# Erste Vorlesung

PHP Grundlagen

Serverseite mit direkter Datenbankanbindung (Code wird nicht an Client vermittelt)

Letztes Semester:  
Client schickt Anfrage mit Webseite an Server, Server liefert HTML, CSS, JS zurück, wird am Browser dargestellt

Dieses Semester:  
Client schickt Anfrage mit Webseite (mit PHP) an Webserver, macht Abfragen mit Datenbank, gibt dann HTML zurück, versteckt Datenbankabfragen vor Client, nur beim Server werden die neu ausgewertet

Am Server hat man z.B. ein login.php, darin ist HTML Text, darin gibt es PHP tags

Client bekommt das File, aber sieht den PHP Code nicht (sieht nur, was zu diesem Zeitpunkt ausgewertet wurde)

Interpreter tauscht PHP Blöcke mit dem aus, was sie zurückliefern

PHP Skriptsprache Hypertext Preprocessor

PHP-Syntax

Datentypen

Variablen werden mit einem $ Präfix gesetzt

Werden dynamisch typisiert

Automatisches Typecasting (weißt z.B. einem String einen Int zuweisen)

Explizites Typecasting (Int wird einem String zugewiesen, String soll aber String bleiben!)

Ist eine Variable gesetzt ‚isset($i)‘ liefert true oder false (wie NULL)

‚Unset($i)‘ löscht Variable

Typüberprüfung möglich mit is\_integer, etc…

Konstante Variablen beginnen nicht mit $ und werden groß geschrieben: define(„YO“, „Nice“);

Mit echo kann man Sachen raus schreiben

Vordefinierte Konstanten (typischerweise Strings)  
PHP\_VERSION welche Version  
PHP\_OS welches Betriebssystem  
PHP\_EOL End of Line  
E\_COMPILE\_ERROR (int)

$data = ‚hello‘ . PHP\_EOL;

Magische Konstanten: Evaluiert wo sie sich befinden, praktisch zum Debugen/Log-Files (werden zur Compile-Zeit ausgewertet)

Zwei Underlines! \_\_FILE\_\_

NICHT FRAGEN: Welche Klassen Operatoren gibt es

\*\* ist ^

== nur den Wert

=== auch den Datentyp

3 <=> 4 gibt Feedback -1, 0, 1 wenn kleiner, gleich, größer (für das Erste)

s$ = ‚Yo‘ . ‚My‘ . ‚Boy‘;

Mehrzeilige Strings mit einem Stringblock  
$multiBlock = STRINGBLOCK  
Hello  
You Fucker  
STRINGBLOCK;

„ erlaubt direkt im String Variablen s1$ = „s$ World“

‚ sieht alles als normalen String an

String Funktionen HTML:

Htmlspecialchars($str)  
Konvertiert HTML-Chars anders: & wird zu %26

Htmlentities($str)  
Konvertiert Sonderzeichen: ü, ä…

Nl2br($str)  
Konvertiert Zeilenumbrüche \n -> <br>

Rawurlencode($str) + rawurldecode($str)  
Codiert/Encondiert Strings für Queries (hello&world -> hello%26world)

Arrays: (dynamische Erweiterung)

Ordered Maps: Key/Value-Pairs, sind geordnet

Datentyp sind beliebig!

Indizierte Arrays: Mit Index 1, 2…(muss nicht durchgängig sein)

Assoziative Arrays: Eigener Key (Key selbst aussuchen, immer String)

Array Indizierung Folie 42

Assoziativ:  
array(‚Us‘ => „United States“)

Array Kombination Folie 44

Erstes Value ohne Key wird mit Index 0 gespeichert

Array Funktionen:  
SizeOf Anzahl der Elemente  
In\_Array Kommt ein Value vor

Print\_R kann ein Array raus schreiben