Ein Bild, das Text enthält.

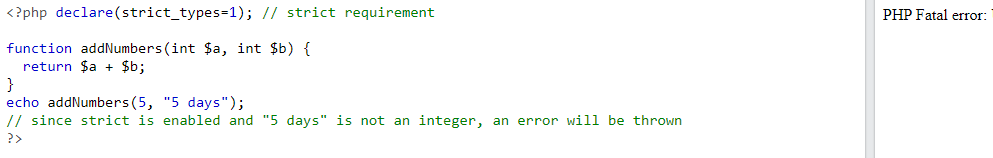
Automatisch generierte Beschreibung

> Ein oder mehrere Argumente (wie Variable) können an Funktionen übergeben werden.



> der Datentyp wird von PHP auch hier automatisch zugeordnet.

> in PHP7 wurden Typendeklarationen hinzugefügt. Im Folgenden Beispiel wird dies gezeigt.



> Die strict-Deklaration erzwingt die beabsichtigte Verwendung der Dinge.

> Es können auch Standardparameter verwendet werden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> auch in PHP können Funktionen eine Rückgabewert haben. Dieser kann entweder lose programmiert oder auch mit strict und einem Doppelpunkt im Funktionskopf wieder explizit angegeben werden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

***ARRAYS***

> Ein Array speichert mehrere Werte in einer einzigen Variable

> Es gibt 3 Arten von Arrays:

- Indexed arrays – Arrays with a numeric index

- Associative arrays – Arrays with named keys

- Multidimensional arrays – Arrays containing one or more arrays

> auch für Array gibt es verschiedene Funktionen.

> Vollständige Funktionen-Referenzen: <https://www.w3schools.com/php/php_ref_array.asp>

> Indexed Arrays:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Loop through an Indexed Array

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Associative Arrays: verwenden benannte Schlüssel die Sie ihnen zuweisen.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Loop through an Associative Array

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Multidimensional Arrays: enthalten ein oder mehrere Arrays

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Loop through an Multidimensional Array

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Sortieren von Arrays

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

***SUPERGLOBALS***

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

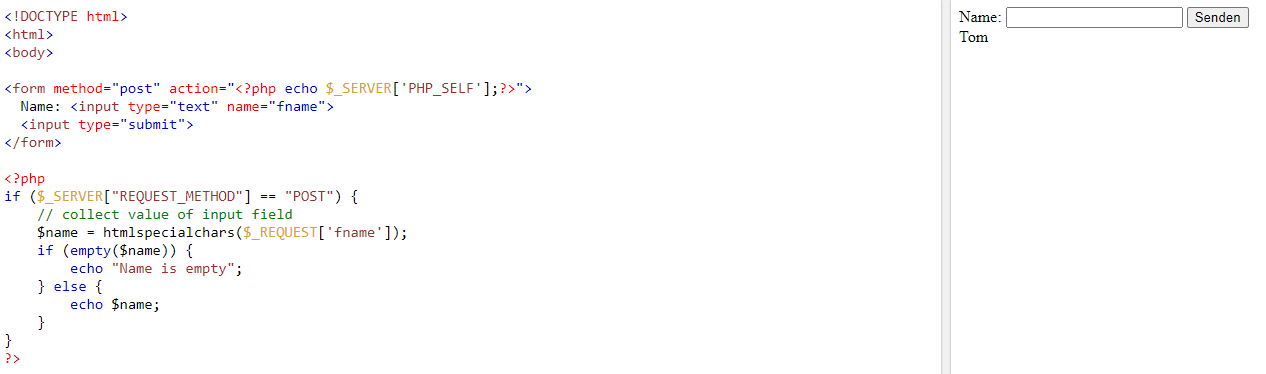
> $\_GLOBALS wird verwendet, um von überall im PHP-Skript auf globale Variablen zuzugreifen.

> $\_SERVER enthält die Informationen über Header, Pfade und Skriptspeicherorte

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> $\_REQUEST wird verwendet, um Daten nach dem Absenden eines HTML-Formulars zu sammeln.



