

## Vorlesung



University of Applied Sciences

# DBSP

**Unit 6**  
**PHP III**  
Datentypen

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

1



University of Applied Sciences



**Prof. Dr. rer. nat.  
Nane Kratzke**  
*Praktische Informatik und  
betriebliche Informationssysteme*

- Raum: 17-0.10
- Tel.: 0451 300 5549
- Email: [kratzke@fh-luebeck.de](mailto:kratzke@fh-luebeck.de)



@NaneKratzke

Updates der Handouts auch über Twitter #dbsp

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

2

## **Übergreifende Ziele der Lehrveranstaltung**



University of Applied Sciences

Client- und Serverseitige Entwicklung

PHP (Serverseitig)

JavaScript (Clientseitig)

„Hosten“ von Apps

Framework Erfahrungen

CMS (Drupal)

WebServices (Google-Maps)

jQuery

Datenbank-Integration

Berücksichtigung von Sicherheitsaspekten

HTML-Injections

SQL-Injections

Session Hijacking

Login-Systeme

Um sich weitere Web-Technologien autodidaktisch erarbeiten zu können.

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

3

## **Units**



University of Applied Sciences

**Unit 1**  
Cloud Computing IaaS

**Unit 2**  
CMS Drupal

**Unit 3**  
HTML und CSS

**Unit 4 - 7**  
PHP I - IV

**Unit 8**  
Sessions, Cookies, Formulare und Login-System

**Unit 9**  
JavaScript

**Unit 10**  
Drupal Module Development

**Unit 11**  
Datenmodellierung

**Unit 12 - 13**  
Datenbanken und SQL  
Vom Datenmodell zur Datenbank

**Unit 14**  
Datenbank-gestützte Web-Anwendungen

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

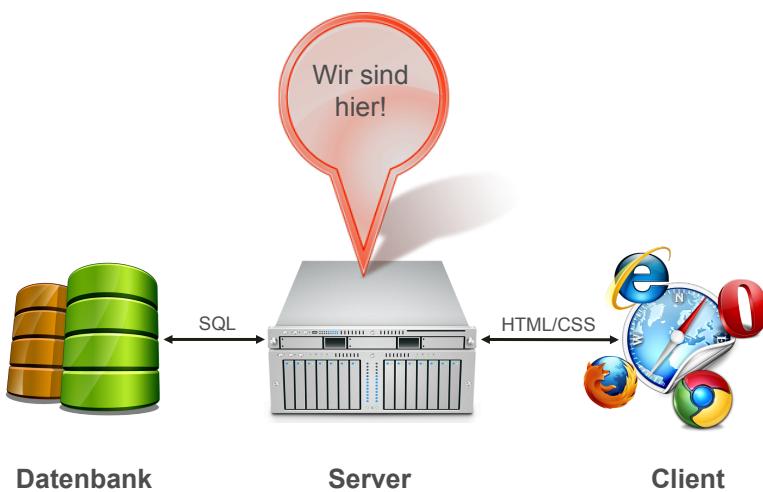
4

## Datenbank – Server – Client

Wo waren wir nochmal?



University of Applied Sciences



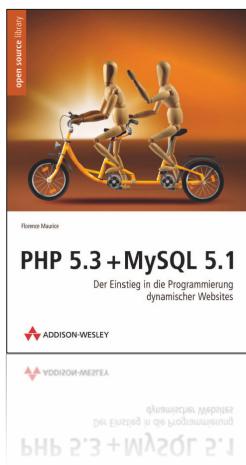
Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

5

## Zum Nachlesen ...



University of Applied Sciences



### Kapitel 4 – PHP Basics

Kapitel 4.5 bis 4.7

### Kapitel 5 – More Basics

Kapitel 5.1 bis 5.4

### Kapitel 6 – Funktionen für ...

Kapitel 6.2 und 6.3

### Kapitel 3 Variablen u. Datentypen

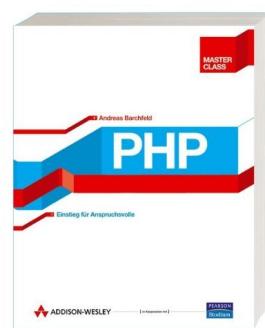
Kapitel 3.8 bis 3.21

### Kapitel 4 Programmstrukturen

Kapitel 4.1 bis 4.3

### Kapitel 5 Arrays

Kapitel 5.1 bis 5.3



Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

6

## Zum Nachlesen ...



University of Applied Sciences



### Bereitgestellte Skripte:

#### **PHP Programmierung**

- **Kapitel 5:** Kontrollstrukturen
- **Kapitel 6:** Datentypen und Datenstrukturen
- **Kapitel 7:** Operatoren

<http://praktische-informatik.fh-luebeck.de/node/39>

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

7

## Inhalte der PHP Units



University of Applied Sciences

- 1 • Laufzeitmodelle und Programmierparadigma
- 2 • Elemente eines PHP-Programms
- 3 • Datentypen
- 4 • Operatoren
- 5 • Kontrollstrukturen

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

8

## Die wesentlichen Datentypen in PHP



University of Applied Sciences

Bool  
Integer  
Float

Strings

Arrays

Type Casting

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

9

## Bool, Integer und Float



University of Applied Sciences

### Bool

- Wahrheitswerte
- TRUE oder FALSE

```
$v = TRUE;  
$w = FALSE;
```

### Integer

- Ganzzahlige und vorzeichen-behaftete Werte
- Angabe in:
  - Dezimalnotation
  - Oktalnotation (0) oder
  - Hexadezimal-notation (0x)

```
$v = 10; //Dez  
$v = 012; //Okt  
$v = 0xFF; //Hex
```

### Float

- Fließkomma-zahlen
- Punktnotation und wissen-schaftliche Schreibweise (Exponenten-darstellung) möglich.

```
$v = 1.5; // 1,5  
$v = 1.5e2; // 150,0  
$v = 7e-2; // 0,07
```

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

10

## Die wesentlichen Datentypen in PHP



University of Applied Sciences

Bool  
Integer  
Float

Strings

Arrays

Type Casting

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

11

## Strings



University of Applied Sciences

In PHP werden  
Zeichenketten durch  
String repräsentiert.

Ein String ist eine  
indizierte Liste von  
Zeichen.

D i e s i s t e i n S a t z .  
D I E S I S T E I N S A T Z .

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

12

## Einige Funktionen und Operatoren für Strings



University of Applied Sciences

Zeichenextraktion

Jedes Zeichen eines Strings kann über einen Index angesprochen werden.

Länge

Das erste Zeichen hat den Index 0.

Vergleichen

```
$str = "Dies ist ein Satz.";
```

Suchen

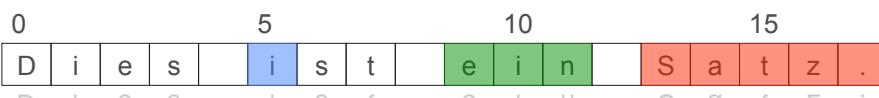
```
$substr = substr($str, 9, 3);
```

Ersetzen

```
$finstr = substr($str, 13, 5);
```

Zerlegen

Verketten



Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

13

## Einige Funktionen und Operatoren für Strings



University of Applied Sciences

Zeichenextraktion

Länge eines Strings entspricht der Anzahl an Zeichen eines Strings.

Länge

Die Länge eines leeren Strings ist 0.

Vergleichen

```
$str = "Dies ist ein Satz.";
```

```
$length = strlen($str);
```

18

Suchen

```
$empty = "";
```

```
$length = strlen($empty);
```

0

Ersetzen

Zerlegen

Verketten



Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

14

## Einige Funktionen und Operatoren für Strings



University of Applied Sciences

- Zeichenextraktion
- Länge
- Vergleichen
- Suchen
- Ersetzen
- Zerlegen
- Verketten

Mit den folgenden Funktionen lassen sich Strings inhaltlich auf Gleichheit vergleichen.

```
$str1 = "hallo";
$str2 = "HALLO";
$gleich = strcmp($str1, $str2);
#0
```

**\$gleich =strcasecmp(\$str1, \$str2);**

**0**

0

D	i	e	s
D	I	E	S

Ist str1 kleiner als str2 wird ein Wert < 0 zurückgegeben, ist str1 größer als str2 ein Wert > 0, und bei Gleichheit gibt die Funktion 0 zurück.

Prof. Dr. rer. nat. Nane Kratzke | 15

## Einige Funktionen und Operatoren für Strings



University of Applied Sciences

- Zeichenextraktion
- Länge
- Vergleichen
- Suchen
- Ersetzen
- Zerlegen
- Verketten

Mit den folgenden Funktionen lassen sich Zeichenketten in Strings finden:

```
$str = "Dies ist ein Satz";
$i = strpos($str, "Satz");
13
$i = strpos($str, "existiert nicht");
FALSE – d.h. Suchstring wurde nicht gefunden
$i = strpos($str, "i");
1
$i = strpos($str, "i", 2);
5
```

0

D	i	e	s	i	s	t	e	i	n	S	a	t	z	.
D	I	E	S	I	S	T	E	I	N	S	A	T	Z	.

Prof. Dr. rer. nat. Nane Kratzke | 16

## Einige Funktionen und Operatoren für Strings



University of Applied Sciences

Zeichenextraktion

Mit den folgenden Funktionen lassen sich  
Ersetzungen in Zeichenketten vornehmen.

Länge

```
$str = "Dies ist ein Satz";  
$newstr = strtolower($str);
```

Vergleichen

dies ist ein satz.

Suchen

```
$newstr = strtoupper($str);
```

Ersetzen

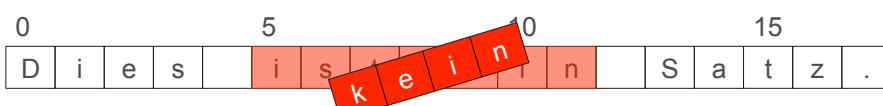
DIES IST EIN SATZ.

Zerlegen

```
$newstr = str_replace("ist ein",  
"kein", $str);
```

Verketten

Dies kein Satz.



Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

17

## Einige Funktionen und Operatoren für Strings



University of Applied Sciences

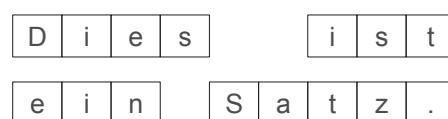
Zeichenextraktion

Mit der **explode** Funktion lassen sich  
Zeichenketten in Teilzeichenketten zerlegen.

Länge

```
$str = "Dies ist ein Satz.";  
$subs = explode(" ", $str);
```

Vergleichen



Suchen

Ersetzen

Zerlegen

Verketten



Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

18

## Einige Funktionen und Operatoren für Strings



University of Applied Sciences

Zeichenextraktion

Mit der **implode** Funktion und dem Konkatenationsoperator lassen sich Zeichenketten verknüpfen.

Länge

```
$str1 = "Hello";
```

Vergleichen

```
$str2 = " ";
```

Suchen

```
$str3 = "World!";
```

Ersetzen

```
echo $str1.$str2.$str3;
```

Zerlegen

Hello World!

Verketten

```
echo implode(" ",
```

```
array("Hello", "World!"));
```

Hello World!

0

5

10

15

D i e s i s t e i n S a t z .

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

19

## Die wesentlichen Datentypen in PHP



University of Applied Sciences

Bool  
Integer  
Float

Strings

Arrays

Type Casting

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

20

## Klassische Datenstrukturen hier: am Bsp. JAVA



University of Applied Sciences



Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

21

## Arrays

All diese fundamentalen Datenstrukturen werden in PHP durch einen einzigen Typ abgebildet.  
Array



University of Applied Sciences

Reihung von Key-Value Paaren beliebigen Typs



Voll-dynamisch

Größe zur Laufzeit festlegbar

Nach Erzeugung beliebig änderbar



Können mehrdimensional sein

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

22

## Erzeugung von Arrays und Zugriff auf Elemente des Arrays



University of Applied Sciences

Arrays werden mit dem array Operator erzeugt.

```
$a = array("Hello", " ", "World");
```

Auf die Elemente innerhalb des Arrays kann über deren Index zugegriffen werden, der in eckigen Klammern notiert wird:

```
echo $a[0];
```

Hello

```
echo $a[1];
```

Leerzeichen

```
echo $a[2];
```

World

An ein Array können weitere Werte angehängt werden:

```
$a[] = 42;
```

array("Hello", " ", "World", 42)

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

23

## Zugriff auf Array-Elemente



University of Applied Sciences

### Schreibender Zugriff:

```
$prim = array(); // Leeres Array erzeugen  
  
$prim[0] = 2; // Setzen von Array-Elementen  
$prim[1] = 3;  
$prim[2] = 5;  
$prim[3] = 7;  
$prim[4] = 11;
```

\$prim: 

2	3	5	7	11
---	---	---	---	----

### Typischer lesender Zugriff über einen Laufindex:

```
foreach ($prim as $p) { echo $p; }
```

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

24

## Arrays Key Value Paare (I)



University of Applied Sciences

```
$prim = array();  
  
$prim[0] = 2;  
$prim[1] = 3;  
$prim[2] = 5;  
$prim[3] = 7;  
$prim[4] = 11;
```

```
print_r($prim);
```

Ergibt folgenden Repräsentation:

```
Array  
(  
    [0] => 2  
    [1] => 3  
    [2] => 5  
    [3] => 7  
    [4] => 11  
)
```

Keys

Values

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

25

## Arrays Key Value Paare (II)



University of Applied Sciences

Schreiben von Key-Value Paaren:

```
$farbcodes['rot'] = "#FF0000";  
$farbcodes['grün'] = "#00FF00";  
$farbcodes['blau'] = "#0000FF";
```

```
Array  
(  
    [rot] => #FF0000  
    [grün] => #00FF00  
    [blau] => #0000FF  
)
```

Lesen von Key-Value Paaren:

```
echo $farbcodes['rot'];
```

#FF0000

```
echo $farbcodes['grün'];
```

#00FF00

```
echo $farbcodes['blau'];
```

#0000FF

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

26

## Mehrdimensionale Arrays (I)



University of Applied Sciences

Beispiel für ein einfaches mehrdimensionales Array.



Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

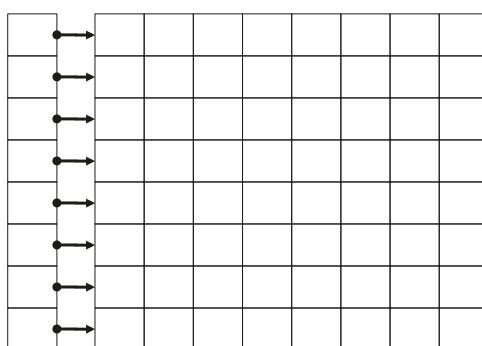
27

## Mehrdimensionale Arrays (II)



University of Applied Sciences

Mehrdimensionale Arrays werden als Arrays von Arrays angelegt.

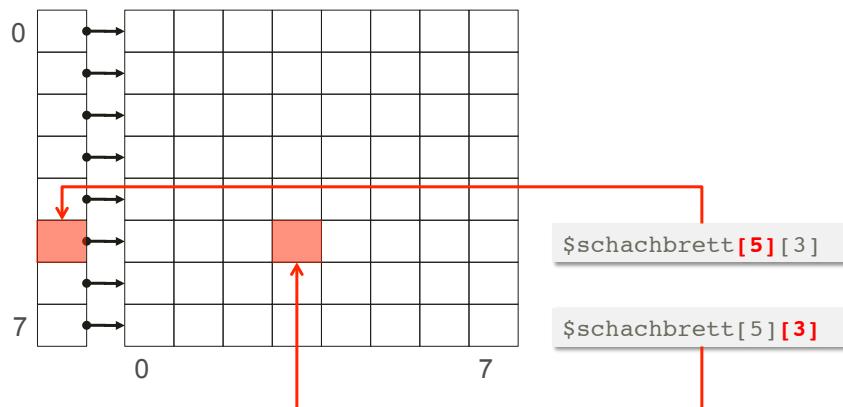


Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

28

## Mehrdimensionale Arrays (III)

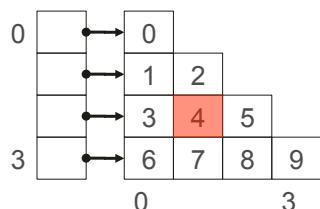
Mehrdimensionale Arrays werden als Arrays von Arrays angelegt.



## Mehrdimensionale Arrays (IV)

Es ist auch möglich nicht rechteckige Arrays anzulegen.

```
$array = array(array(0),  
              array(1, 2),  
              array(3, 4, 5),  
              array(6, 7, 8, 9)  
);
```



Welchen Wert hat  
dieser Ausdruck?

`$a[2][1] = ???`

## Spezielle Funktionen für Arrays (I)



University of Applied Sciences

Funktion	Wirkungsweise
<code>array_map</code>	Wendet eine Funktion auf alle Elemente eines Arrays an und liefert darüber ein Array zurück
<code>array_keys</code>	Liefert alle Schlüssel eines Arrays als Array
<code>array_values</code>	Liefert alle Werte eines Arrays als Array

Darüberhinaus gibt es viele weitere vergleichbare Funktionen zu Arrays in PHP, die Datenstruktur Array zu einer sehr mächtigen Metastruktur machen.

Viele Funktionalitäten in PHP Applikationen basieren auf der intensiven Nutzung, dieser vielseitigen und flexiblen Datenstruktur und Funktionen wie den oben gezeigten.

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

31

## Spezielle Funktionen für Arrays (II)



University of Applied Sciences

```
$array = array(1,2,3);
echo array_map(function($x) { return $x * $x; }, $array);
Array(1, 4, 9)
```

```
$farbcodes['rot'] = "#FF0000";
$farbcodes['grün'] = "#00FF00";
$farbcodes['blau'] = "#0000FF";
```

```
echo array_keys($farbcodes);      Array(rot, grün, blau)
echo array_values($farbcodes);    Array(#FF0000, #00FF00, #0000FF)
```

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

32

## Mini-Übung: Arrays



University of Applied Sciences

Gegeben ist folgendes Array

```
$array = array("a" => array("I" => 0,  
    "b" => array("II" => 1, "III" => 2),  
    "c" => array("IV" => 3, "V" => 4, "VI" => 5)  
);
```

Wie lauten die Ausgaben oder gesuchten Werte (?)

- |                         |                              |
|-------------------------|------------------------------|
| echo \$array["a"]["I"]; | 0                            |
| \$array["c"][?] == 4;   | V                            |
| \$array[?][?] == 2;     | \$array["b"]["III"]          |
| print_r(\$array["b"]);  | Array("II" => 1, "III" => 2) |

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

33

## Die wesentlichen Datentypen in PHP



University of Applied Sciences

Bool  
Integer  
Float

Strings

Arrays

Type Casting

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

34

## Type Casting (I)



University of Applied Sciences

- In PHP kann eine Variable innerhalb eines Skripts beliebig
  - Wert und
  - Typ wechseln
- Der Typ wird automatisch aus dem Kontext ermittelt.

```
$a = 5;           // Typ Integer
$a = 1.5;         // Nun Typ Float
$a = "Hello";    // Nun Typ String
$a = array();     // Nun Typ Array
$a = TRUE;        // Nun Typ Bool
```

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

35

## Type Casting (II)



University of Applied Sciences

- Dennoch kann es erforderlich sein, Variablen von einem Typ in einen anderen zu wandeln
- Dies wird als Type Casting bezeichnet
- Hierzu gibt es type cast Operatoren
  - Der Zieltyp wird dazu einfach vor einen Ausdruck geschrieben

### Beispiel:

```
$a = 5;           // Typ Integer
echo (float)$a;  // $a wird als float ausgegeben, d.h.
                  // als 5.0
```

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

36

## Type Casting (III)



University of Applied Sciences

Operator	Wirkungsweise
<code>(int), (integer)</code>	Wandelt den Wert eines Ausdrucks in den Typ Integer
<code>(bool), (boolean)</code>	Wandelt den Wert eines Ausdruck in den Typ Boolean
<code>(float), (double), (real)</code>	Wandelt den Wert eines Ausdrucks in den Typ Float
<code>(string)</code>	Wandelt den Wert eines Ausdruck in den Typ String
<code>(array)</code>	Wandelt den Wert eines Ausdrucks in den Typ Array
<code>(object)</code>	Wandelt einen Wert in ein Object.

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

37

## Zusammenfassung



University of Applied Sciences

- **Datentypen**
  - Primitive Datentypen (Bool, Integer, Float)
  - Strings
  - Arrays
- **Typecasting**

Prof. Dr. rer. nat. Nane Kratzke  
Praktische Informatik und betriebliche Informationssysteme

38