> $\_POST wird verwendet, um Formulardaten nach dem Absenden eines HTML-Formulars mit method=“post“ zu sammeln. Wird auch häufig verwendet um Variablen zu übergebe

> $\_GET wird verwendet, um Formulardaten nach dem Absenden eines HTML-Formulars mit method=“get“ zu sammeln.

***REGULÄRE AUSDRÜCKE (RegEx)***

> ist eine Folge von Zeichen, die ein Suchmuster bildet. Kann ein einzelnes Zeichen oder ein kompliziertes Muster sein.

> können verwendet werden, um alle Arten von Textsuch- und Textersetzungsoperationen durchzuführen.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Vollständige RegEx-Referenz: <https://www.w3schools.com/php/php_ref_regex.asp>

PHP – Formulare

***Handling***

> Superglobals $\_GET und $\_POST werden verwendet um Formulardaten zu sammeln.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> wird hier auf Senden geklickt, werden die Formulardaten zur Verarbeitung an eine PHP-Datei namens „welcome.php“ gesendet. Diese Daten werden mit der Methode HTTP POST gesendet.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> das gleiche Ergebnis könnte auch mit der HTTP-GET Methode erreicht werden.

> Das wichtigste fehlt bei beiden. Formulardaten müssen validiert werden um das Skript vor bösartigem Code zu schützen.

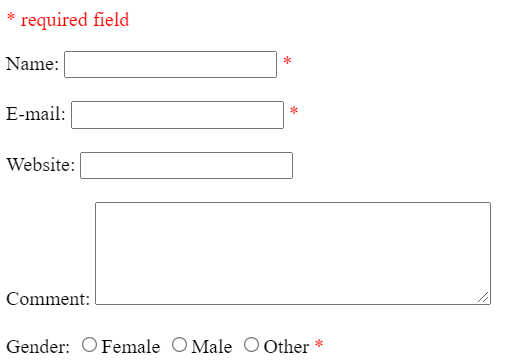
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

***VALIDATION***

> Die ordnungsgemäße Validierung der Formulardaten ist wichtig, um das Formular vor Hackern und Spammern zu schützen!

- PHP Form Validation Example



Name Required. + Darf nur Buchstaben und Leerzeichen haben



E-Mail Required. + Darf nur eine gültige email-Adresse haben ( @ und . )



Website Optional. Wenn angegeben, muss sie eine gültige URL sein



Comment Optional. Mehrzeiliges Input-Feld (textarea)

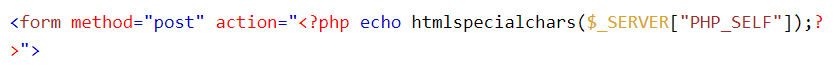


Gender Required. Man muss eines selektieren

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

- Das Formularelement



> Beim Absenden des Formulars werden die Formulardaten mit method=“post“ versendet

> $\_SERVER[„PHP-SELF“] -> Super globale Variable, gibt den Dateinamen des aktuell ausgeführten Skripts zurück

> htmlspecialchars() konvertiert Sonderzeichen in HTML-Entitäten. Also HTML Buchstaben wie < und > werden mit &lt; und &gt; ersetzt. Das verhindert Attacken von auf das Sktipt.

***REQUIRED***

> PHP-Pflichfelder können mit dem Attribut required als verpflichtend beschrieben werden.

> mit PHP-Code kann man mit Hilfe eines Skriptes, Fehlermeldungen generieren lassen wenn der Benutzer abschicken will obwohl verpflichtende Felder noch leer sind.

***URL/E-Mail***

> mit der Funktion preg\_match() durchsucht einen String nach Muster. So können Namensfelder validiert werden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> E-Mail-Adressen können mit Hilfe der Funktion filter\_var() überprüft werden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

> Die URL-Syntax kann zB auch mit preg\_match() überprüft werden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

***COMPLETE***

> Um die Werte der Benutzer in den Eingabefeldern weiterhin anzeigen zu lassen, nachdem auf Senden geklickt wurde, fügt man noch ein Skript hinzu mit dem value-Attribut.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung