

INTERNET 2

(Skript im Aufbau, Inhalte können sich noch ändern)

Prof. Dr. Wolf-Fritz Riekert
Hochschule der Medien (HdM) Stuttgart
Stuttgart Media University

<mailto:riekert@hdm-stuttgart.de>

<http://www.hdm-stuttgart.de/~riekert>

Lernziele:

- Verständnis der Funktionsweise von Webapplikationen
- Fähigkeit zur Entwicklung einer einfachen datenbankbasierten Webapplikation

Inhalte:

- Aufbau von Websites (Wiederholung/Steilkurs)
- Relationale Datenbanken
 - ⇒ Datenbanksystem MySQL
 - ⇒ Datenbankadministration mit phpMyAdmin
- Einführung in die Skriptsprache PHP
- Entwicklung einer einfachen Webapplikation mit PHP und MySQL

Wesentliches Ziel dieser Lehrveranstaltung ist der Aufbau einer datenbankgestützten Website mit Hilfe von PHP und MySQL.

Hierfür sind folgende **V**orkenntnisse erforderlich:

- Allgemeine Funktionsweise des WWW
- Webseitengestaltung mit der Hypertext Markup Language (HTML)
- Übertragung von Dateien (insbesondere HTML-Dateien, Grafiken, PHP-Skripte) auf einen Webserver mit Hilfe von sicherem FTP (FTP über SSH).

Dies wird im Folgenden erklärt. Wer die Vorkenntnisse bereits besitzt, kann den folgenden Abschnitt überspringen.

DAS WORLD WIDE WEB (WWW)

Client: Internet-Browser (z.B. Mozilla Firefox, Google Chrome, Microsoft Internet Explorer)

Server: Webserver (z.B. Apache HTTP Server, Microsoft Internet Information Services)

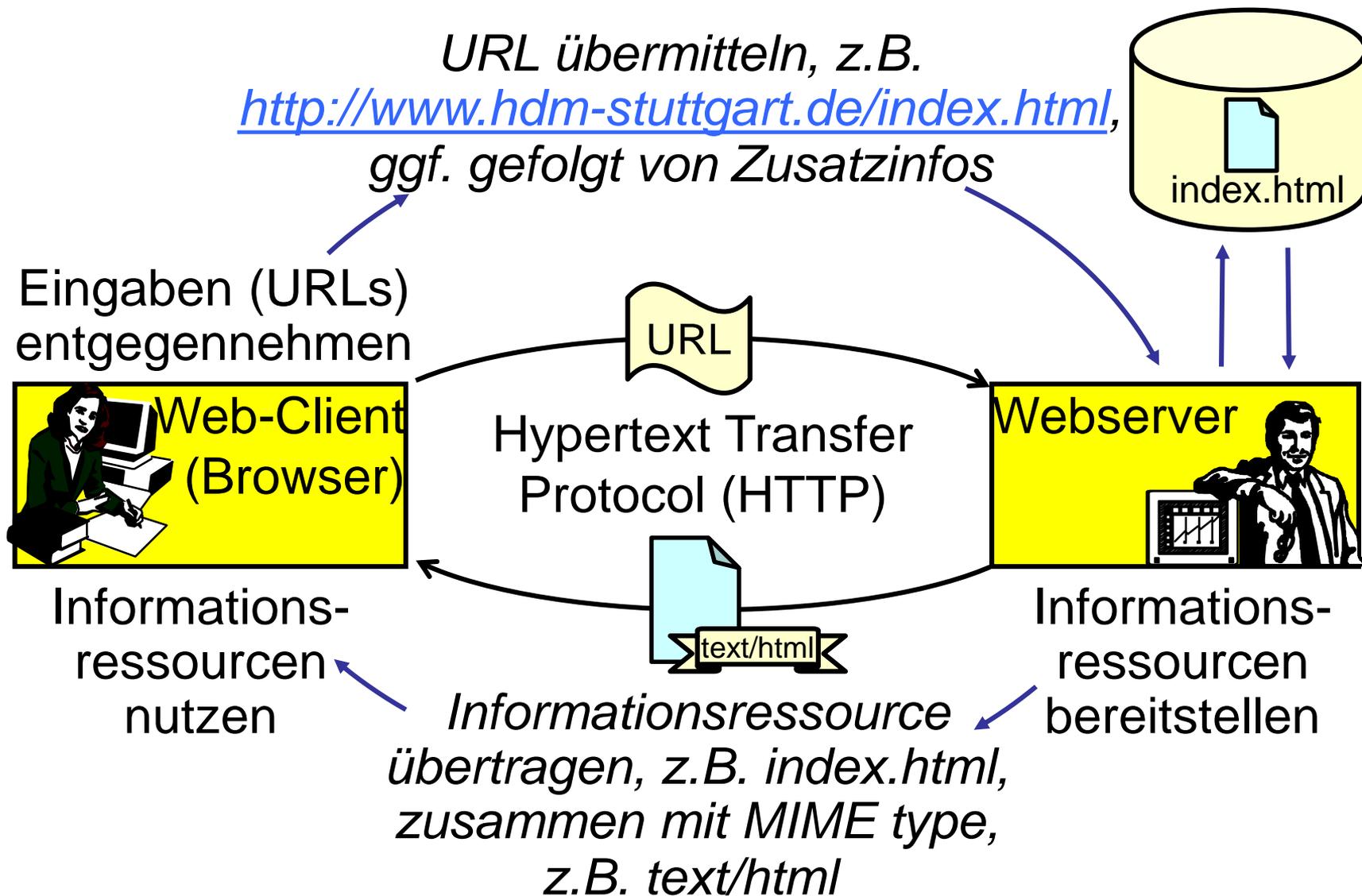
Dienst: Bereitstellen von Hypertextseiten und anderen Informationsressourcen (typisiert mit MIME Types) nach Angabe einer Adresse, der URL (Uniform Resource Locator)

Art des Dienstes: Verbindungsloser Anfrage/Antwort-Dienst

Protokolle: Hypertext Transfer Protokoll (HTTP), sichere Protokollvariante HTTPS über SSL (verschlüsselt, signiert)

Transportprotokoll: TCP (verbindungsorientiert!) über Port 80 (HTTP) bzw. Port 443 (HTTPS)

WEB-CLIENT (BROWSER) UND WEBSERVER



UNIFORM RESOURCE LOCATOR (URL)

URLs adressieren weltweit eindeutig Informationsressourcen (d.h. Daten, Dienstprogramme und multimediale Dokumente):

Aufbau: *Protokoll://Domain:Port/Pfad*

Beispiel: `http://dvmmail.zeppelein-nt.com:8080/lisa/index.html`

(Die Zeichen //, :, / sind syntaktische Kennzeichnungen für die verschiedenen Elemente der URL)

Protokoll: = Übertragungsprotokoll
(http: = Hypertext Transfer Protocol)

//Domain = Bezeichnung des Servercomputers im Internet

:Port = Kommunikationsport des Webserver-Programms,
i.d.R. nicht erforderlich, da Standardwert = 80

/Pfad = Ortsangabe im Dateisystem des Servers,
bestehend aus Verzeichnis(pfad) und Dateiname

URLs: VARIANTEN

Relative URLs: Hypertextseiten enthalten oft relative Links. Das Protokoll, die Domain und der Schrägstrich vor dem Verzeichnispfad werden dann weggelassen. Beispiele:

- english.html (d.h. die Seite liegt im gleichen Verzeichnis wie aktuelle Hypertextseite)
- ../cgi-bin/test.cgi (liegt im Nachbarverzeichnis cgi-bin)

Andere Protokolle: Außer http: sind noch andere Protokolle möglich: **https:** (verschlüsselte Datenübertragung im Web, z.B. für Internet Banking etc.), **ftp:** (Verwendung des klassischen File Transfer Protocols).

Wie ein Protokoll behandelt werden **mailto:** und **telnet:** (Aufruf des Mailsystems bzw. des Telnet-Clients für eine bestimmte Adresse, **file:** (lokaler Dateizugriff ohne Server).

- Hypertext Markup Language (HTML) = Dokumentenbeschreibungssprache des WWW
 - ⇒ Webseiten werden durch HTML-Dateien beschrieben
 - ⇒ Hypertext: Die Dokumente sind über Links verknüpft
 - ⇒ Markup: Die Bedeutung der Dokumentinhalte wird durch Markierungen mit der HTML-Sprache festgelegt.
- HTML legt primär die logische **Struktur** von Dokumenten fest: Überschriften, Absätze, Abbildungen, Tabellen, Links
- Mit Formatvorlagen in der Sprache CSS (Cascaded Style Sheets) wird die genaue Darstellung (das **Layout**) festgelegt
- Gestaltung von HTML-Seiten
 - ⇒ im HTML-Quelltext mit einem reinen Texteditor
 - ⇒ oder mit einem so genannten WYSIWYG-Editor (*What you see is what you get*)

EIN EINFACHES HTML-BEISPIEL

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Riekerts Homepage</title>
  </head>

  <body>
    <h1>Willkommen!</h1>
    
    <p>Wolf-Fritz <em>Riekert</em><br />
      <a href="http://www.hdm-stuttgart.de">
        HdM Stuttgart
      </a>
    </p>
  </body>
</html>
```



- HTML besteht aus Elementen
 - ⇒ Elemente sind markiert durch „Tags“ (sprich „Tägs“)
 - ⇒ Syntax: **Start-Tag** `<tagname>` *Inhalt* **Ende-Tag** `</tagname>`
 - ⇒ Anordnung nacheinander (z.B.: `......<i>...</i>`) oder verschachtelt (z.B.: `...<i>...</i>...`)
 - ⇒ Manche Tags haben keinen Ende-Tag, z.B. `
`, wird oft gekennzeichnet durch einen Schrägstrich: `
`
- Manche Tags haben Attribute
 - ⇒ Syntax: `<tagname attributname = "Wert">`
- Die Tags werden im Browser nicht angezeigt
 - ⇒ Sie „sagen“ dem Browser, was der Tag-Inhalt bedeutet
 - ⇒ z.B. dass es sich um eine Überschrift oder einen Hyperlink handelt

HTML-Dokumente bezeichnet man als **wohlgeformt**, wenn sie die Syntax von XML (einer Sprachfamilie HTML-artiger Sprachen) einhalten. Insbesondere muss Folgendes gelten:

- Start-Tags, die kein End-Tag haben, soll man mit einem schließenden Schrägstrich kennzeichnen (z.B. `
`)
- Konsistente Groß-/Kleinschreibung (nicht: ` ... `)
⇒ Empfehlung: Durchweg Kleinschreibung verwenden!
- Kein Überlappen von Tags (also nicht: `<i>...</i>`)
- Alle Attribute müssen einen Wert haben, der in Anführungszeichen (" " oder ' ') eingeschlossen ist.
Verboten: `<tag att=wert>...</tag>` und `<tag att>...</tag>`
Richtig: `<tag att="wert"> ... </tag>`

Wohlgeformtheit ist nicht zwingend gefordert, aber sinnvoll.

EINIGE BEISPIELHAFTE HTML-ELEMENTE

HTML-Element	Anfangskennung	Endekennung
HTML-Wurzel	<code><html></code>	<code></html></code>
Kopfteil	<code><head></code>	<code></head></code>
Metadaten	<code><meta ... /></code>	
Dokumenttitel	<code><title></code>	<code></title></code>
Dokumentrumpf	<code><body></code>	<code></body></code>
Überschrift Gr. 1	<code><h1></code>	<code></h1></code>
Überschrift Gr. 2	<code><h2></code>	<code></h2></code>
Absatz (paragraph)	<code><p></code>	<code></p></code>
Zeilenwechsel (break)	<code>
</code>	
Betont (emphasized)	<code></code>	<code></code>
Hyperlink	<code></code>	<code></code>
Graphik	<code></code>	

- Der Browser kann HTML-Dokumente nur darstellen, wenn sie korrektes HTML enthalten. Man nennt sie dann valide.
- Valide **HTML-Dokumente** erfüllen folgende Forderungen:
 - ⇒ Sie erfüllen die allgemeine HTML-Syntax (Notation von Tags, Attributen usw.), ggf. Wohlgeformtheit
 - ⇒ Sie enthalten nur definierte Elemente und Attribute.
- Überprüfung im „Validator“ (<http://validator.w3.org/>) oder im HTML-Editor (z.B. Dreamweaver)
 - ⇒ Es gibt verschiedene HTML-Versionen. Die aktuellste Version ist HTML 5; diese sollte bei der Validitätsprüfung eingestellt sein.



Zum Erstellen von Webseiten mit Hilfe von HTML benötigt man ein Handbuch, um HTML-Elemente nachzuschlagen.

Hier zwei Empfehlungen:

- SELFHTML e.V.: SELFHTML-Wiki. Ein Online-Handbuch zu HTML, CSS und Javascript in Form eines Wiki.
<http://wiki.selfhtml.org/>
- Stefan Münz und Clemens Gull (2013): HTML5-Handbuch. 9. Aufl. Haar bei München: Franzis Verlag GmbH. Online:
<http://webkompetenz.wikidot.com/docs:html-handbuch>

- Einen für HTML-Code geeigneten Editor starten
 - ⇒ eine **neue Webseite erstellen**
 - ⇒ Webseite im Dateisystem **speichern**
- **Vorschau** mit Web-Browser (Mozilla Firefox, Google Chrome, Safari, Internet-Explorer, ...), am besten mehrere Browser verwenden
- Seite mit sicherem **FTP** auf den Webserver kopieren
- **Ergebnis** auf Webserver mit Web-Browser **anschauen**

Zum Erstellen von Webseiten können verschiedene Arten von Editoren verwendet werden:

- einfache Text-Editoren wie [Editor](#) bzw. [Notepad](#) (in Windows integriert) oder [Textedit](#) (in MacOS integriert).
Speichern als reinen Text (plain text) mit Encoding UTF-8.
- Text-Editoren mit Syntaxunterstützung: [Brackets](#) (Windows und Mac), [Textwrangler](#) (Mac), [Notepad++](#) (Windows), [Phase 5](#) (Windows), alle zum freien Download
- Editoren mit WYSIWYG-Unterstützung („What you see is what you get“): Professionell und kostenpflichtig: [Adobe Dreamweaver](#) (in einigen Laboren installiert).
Freie Alternativen: [Microsoft Expression Web 4](#), [Microsoft Visual Studio Community](#), [Mozilla Seamonkey Composer](#).

WEBSEITE ERSTELLEN MIT EINEM REINEN TEXTEDITOR

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Wolf-Fritz Riekerts Homepage</title>
  </head>
  <body>
    <h1>Wolf-Fritz Riekert</h1>
    <p><em>Dies ist meine Homepage</em></p>
    <p></p>
    <p><a href="http://www.hdm-stuttgart.de">
    Hochschule der Medien Stuttgart</a></p>
    <p><a href="lehrveranstaltungen.html">
    Lehrveranstaltungen</a></p;
  </body>
</html>
```

Allgemeiner Rahmen, kann immer gleich bleiben

Erscheint als Titel von Browserfenster bzw. Tab

Texte eingeben und z.B. als Überschrift <h1>, Absatz <p> oder „betont“ („emphasized“) auszeichnen

Bild einbetten

relative URL

Hyperlink, absolute URL

Hyperlink, relative URL, d.h. Datei liegt in selbem Verzeichnis.

URL (UNIFORM RESOURCE LOCATOR) – WIEDERHOLUNG

URLs adressieren weltweit eindeutig Informationsressourcen (d.h. Daten, Dienstprogramme und multimediale Dokumente):

Aufbau: *Protokoll://Domain:Port/Pfad*

Beispiel: `http://dvmmail.zeppelin-nt.com:8080/lisa/index.html`

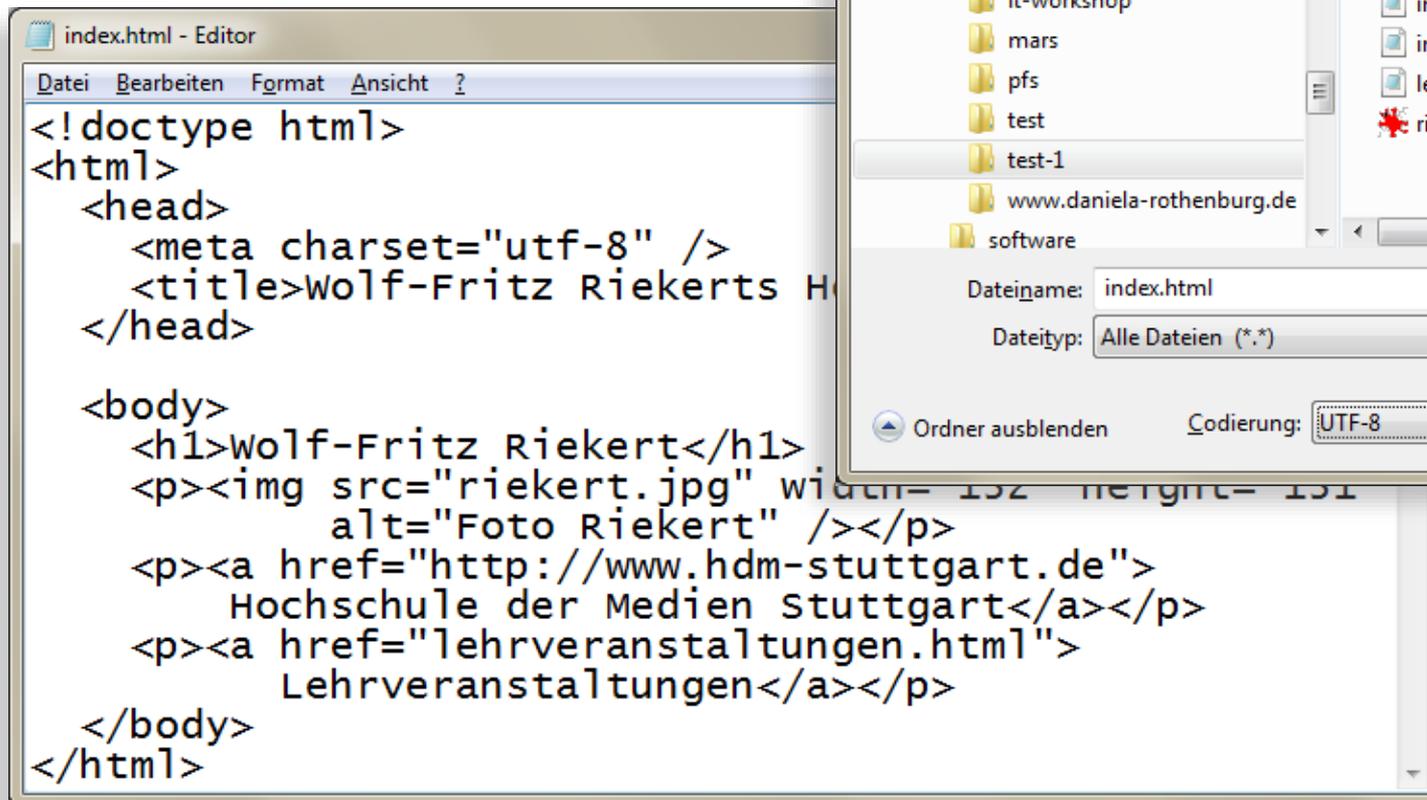
Relative URLs: Das Protokoll, die Domain und der Schrägstrich vor dem Verzeichnispfad werden weggelassen. Beispiele:

- english.html (d.h. die Seite liegt im gleichen Verzeichnis wie aktuelle Hypertextseite)
- ../cgi-bin/test.cgi (liegt im Nachbarverzeichnis cgi-bin)

Andere Protokolle: Außer http: sind noch andere Protokolle möglich: **https:** (verschlüsselte und signierte Datenübertragung im Web, immer häufiger auch auf normalen Websites z.B. HdM), **ftp:** (Verwendung des klassischen File Transfer Protocols).

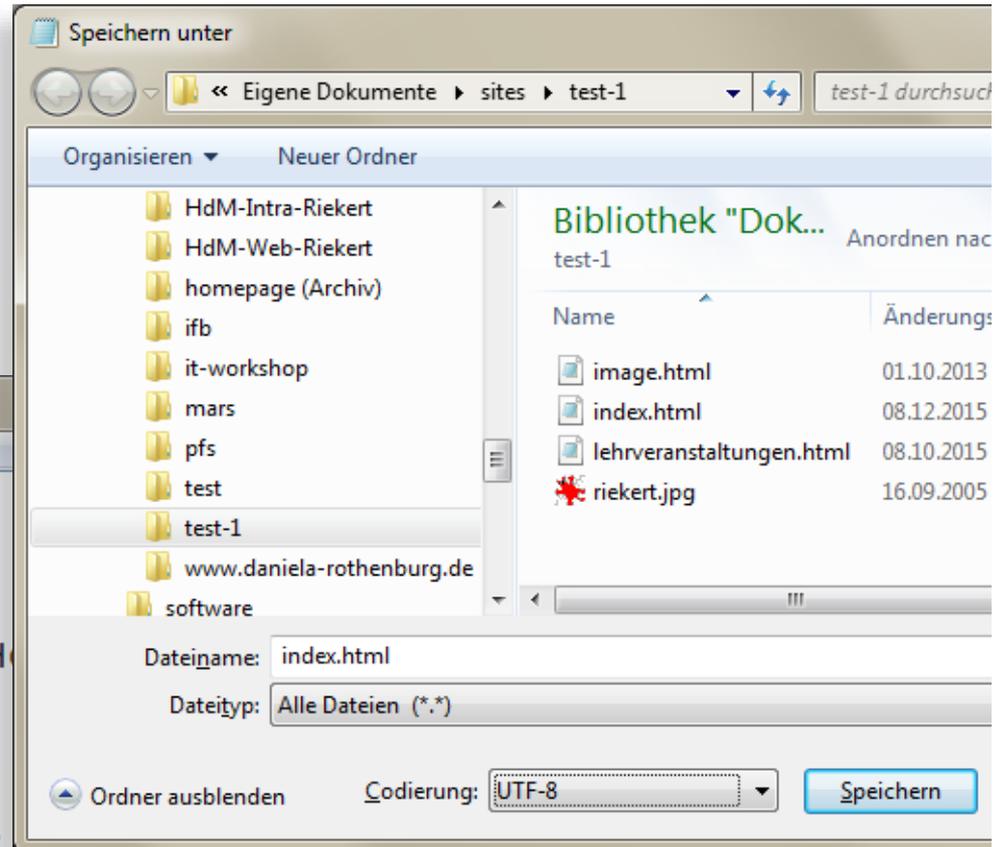
EDITIEREN VON HTML-CODE MIT DEM „EDITOR“ UNTER WINDOWS

Editor: Aufruf über
Startmenü /
Alle Programme /
Zubehör / Editor



```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Wolf-Fritz Riekerts H
  </head>

  <body>
    <h1>wolf-Fritz Riekert</h1>
    <p></p>
    <p><a href="http://www.hdm-stuttgart.de">
      Hochschule der Medien Stuttgart</a></p>
    <p><a href="lehrveranstaltungen.html">
      Lehrveranstaltungen</a></p>
  </body>
</html>
```



Speichern
mit
Codierung
UTF-8

TEXTEDIT: HTML-DOKUMENTE NEU ERSTELLEN AUF DEM MAC

The image shows the TextEdit application interface on a Mac. The top menu bar includes 'TextEdit', 'Ablage', 'Bearbeiten', 'Format', 'Darstellung', 'Fenster', and 'Hilfe'. The 'Einstellungen' (Settings) window is open, showing the 'Format' section. The 'Reiner Text' (Plain Text) option is selected, indicated by a red circle. A yellow callout bubble points to this option with the text: 'Einstellungen der Anwendung Textedit: Format = Reiner Text'. Below the settings, a 'Neues Dokument erstellen und abspeichern als UTF-8' callout bubble points to the 'Neues Dokument' button. In the foreground, a new document window titled 'Ohne Titel — Bearbeitet' is open, displaying HTML code. A 'Sichern unter' (Save As) dialog box is also visible, showing the filename 'lehrveranstaltungen.html', the location 'Schreibtisch', and the encoding 'Unicode (UTF-8)'. The dialog has 'Abbrechen' and 'Sichern' buttons.

TextEdit Ablage Bearbeiten Format Darstellung Fenster Hilfe

Einstellungen

Neues Dokument Öffnen und Sichern

Format
Verwenden Sie das Menü „Format“, um Einstellungen für ein geöffnetes Dokument festzulegen.

Formatierter Text Seitenränder einblenden
 Reiner Text

Neues Dokument erstellen
und abspeichern als UTF-8

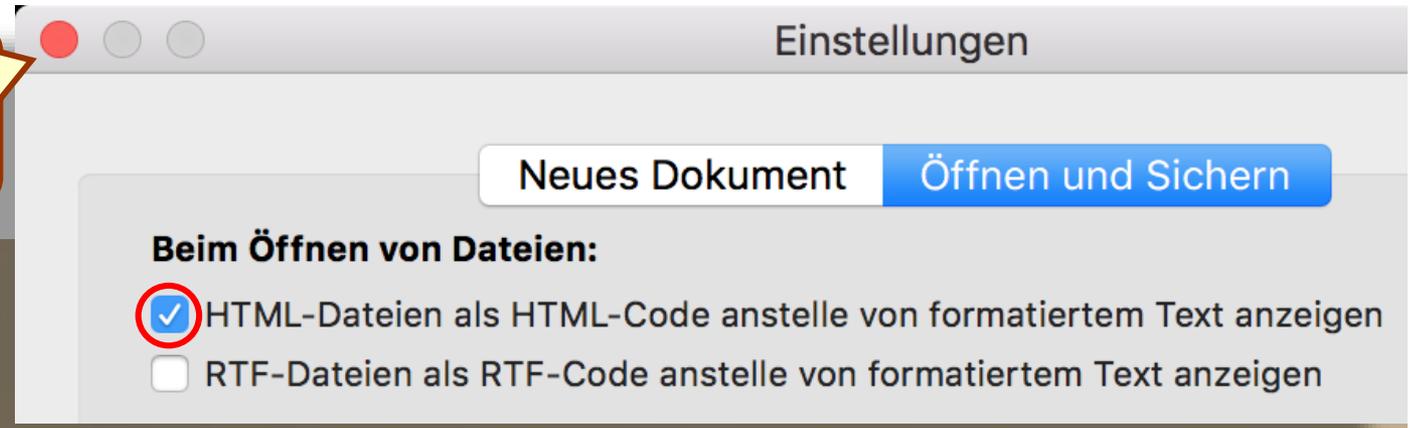
Sichern unter: lehrveranstaltungen.html
Tags:
Ort: Schreibtisch
Codierung für reinen Text: Unicode (UTF-8)
 .txt verwenden, falls kein Suffix angegeben ist

Abbrechen Sichern

```
<!doctype html>  
<html>  
  <head>  
    <meta charset="utf-8" />  
    <title>Lehrveranstaltungen</title>  
  </head>  
  
  <body>  
    <h1>Lehrveranstaltungen</h1>  
    <p>Internet 1</p>  
    <p>Internet 2</p>  
  </body>  
</html>
```

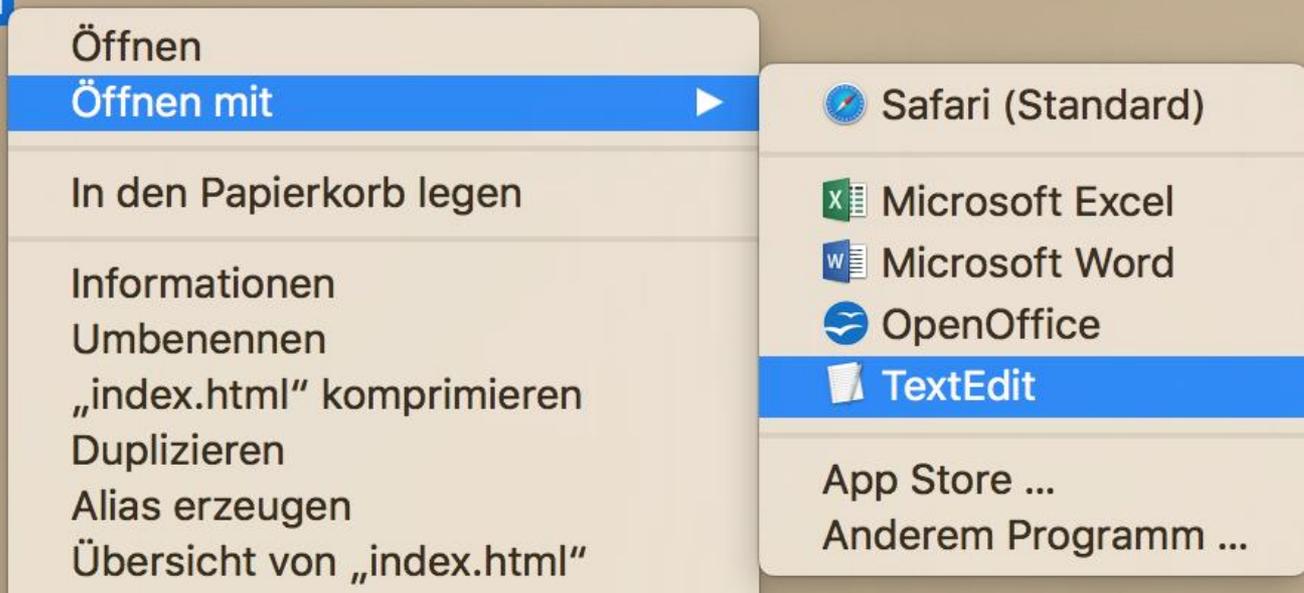
TEXTEDIT: HTML-DOKUMENTE BEARBEITEN AUF DEM MAC

Einstellungen
der Anwendung
Textedit

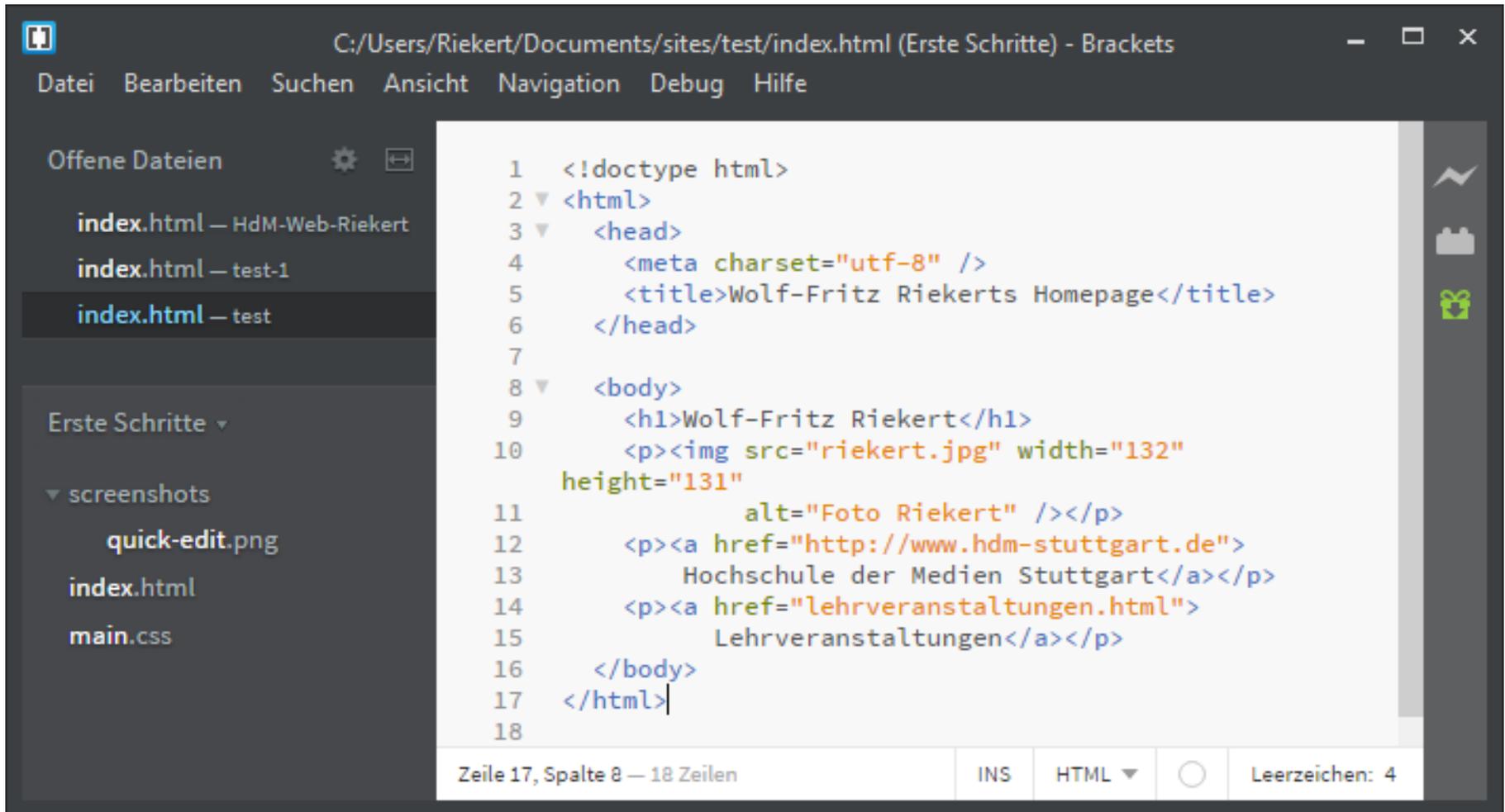


index.html

„rechter“
Mausklick
mit zwei
Fingern



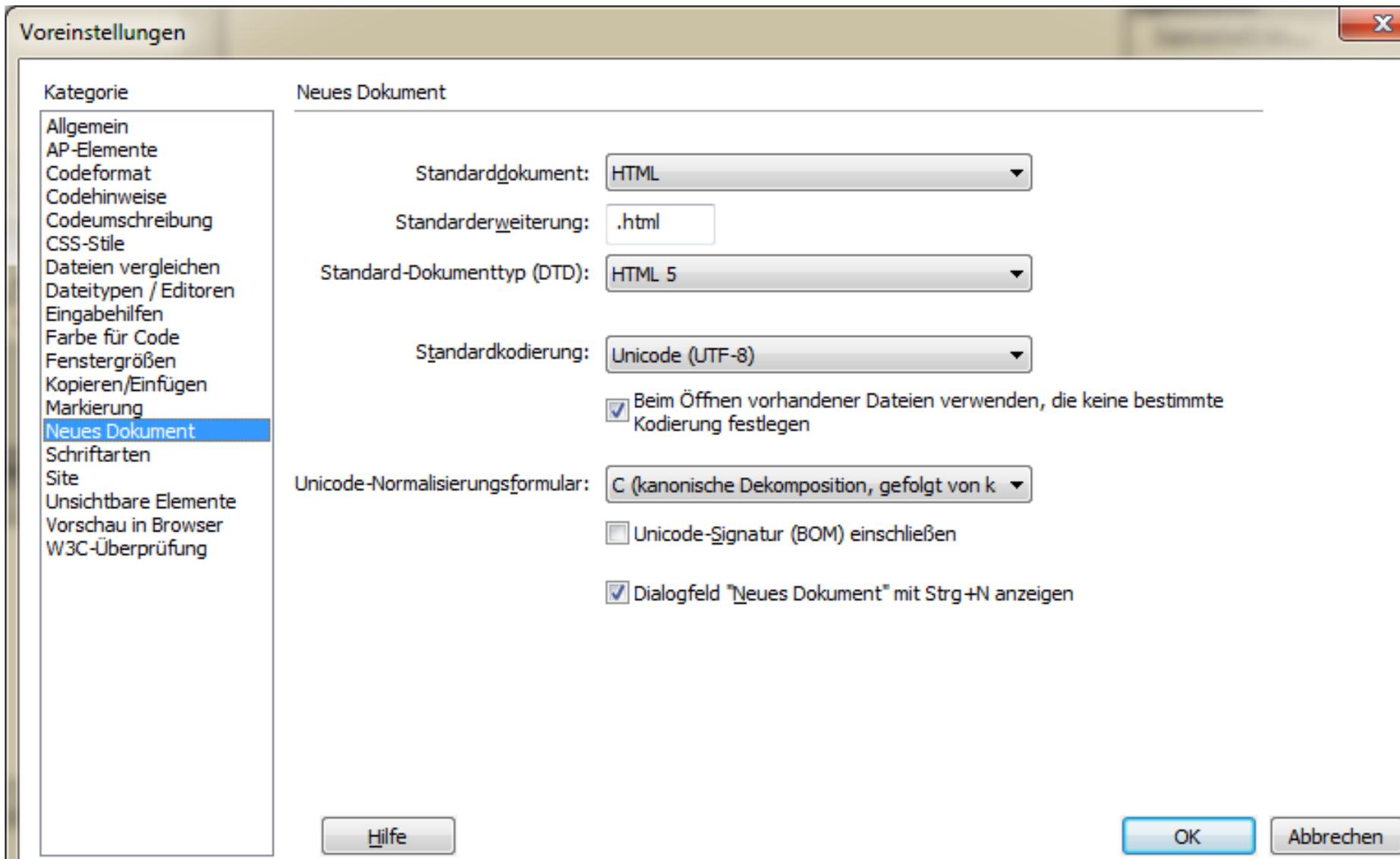
BRACKETS: EIN HTML-EDITOR MIT SYNTAXUNTERSTÜTZUNG



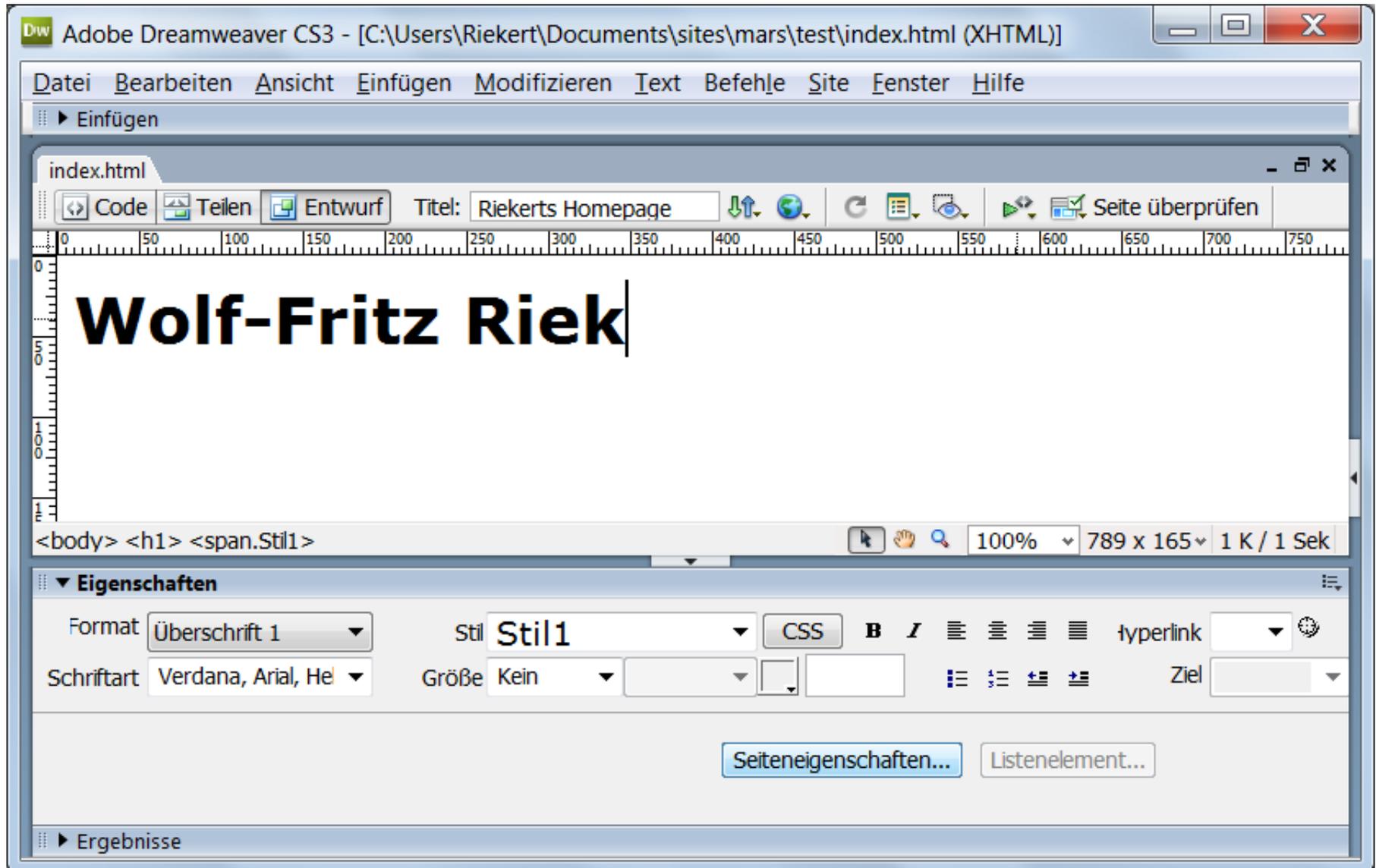
Brackets: ein HTML-Editor mit Syntaxunterstützung.
Freier Download für Windows und Mac unter <http://brackets.io>

- **Voreinstellungen** vornehmen: HTML 5, Unicode (UTF-8), wenn möglich.
- Eigentliche Erstellung der Webseite ähnlich wie in Word:
 - ⇒ **Texte** nach Belieben schreiben. Formate ändern usw.
 - ⇒ **Bilder** (JPG, GIF, PNG) einfügen (eigentlich nur verknüpfen)
 - Am besten aus selbem Verzeichnis wie HTML-Datei
 - ⇒ **Hyperlinks** einfügen
 - Textbereich oder Grafik mit der Maus markieren
 - mit Schaltfläche **Verknüpfung** URL festlegen
- Sichern mit Menü **Datei - Speichern unter**
 - ⇒ Als Namen für die Startseite verwendet man **index.html**
 - ⇒ Weitere Webseiten erhalten beliebige Namen mit Endung **.html**

DREAMWEAVER: VOREINSTELLUNGEN



DREAMWEAVER: NEUE WEBSEITE ERSTELLEN



DREAMWEAVER: ENTWURFSANSICHT

The screenshot shows the Adobe Dreamweaver CS3 interface in Design View. The main workspace displays a webpage design for 'index.html' with the title 'Riekerts Homepage'. The design features a large heading 'Wolf-Fritz Riekert' and a portrait of a man. Below the portrait is the text 'Dies ist meine Homepage'. A tooltip 'Entwurfsansicht anzeigen' is visible over the design area. The status bar at the bottom shows the code view with '<body>' and '<h1>'. The right-hand panels include the 'Dateien' (Files) panel showing a file list for the 'mars' site, and the 'Codefragmente' (Code Snippets) panel. The status bar at the bottom right indicates '1 lokale Elemente mit insgesamt Protokoll...'. The top menu bar includes 'Datei', 'Bearbeiten', 'Ansicht', 'Einfügen', 'Modifizieren', 'Text', 'Befehle', 'Site', 'Fenster', and 'Hilfe'. The 'Einfügen' (Insert) menu is open, showing options like 'Allgemein', 'Layout/Form', 'Daten', 'Spry', 'Text', and 'Favorit'.

Lokale Dateien	Gr...	Typ	Geär
riekert.jpg	14KB	IrfanVi...	07.10
quadratzahl...	1KB	PHP S...	22.11
quadratzahl...	1KB	Firefox...	04.11
mysql-zuga...	1KB	Textd...	03.04
mailtest.php	1KB	PHP S...	04.12
lehrveranst...	1KB	Firefox...	07.10
index.php	2KB	PHP S...	22.11
index.html	1KB	Firefox...	05.10
homepage....	1KB	Firefox...	20.10
hello.php	1KB	PHP S...	22.11
donald.jpg	27KB	IrfanVi...	20.10
divform.html	1KB	Firefox...	19.10

DREAMWEAVER: CODEANSICHT

Adobe Dreamweaver CS3 - [C:\Users\Riekert\Documents\sites\mars\index.html (XHTML)]

Datei Bearbeiten Ansicht Einfügen Modifizieren Text Befehle Site Fenster Hilfe

Einfügen Allgemein Layout Form Daten Spry Text Favorit

index.html Code Teilen Entwurf Titel: Riekerts Homepage

```
7 Codeansicht anzeigen
8 <body>
9 <h1>Wolf-Fritz Riekert </h1>
10 <p></p>
11 >
12 <p>Dies ist meine <em>Homepage</em>
    </p>
13 <p><a href=
    "http://www.hdm-stuttgart.de">HdM
    Stuttgart</a></p>
14 </body>
15 </html>
```

Dateien Dateien Elemente Codefragmente

mars Lokale Ansicht

Lokale Dateien	Gr...	Typ	Geär
riekert.jpg	14KB	IrfanVi...	07.10
quadratzahl...	1KB	PHP S...	22.11
quadratzahl...	1KB	Firefox...	04.11
mysql-zuga...	1KB	Textd...	03.04
mailtest.php	1KB	PHP S...	04.12
lehrveranst...	1KB	Firefox...	07.10
index.php	2KB	PHP S...	22.11
index.html	1KB	Firefox...	05.10
homepage...	1KB	Firefox...	20.10
hello.php	1KB	PHP S...	22.11
donald.jpg	27KB	IrfanVi...	20.10
divform.html	1KB	Firefox...	19.10

1 lokale Elemente mit insgesamt Protokoll...

DREAMWEAVER: GETEILTE ANSICHT

Adobe Dreamweaver CS3 - [C:\Users\Riekert\Documents\sites\mars\index.html (XHTML)]

File Edit View Insert Modify Text Commands Site Window Help

▼ Einfügen Allgemein Layout/Formul Daten Spry Text Favorit

index.html Code Teilen Entwurf Titel: Riekerts Homepage

Code- und Entwurfsansicht anzeigen

```
7  
8 <body>  
9 <h1>Wolf-Fritz Riekert</h1>  
10 <p></p>
```

0 50 100 150 200 250 300 350 4

Wolf-Fritz Riekert



<body> <h1>

100% 407 x 147 15

1 lokale Elemente mit insgesamt Protokoll...

▼ Dateien

Dateien Elemente Codefragmente

mars Lokale Ansicht

Lokale Dateien	Gr...	Typ	Geär
riekert.jpg	14KB	IrfanVi...	07.10
quadratzahl...	1KB	PHP S...	22.11
quadratzahl...	1KB	Firefox...	04.11
mysql-zuga...	1KB	Textd...	03.04
mailtest.php	1KB	PHP S...	04.12
lehrveranst...	1KB	Firefox...	07.10
index.php	2KB	PHP S...	22.11
index.html	1KB	Firefox...	05.10
homepage....	1KB	Firefox...	20.10
hello.php	1KB	PHP S...	22.11
donald.jpg	27KB	IrfanVi...	20.10
divform.html	1KB	Firefox...	19.10

ALTERNATIVE ZU DREAMWEAVER: MICROSOFT EXPRESSION WEB

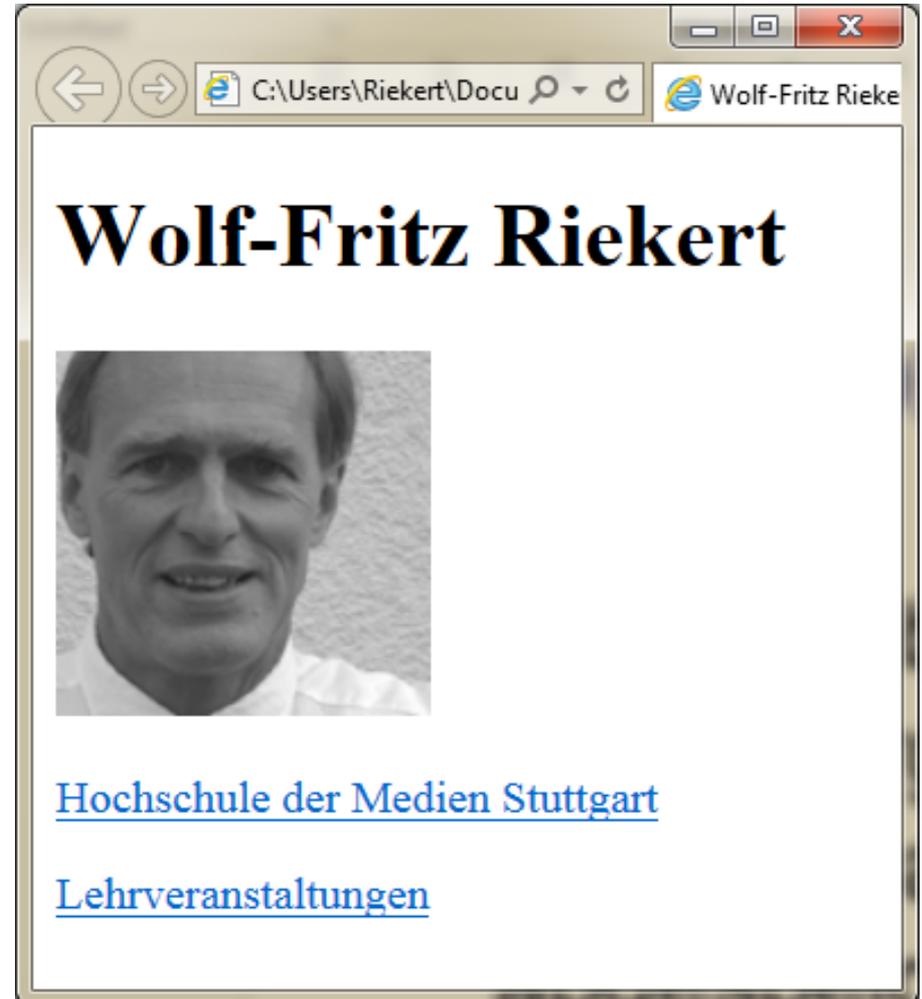
The screenshot shows the Microsoft Expression Web 4 interface. The title bar reads "Riekerts Homepage (Z:\Users\Riekert\Documents\sites\mars\index.html) - Microsoft Expression Web 4". The menu bar includes "Datei", "Bearbeiten", "Ansicht", "Einfügen", "Format", "Extras", "Tabelle", "Site", "Datenansicht", "Bereiche", "Fenster", and "Hilfe". The toolbar contains icons for file operations and text formatting (bold, italic, underline). The left pane shows a file explorer with the path "C:\Dokumente und Einstellungen\riekert\Eig...". The main workspace is split into a code view (top) and a WYSIWYG view (bottom). The code view shows the following HTML code:

```
<body> <h1>  
9 <h1>Wolf-Fritz Riekert </h1>  
10 <p>Dies ist meine <em>Homepage</em></p>
```

The WYSIWYG view shows a preview of the page with the heading "Wolf-Fritz Riekert" and a photo of a man. Below the photo is the text "Dies ist meine Homepage". The right pane contains a "Toolbox" with HTML tags and a "Formatvor..." panel with options for "Neue Formatvorlage...", "Optionsen", and "Stylesheet anfügen...". The bottom status bar shows "XHTML 1.0 T 14,1 KB" and "CSS 2.1 365 x 283".

Freier Download unter:
<http://www.microsoft.com/en-us/download/details.aspx?id=36179>

- Das Ergebnis des Editiervorgangs wird im lokalen Dateisystem abgespeichert
- Dort kann es mit einem Browser angezeigt werden.
- Damit es aber im WWW sichtbar wird, muss es auf einen Webserver übertragen werden.

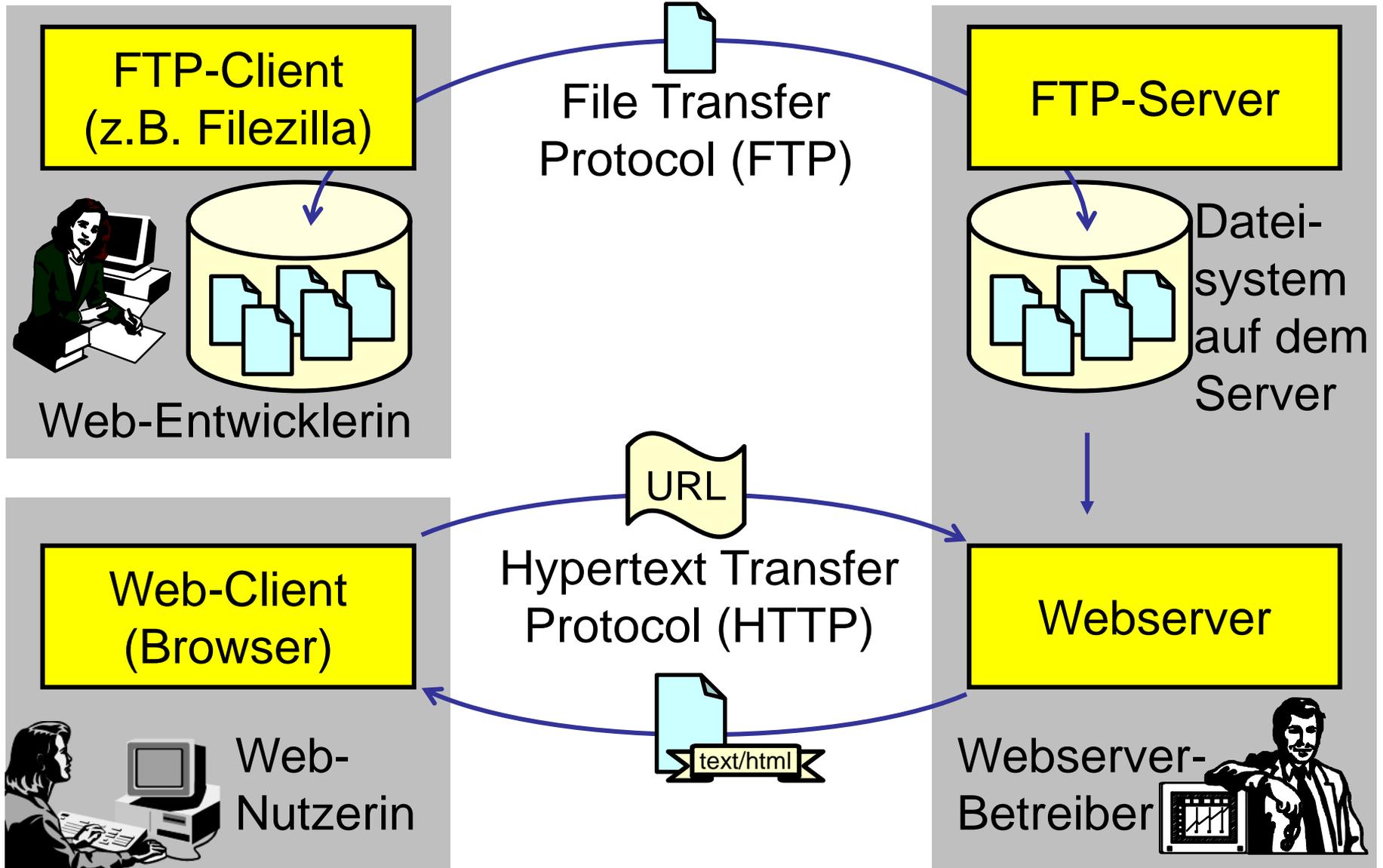


Frage: Wie kommen die Webseiten auf den Webserver?

Antwort: Mit **FTP** (File Transfer Protocol): einem der ältesten Verfahren zur Übertragung von Dateien zwischen Computern im Internet

- Webserver bieten i.d.R. einen FTP-Dienst an.
- Nutzung mit verschiedenen **FTP-Clients**:
 - ⇒ Windows-basierter FTP-Client **Filezilla** (hier erklärt)
 - ⇒ Eingebaute FTP-Funktion in HTML-Editoren, z.B. Dreamweaver oder Expression Web
- Problem: Der normale FTP-Dienst verschlüsselt Passwörter und Daten nicht
- Abhilfe: Neuere verschlüsselte FTP-Dienste verwenden: sicheres SFTP über SSH (alternativ: FTP über TLS)

FTP: DIENST ZUR ÜBERTRAGUNG VON WEBSEITEN AUF DEN SERVER



FTP (File Transfer Protocol): eines der ältesten Verfahren zur Übertragung von Dateien zwischen Computern im Internet

- Webserver bieten i.d.R. einen FTP-Dienst an.
- Nutzung mit verschiedenen **FTP-Clients**:
 - ⇒ Windows-basierter FTP-Client Filezilla (hier erklärt)
 - ⇒ Eingebaute FTP-Funktion in HTML-Editoren, z.B. Dreamweaver oder Expression Web
- Problem: Der normale FTP-Dienst verschlüsselt Passwörter und Daten nicht
- Abhilfe: Neuere verschlüsselte FTP-Dienste verwenden:
Der Übungsserver mars.iuk.hdm-stuttgart.de verwendet sicheres SFTP über SSH an.

FILEZILLA: WINDOWS-BASIERTER FTP-CLIENT

FileZilla

Datei Bearbeiten Ansicht Transfer Server Lesezeichen Hilfe

Server: Username: Passwort: Port: Verbinden

Lokal: C:\Users\Riekert\Documents\sites\mars\public_html

Server:

Dateiname	Dateigrö...	Dateit...
..		
index.php	1.293	PHP S...
time.php	384	PHP S...
final.zip	27.898	zip Ar...

1 Datei ausgewählt. Gesamtgröße: 1.293 Bytes

Derzeit mit keinem Server verbunden

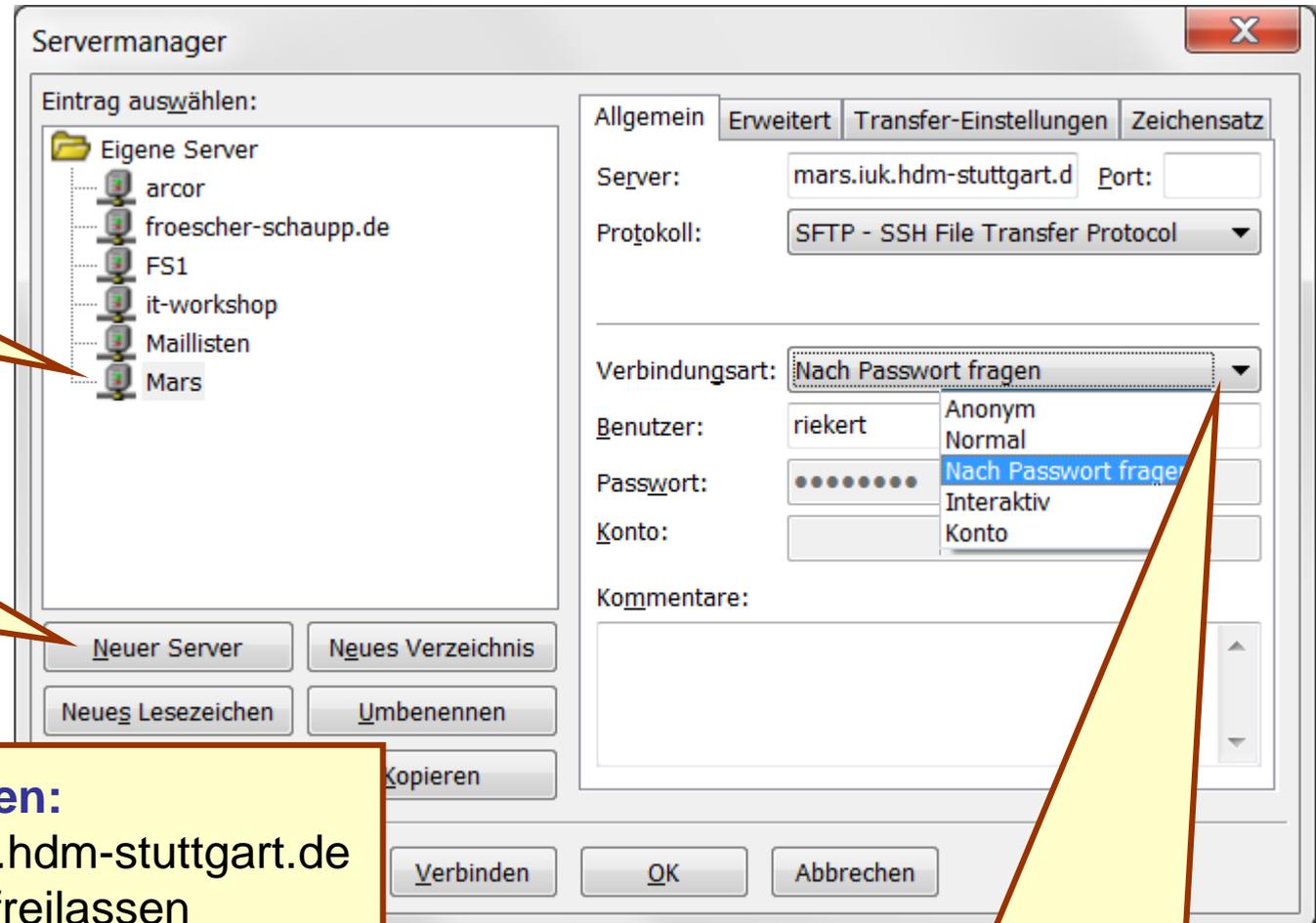
Nicht verbunden.

Server/Lokale Datei Richt... Datei auf Server

Zu übertragende Dateien Fehlgeschlagene Übertragungen Erf...

Download über <http://sourceforge.net/projects/filezilla>

FILEZILLA: SERVERMANAGER



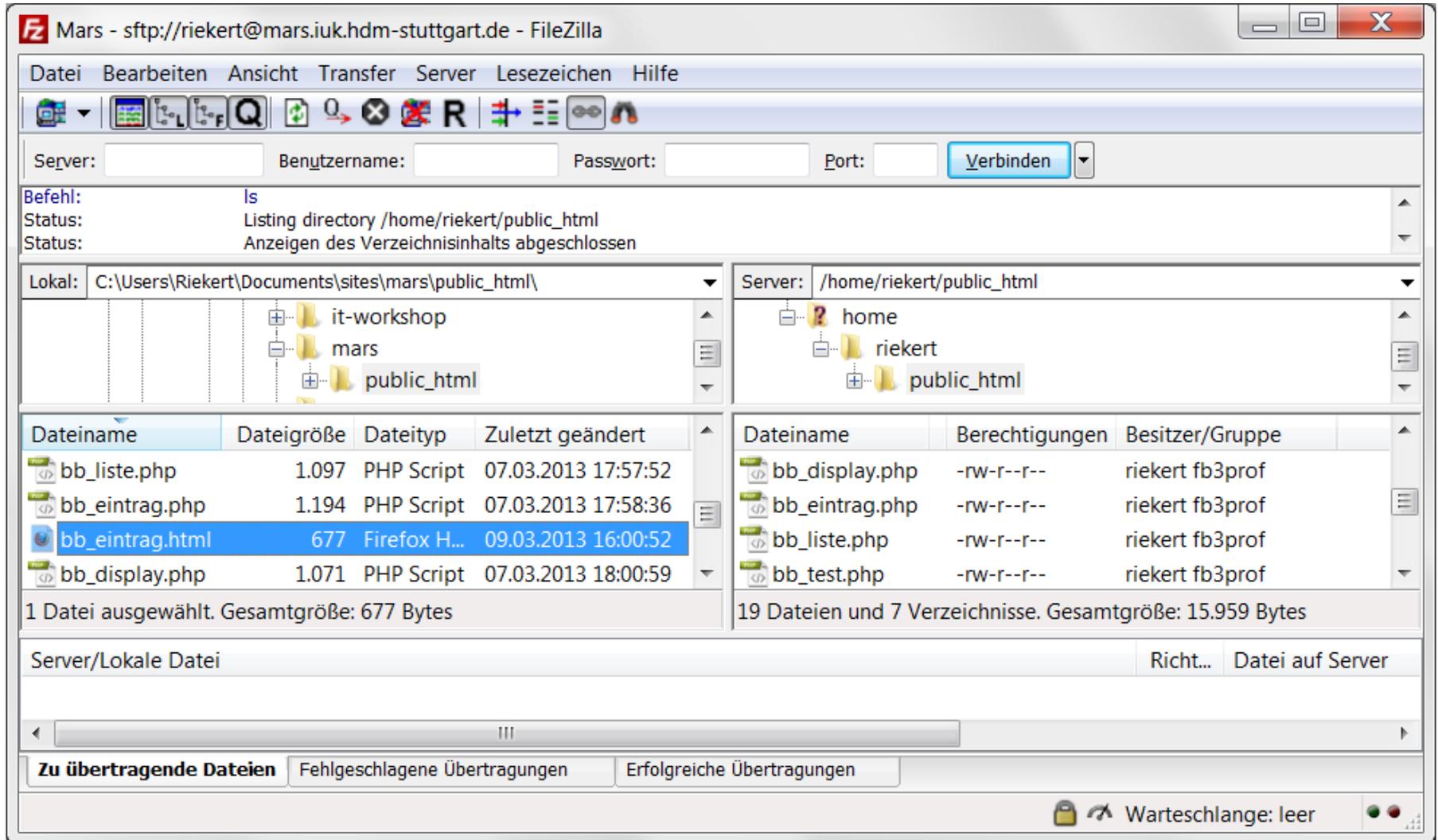
Hier Server
selektieren

Beim ersten Mal
neuen Server
Mars anlegen

Server-Einstellungen:
Host: mars.iuk.hdm-stuttgart.de
Port: 22 oder freilassen
Protokoll: SFTP über SSH
Benutzer: HdM-Namenskürzel
Passwort: wie für E-Mail

Verbindungsart:
Empfohlen: **Nach Passwort fragen!**
Auf eigenem Rechner auch: Normal

FILEZILLA: FILETRANSFER



Dateien können durch Ziehen auf den Server kopiert werden.

SPEICHERORT VON WEBDATEIEN AUF DEM WEBSERVER

Die Webdateien (HTML-Seiten, Grafiken, PHP-Skripte) müssen auf ein Verzeichnis des Webserver kopiert werden, das für das Web freigegeben ist.

- Auf Linux/Unix-Systemen mit dem Apache-Webserver ist dafür i.d.R. das Unterverzeichnis `public_html` vorgesehen.
- Wenn `public_html` noch nicht vorhanden ist, kann es mit Filezilla erzeugt werden (rechter Mausklick – Verzeichnis erstellen).
- Dateien im Unterverzeichnis `public_html` können vom Web aus mit der URL <http://rechnernamen/~username/dateiname> angesprochen werden. Beispiele:
 - ⇒ <http://mars.iuk.hdm-stuttgart.de/~xy999/datei.html>
 - ⇒ <http://mars.iuk.hdm-stuttgart.de/~xy999/> erreicht die Datei `index.html` oder `index.php` in `public_html`.

ERGEBNIS AUF DEM WEBSERVER MIT DEM BROWSER BETRACHTEN



Wolf-Fritz Riekert



[Hochschule der Medien Stuttgart](#)

Wenn man den Dateinamen weglässt, wird die Datei index.html oder index.php angezeigt (gilt für die meisten Unix/Linux-Server, auf Microsoft-Servern ist dies i.d.R. die Datei default.htm)

- Relationales Datenbanksystem
- Frei erhältlich (www.mysql.com)
- Läuft als Serverprozess (unter Unix/Linux und Windows)
- Keine graphische Benutzungsoberfläche wie z.B. MS Access
- Benutzung mittels Datenbanksprache SQL
- Gut funktionierende Schnittstelle zu PHP und anderen Programmiersprachen
- Fensterorientierte Bedienung über webbasiertes Administrationstool PHPMYADMIN

A. Kelz: *Relationale Datenbanken*, 1998.

<https://www.hdm-stuttgart.de/~riekert/lehre/db-kelz/index.htm>

[Gut verständliche Einführung in den Aufbau von relationalen Datenbanken und die Datenbanksprache SQL]

[SQL - Grundlagen und Datenbankdesign](#). Hannover : Leibniz Universität IT Services. [Gut geeignet als begleitende Lektüre. Auf das relationale Datenbanksystem MySQL wird besonders eingegangen. Erhältlich zum Preis von ca. 5,50 für HdM-Angehörige bei der [Benutzerberatung des Rechenzentrums der Universität Stuttgart](#)]

MySQL Documentation. <http://dev.mysql.com/doc/>

TABELLEN (RELATIONEN)

- Relationale DBMS speichern Daten in **Tabellen** (auch **Relationen** genannt, engl.: *table, relation*)
- Tabellen: intuitiv verständliche, einfach interpretierbare Art der Repräsentation von Informationen
- Die **Zeilen** der Tabelle (auch **Datensätze** genannt) repräsentieren untereinander gleichartige Informationseinheiten
- Datensätze sind gegliedert in **Felder** (auch **Merkmale** oder **Attribute** genannt).
- Die **Spalten** der Tabelle enthalten gleichartige Felder der Datensätze. Sie sind mit den Namen der Felder überschrieben
- In den Spalten stehen **Datenwerte** von gleichartigem **Datentyp** mit vordefiniertem **Wertebereich** (engl. *domain*)

BEISPIELTABELLE MITARBEITER

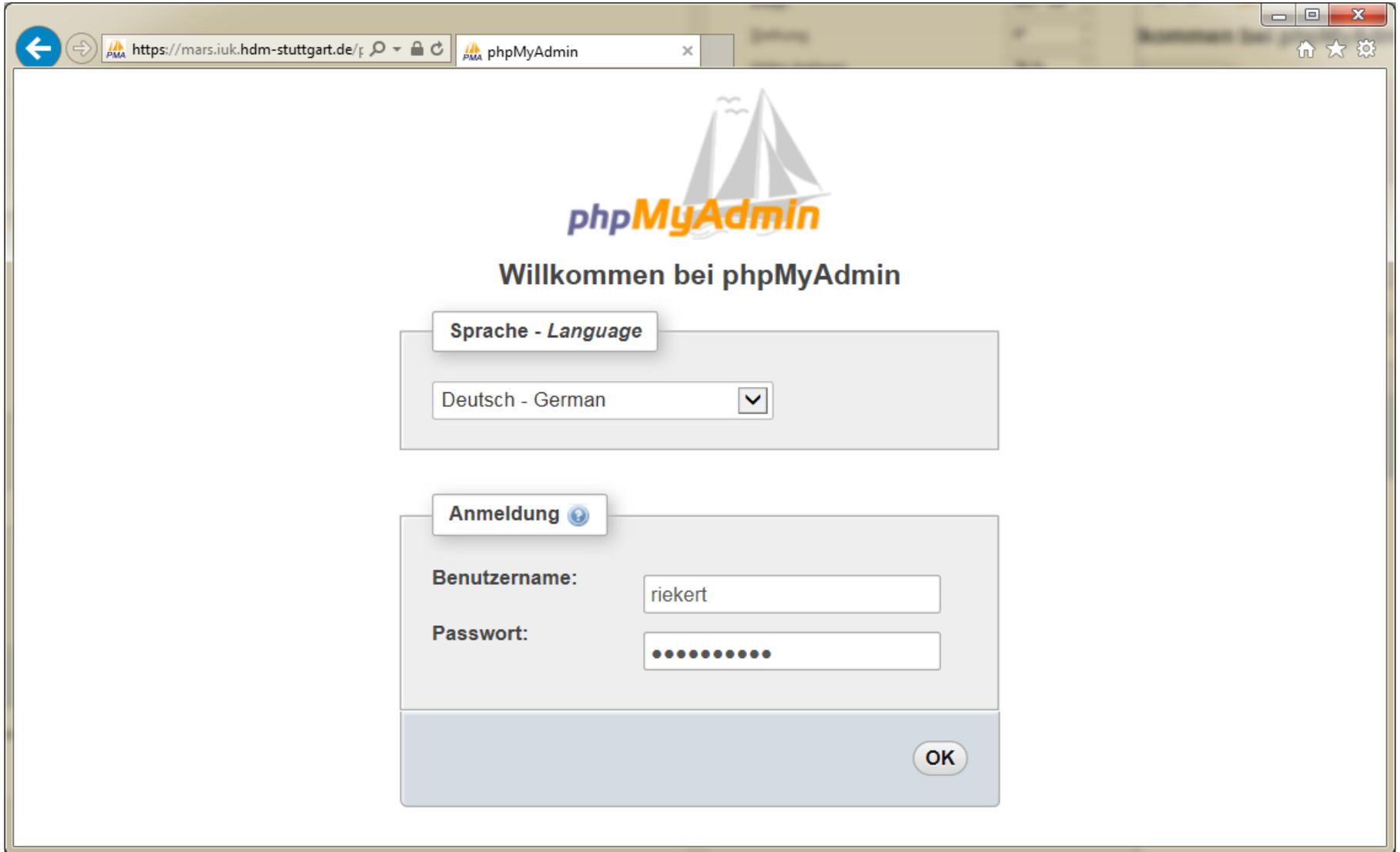
m_id	nachname	strasse	ort	a_id
1	Maier	Bahnhofstr. 3	Neu-Ulm	3
2	Huber	Türlenstr. 32	Stuttgart	2
3	Müller	Pragstr. 10	Stuttgart	0
4	Schreiber	Schlossstr. 20	Böblingen	3
5	König	Schillerstr. 29	Ludwigsburg	2
6	Kerner	Kelterstr. 51	Esslingen	3
7	Einstein	Planckstr. 6	Karlsruhe	2
8	Walz	Industriestr. 4	Mannheim	1
9	Braun	Waldstr. 4	Leonberg	2
10	Filzer	Kohlstr. 45	Filderstadt	3
11	Graf	Goethestr. 9	Sindelfingen	3
12	Kaiser	Pfarrstr. 40	Stuttgart	1
13	Hacker	Zusestr. 200	München	4

Auf dem Server Mars ist ein MySQL-Datenbanksystem eingerichtet.

- Jedes Hochschulmitglied hat eine Benutzungsberechtigung
- Die MySQL-Zugangsdaten stehen in einer Datei im Homeverzeichnis auf dem Server (ggf. Filezilla Anzeige aktualisieren)
 - ⇒ Benutzername
 - ⇒ Passwort (kann später geändert werden)
 - ⇒ Datenbank (für jeden Benutzer ist eine eingerichtet)
- Die Datenbank kann über das Tool PHPMYADMIN bearbeitet werden
- oder über ein Programm (z.B. in PHP) genutzt werden.

Das Administrationstool PHPMYADMIN erlaubt es die MySQL-Tabellen zu verwalten

- Webapplikation, erreichbar über Browser, meist auf selbem Server installiert wie MySQL selbst (auf Mars: <http://mars.iuk.hdm-stuttgart.de/phpmyadmin>)
- Einloggen mit
 - ⇒ Datenbank-Benutzername (z.B. `xy999`) und
 - ⇒ Datenbank-Passwort `*****`
- Datenbank auswählen (z.B. `u-xy999`) oder erstellen (Letzteres nicht für alle Nutzer erlaubt)
- Tabelle auswählen oder neue Tabelle erstellen
- Verschiedene Ansichten: Struktur („Entwurfsansicht“), Anzeigen (Inhalt), SQL (Abfrageeditor), Suche usw.



PHPMYADMIN: NEUE TABELLE ANLEGEN

phpMyAdmin

Server: localhost

Datenbanken SQL Status Exportieren Mehr

Datenbanken

Neue Datenbank anlegen: ?

Keine Rechte

⚠ Bitte beachten Sie: Das Aktivieren der Datenbankstatistiken kann starken Traffic zwischen dem Web- und dem MySQL-Server zur Folge haben.

Datenbank	Kollation
information_schema	utf8_general_ci
u-riekert	latin1_swedish_ci
Insgesamt: 2	latin1_swedish_ci

• [Datenbankstatistiken aktivieren](#)

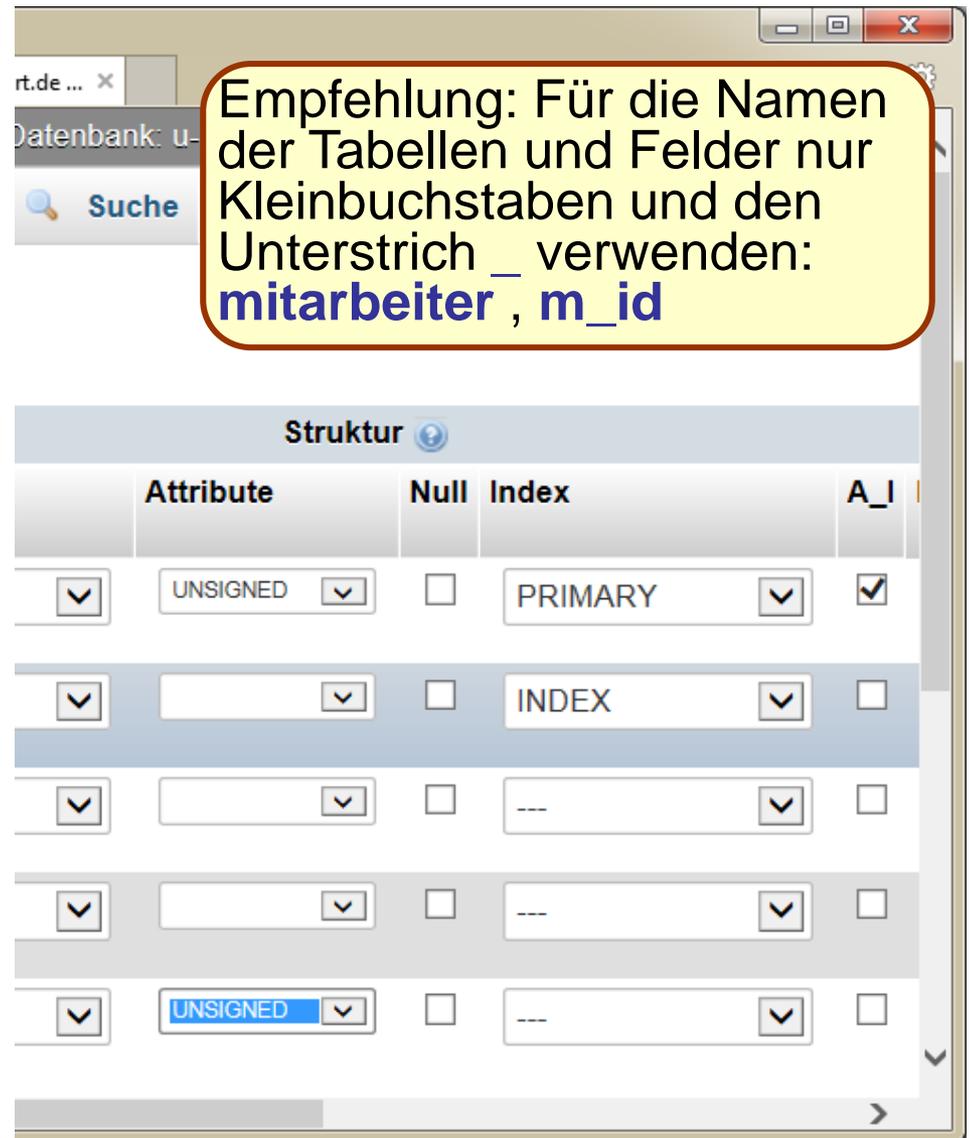
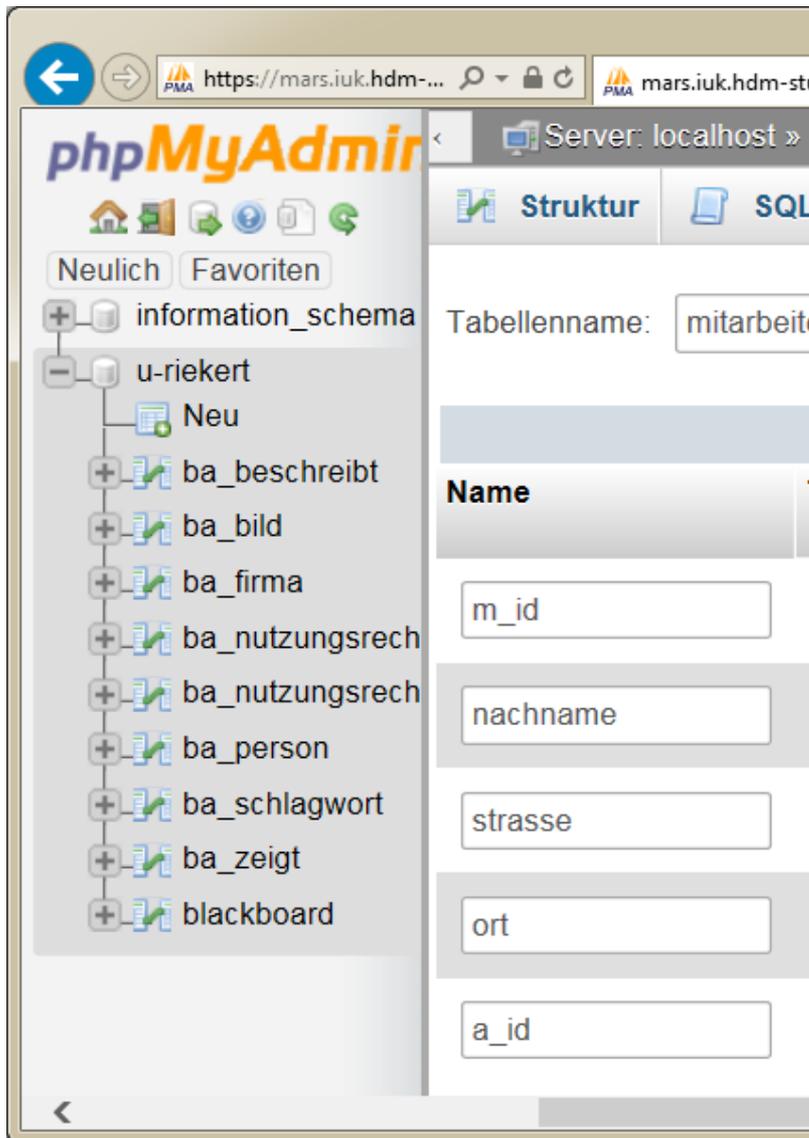
PHPMYADMIN: NEUE TABELLE DEFINIEREN (1)

The screenshot shows the phpMyAdmin interface in a browser window. The address bar shows the URL `https://mars.iuk.hdm-stuttgart.de`. The main header indicates the server is `localhost` and the database is `u-riekert`. The 'Struktur' (Structure) tab is active, showing the process of defining a new table named `mitarbeiter`. The 'Tabellenname' field contains `mitarbeiter` and the 'Spalte(n)' (Columns) field is set to `1`. Below this, a table lists the columns to be created:

Name	Typ	Länge/Werte	Standard
<input type="text" value="m_id"/>	INT	<input type="text"/>	Kein(e)
<input type="text" value="nachname"/>	VARCHAR	50	Kein(e)
<input type="text" value="strasse"/>	VARCHAR	50	Kein(e)
<input type="text" value="ort"/>	VARCHAR	50	Kein(e)
<input type="text" value="a_id"/>	INT	<input type="text"/>	Kein(e)

The left sidebar shows the database structure, including the `u-riekert` database and a 'Neu' (New) button for creating a new table. The 'Neu' button is highlighted, indicating the current step in the process.

PHPMYADMIN: NEUE TABELLE DEFINIEREN (2)



PHPMYADMIN: STRUKTUR DER TABELLE

Server: localhost » Datenbank: u-riekert » Tabelle: mitarbeiter

Anzeigen Struktur SQL Suche Einfügen Mehr

#	Name	Typ	Kollation	Attribute	Null	Standard	Extra
<input type="checkbox"/>	1 m_id	int(10)		UNSIGNED	Nein	kein(e)	AUTO_INCRE
<input type="checkbox"/>	2 nachname	varchar(50)	latin1_german1_ci		Nein	kein(e)	
<input type="checkbox"/>	3 strasse	varchar(50)	latin1_german1_ci		Nein	kein(e)	
<input type="checkbox"/>	4 ort	varchar(50)	latin1_german1_ci		Nein	kein(e)	
<input type="checkbox"/>	5 a_id	int(10)		UNSIGNED	Nein	kein(e)	

Alle auswählen markierte: Anzeigen Bearbeiten Löschen

Primärschlüssel Unique Index

Druckansicht Beziehungsübersicht Tabellenstruktur analysieren

Tabelle verfolgen Spalten verschieben

1 Spalte(n) einfügen An das Ende der Tabelle An den Anfang der

Tabelle Nach m_id OK

+ Indizes

PHPMYADMIN: TABELLE FÜLLEN (1)

The screenshot shows the phpMyAdmin interface in a browser window. The address bar shows the URL `https://mars.iuk.hdm-stuttgart.de`. The main content area displays the structure of the 'mitarbeiter' table in the 'u-riekert' database. The table has columns: m_id (int(10) unsigned), nachname (varchar(50)), strasse (varchar(50)), ort (varchar(50)), and a_id (int(10) unsigned). The 'a_id' column is highlighted in blue, indicating it is the primary key. Below the structure, there is a data entry form with input fields for each column. The 'nachname' field contains 'Maier', the 'strasse' field contains 'Bahnhofstr. 3', and the 'ort' field contains 'Neu-Ulm'. The 'a_id' field contains '3'. There is also a section for 'Ignorieren' with a checkbox and a table structure for the 'mitarbeiter' table with columns: m_id (int(10) unsigned), nachname (varchar(50)), and strasse (varchar(50)). The 'nachname' field contains 'Huber' and the 'strasse' field contains 'Türlestr. 32'.

Spalte	Typ	Funktion	Null	Wert
m_id	int(10) unsigned			
nachname	varchar(50)			Maier
strasse	varchar(50)			Bahnhofstr. 3
ort	varchar(50)			Neu-Ulm
a_id	int(10) unsigned			3

Spalte	Typ	Funktion	Null	Wert
m_id	int(10) unsigned			
nachname	varchar(50)			Huber
strasse	varchar(50)			Türlestr. 32

PHPMYADMIN: TABELLE FÜLLEN (2)

The screenshot shows the phpMyAdmin interface for a MySQL database. The browser address bar shows the URL `https://mars.iuk.hdm-stuttgart.de`. The interface title is `Server: localhost » Datenbank: u-riekert » Tabelle: mitarbeiter`. The left sidebar shows a tree view of databases and tables, with `u-riekert` expanded to show a list of tables including `mitarbeiter`. The main area has tabs for `Anzeigen`, `Struktur`, `SQL`, `Suche`, `Einfügen`, and `Mehr`. A green notification box at the top states: `✓ 2 Datensätze eingefügt. ID der eingefügten Zeile: 2`. Below this, the SQL command is displayed:

```
INSERT INTO `u-riekert`.`mitarbeiter` (`m_id`, `nachname`, `strasse`, `ort`, `a_id`) VALUES (NULL, 'Maier', 'Bahnhofstr. 3', 'Neu-Ulm', '3'), (NULL, 'Huber', 'Türlenstr. 32', 'Stuttgart', '2');
```

 At the bottom, a box titled `SQL-Befehl(e) in Datenbank u-riekert ausführen:` shows the same SQL command being executed. To the right of the command, a list of columns is shown under the heading `Spalten`: `m_id`, `nachname`, `strasse`, `ort`, and `a_id`.

PHPMYADMIN: TABELLE ANZEIGEN

The screenshot shows the phpMyAdmin interface for a MySQL database. The browser address bar shows the URL `https://mars.iuk.hdm-stuttgart.de`. The interface is in German. The left sidebar shows a tree view of the database structure, with the 'mitarbeiter' table selected. The main area displays the table view for 'mitarbeiter' with the following data:

Server: localhost » Datenbank: u-riekert » Tabelle: mitarbeiter

Anzeigen Struktur SQL Suche Einfügen Exportieren Mehr

✓ Zeige Datensätze 0 - 1 (2 insgesamt, Die Abfrage dauerte 0.0004 Sekunden.)

```
SELECT * FROM `mitarbeiter`
```

Messen [Inline] [Bearbeiten] [SQL erklären] [PHP-Code erzeugen] [Aktualisieren]

Anzahl der Datensätze: 25 Zeilen filtern: Diese Tabelle durchsuch

Nach Schlüssel sortieren: keine

+ Optionen

				m_id	nachname	strasse	ort	a_id			
<input type="checkbox"/>		Bearbeiten		Kopieren		Löschen	1	Maier	Bahnhofstr. 3	Neu-Ulm	3
<input type="checkbox"/>		Bearbeiten		Kopieren		Löschen	2	Huber	Türlestr. 32	Stuttgart	2

Alle auswählen markierte: Bearbeiten Löschen Exportieren

Anzahl der Datensätze: 25 Zeilen filtern: Diese Tabelle durchsuch

Operationen für das Abfrageergebnis

PROBLEME BEIM DATENBANKENTWURF

StudierendeStudiengang				
Matrikelnr	Name	Kürzel	Studiengang	
29401	Fatma Gül	WIB	Wirtschaftsinformatik (B.Sc.)	
31634	Sarah Hoppe	BIB	Bibliotheks- und Informationsmanagement (B.A.)	
32013	Nils Lindström	BIM	Bibliotheks- und Informationsmanagement (M.A.)	
32014	Marco Barber	WIB	Wirtschaftsinformatik (B.Sc.)	
32045	Ben Adelman	IDB	Informationsdesign (B.A.)	

Das Definieren von Datenbanktabellen ist eigentlich nicht schwer.

Trotzdem kann man beim Entwurf einer Datenbank vieles falsch machen.

Wo liegen die Probleme bei dieser Tabelle?

- Datenfelder sollen nur **elementare Werte** enthalten!
 - ⇒ Im vorigen Beispiel: Nachname und Vorname in unterschiedlichen Feldern speichern.
- **Redundanzen vermeiden!** Redundanz bedeutet, dass Daten ohne Informationsverlust weggelassen werden könnten.
 - ⇒ In dieser Tabelle wird der Zusammenhang zwischen Studiengang und Studiengangkürzel redundant (d.h. überflüssigerweise mehrfach) gespeichert.
 - ⇒ Das Problem besteht nicht darin, dass der Wert „WIB“ mehrfach in der Tabelle steht. Es besteht auch nicht darin, dass „WIB“ und „Wirtschaftsinformatik (B.Sc.)“ dasselbe bedeutet.
 - ⇒ **Das Problem besteht darin, dass der Zusammenhang zwischen „WIB“ und „Wirtschaftsinformatik (B.Sc.)“ mehrfach in der Tabelle steht**

Redundanz vermeiden!

- **Kompliziertere Informationen in mehreren Tabellen repräsentieren!** Z.B. eine Tabelle für Studierende, eine zweite Tabelle für Studiengänge. (Dies wird im Folgenden noch erklärt.)
- **Tabellenfelder weglassen, die aus anderen berechnet werden können!** Beispiel: In der nachfolgenden Bestellliste ist das Feld Gesamtpreis überflüssig. Wenn benötigt, kann es durch Abfragen berechnet werden.

Ware	Anzahl	Einzelpreis	Gesamtpreis
Cola	2	1,60	3,20
Bier	1	2,80	2,80
Tee	4	1,80	7,20

Redundantes
Feld: In Tabelle
weglassen!

Der Aufbau von Datenbanken erfolgt in vier Schritten:

1. **Anforderungsanalyse**: Erfassung der zum Aufbau der Datenbank erforderlichen Sachverhalte in der Sprache der künftigen Anwender
2. **Entitäten-Beziehungsmodell**: Entwurf einer graphischen Repräsentation der darzustellenden Objekte („Entitäten“) und ihrer Beziehungen
3. Umsetzung des Entitäten-Beziehungsmodells durch **Definition** von geeigneten **Tabellen**
⇒ **Datendefinition**
4. **Füllen** der Datenbank mit aktuellen **Inhalten**
⇒ **Datenmanipulation**

AUFBAU VON DATENBANKEN: 1. ANFORDERUNGSANALYSE

Anforderungsanalyse:

Erfassung der in der Datenbank abzubildenden Sachverhalte in der Sprache der künftigen Anwender.

Typischer Ablauf einer „Anforderungsanalyse“:

Der Datenbankentwickler führt ein Gespräch mit den künftigen Anwendern der Datenbank, also mit seinen Kunden. Er erfragt die Anforderungen, d.h. welche Art von Informationen in der Datenbank repräsentiert werden sollen. Das Ergebnis dieses Gesprächs wird in einem kleinen Protokoll festgehalten.

BEISPIEL FÜR DAS ERGEBNIS EINER ANFORDERUNGSANALYSE

In einer Datenbank sollen Sachverhalte der folgenden Art repräsentiert werden können:

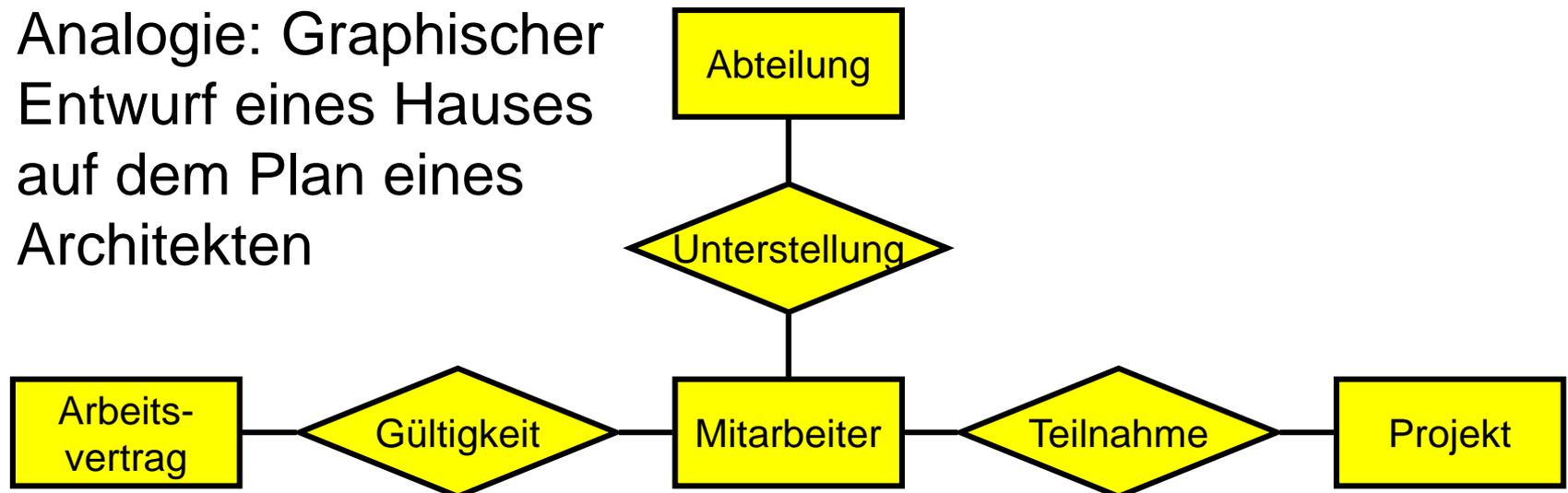
- Jede **Abteilung** besitzt eine Bezeichnung.
- Jeder **Mitarbeiter** ist charakterisiert durch einen Namen sowie durch Straße und Ort, in welchen er wohnt.
- Jeder **Arbeitsvertrag** legt eine Funktion (z.B. „Buchhalter“) eines Mitarbeiters fest sowie das Gehalt, das er verdient.
- Jedes **Projekt** besitzt einen Namen und eine eindeutige Projektnummer.
- Für jeden Mitarbeiter ist ein Arbeitsvertrag **gültig**, jeder Arbeitsvertrag ist genau für einen Mitarbeiter gültig.
- Jeder Mitarbeiter ist einer Abteilung **unterstellt**.
- Mitarbeiter können gleichzeitig an mehreren Projekten **teilnehmen**, wobei die jeweiligen Prozentanteile erfasst werden.

AUFBAU VON DATENBANKEN:

2. ENTITÄTEN-BEZIEHUNGSMODELL

Entitäten-Beziehungsmodell (*entity relationship model - ERM*)

- Graphischer Entwurf der Datenbank
- Alle wesentlichen Elemente werden graphisch dargestellt
- Umsetzung der Anforderungsanalyse in ein erstes Design
 - ⇒ dient auch zur Kommunikation mit dem Kunden
 - ⇒ und zur Verifizierung der Anforderungsanalyse
- Analogie: Graphischer Entwurf eines Hauses auf dem Plan eines Architekten



Elemente eines Entitäten-Beziehungsmodells

 **Entität** (*entity*): Wohlunterscheidbares Objekt der realen Welt oder unserer Vorstellung. Beispiele für Entitäten: Individuen, Gegenstände, Begriffe, Ereignisse.

⇒ Entitäten gleichen Typs bilden sog. **Entitätsmengen** (alternativer Name: Entitätstypen, engl.: *entity types*) und besitzen bestimmte Merkmale.

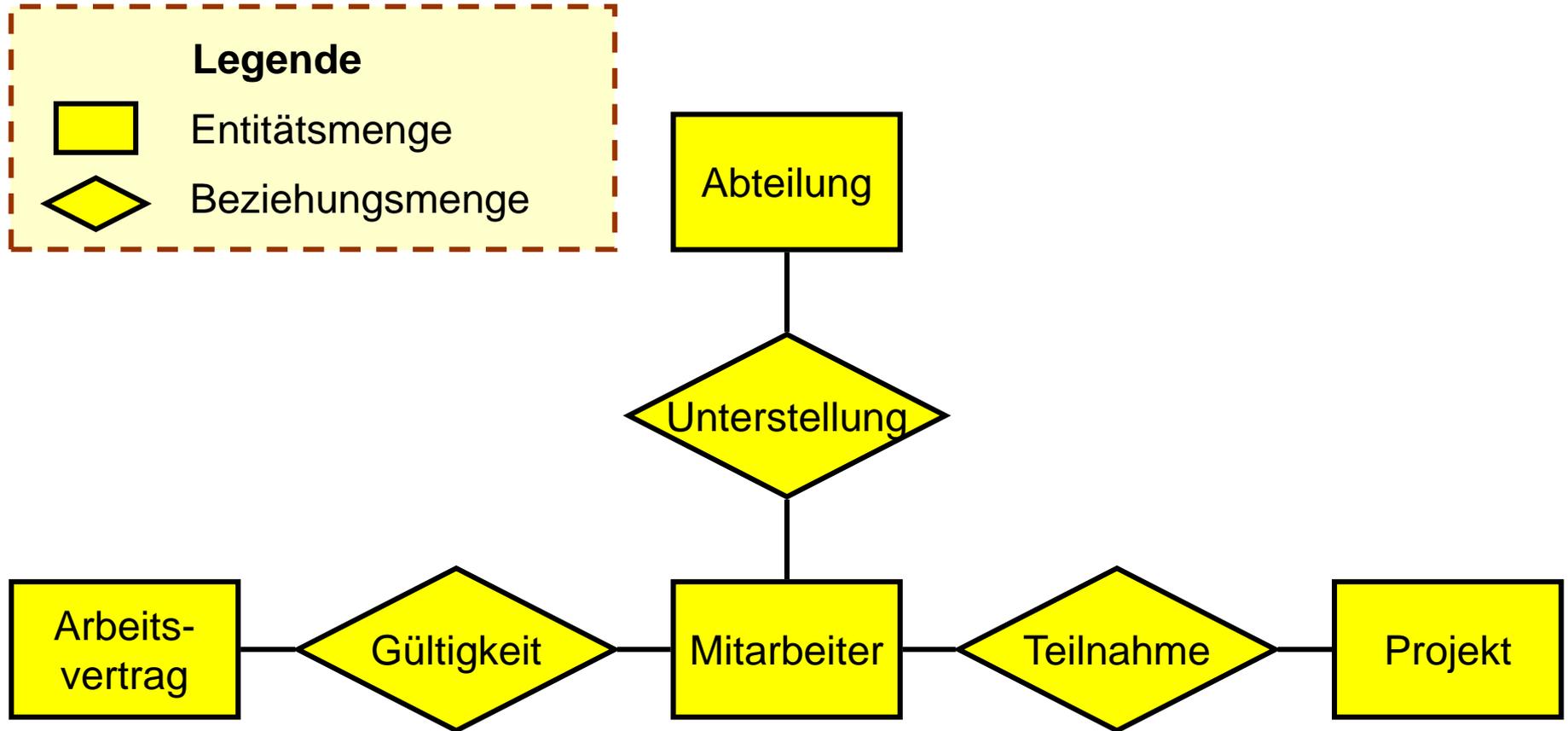
⇒ Entitäten besitzen eindeutigen Identifikationsschlüssel.

 **Beziehung** (*relationship*):

⇒ stellt eine Beziehung zwischen Entitäten dar.

⇒ Beziehungen gleichen Typs bilden **Beziehungsmengen** (auch: Beziehungstypen, engl.: *relationship types*) und können zusätzlich bestimmte Merkmale tragen.

ENTITÄTEN-BEZIEHUNGSMODELL: GRAPHISCHE DARSTELLUNG



BEISPIEL: ENTITÄT MAIER UND ENTITÄTSMENGE MITARBEITER

- Entität: **Mitarbeiter Maier**, wohnhaft in der **Türlenstraße** in **Stuttgart**
- Entitätsmenge: Menge aller **Mitarbeiter** mit den Merkmalen **Name**, **Straße** und **Ort**
- Identifikationsschlüssel: **Mitarbeiternummer** als künstlicher Schlüssel

Darstellung im Entitäten-Beziehungsmodell:



BEISPIEL: BEZIEHUNG UND BEZIEHUNGSMENGE

Beziehung: Mitarbeiter **Maier arbeitet** zu **70%** im Projekt Nr. **17**

Beziehungsmenge: Menge aller **Mitarbeiter-Projekt-Teilnahmen** mit den Merkmalen **Mitarbeiternummer**, **Projektnummer** und **Prozentanteil**

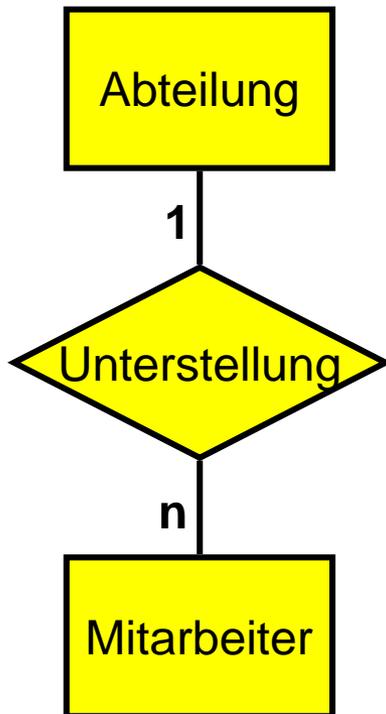
Identifikationsschlüssel: Zusammengesetzter Schlüssel aus **Mitarbeiter- und Projektnummer**

Darstellung im Entitäten-Beziehungsmodell:



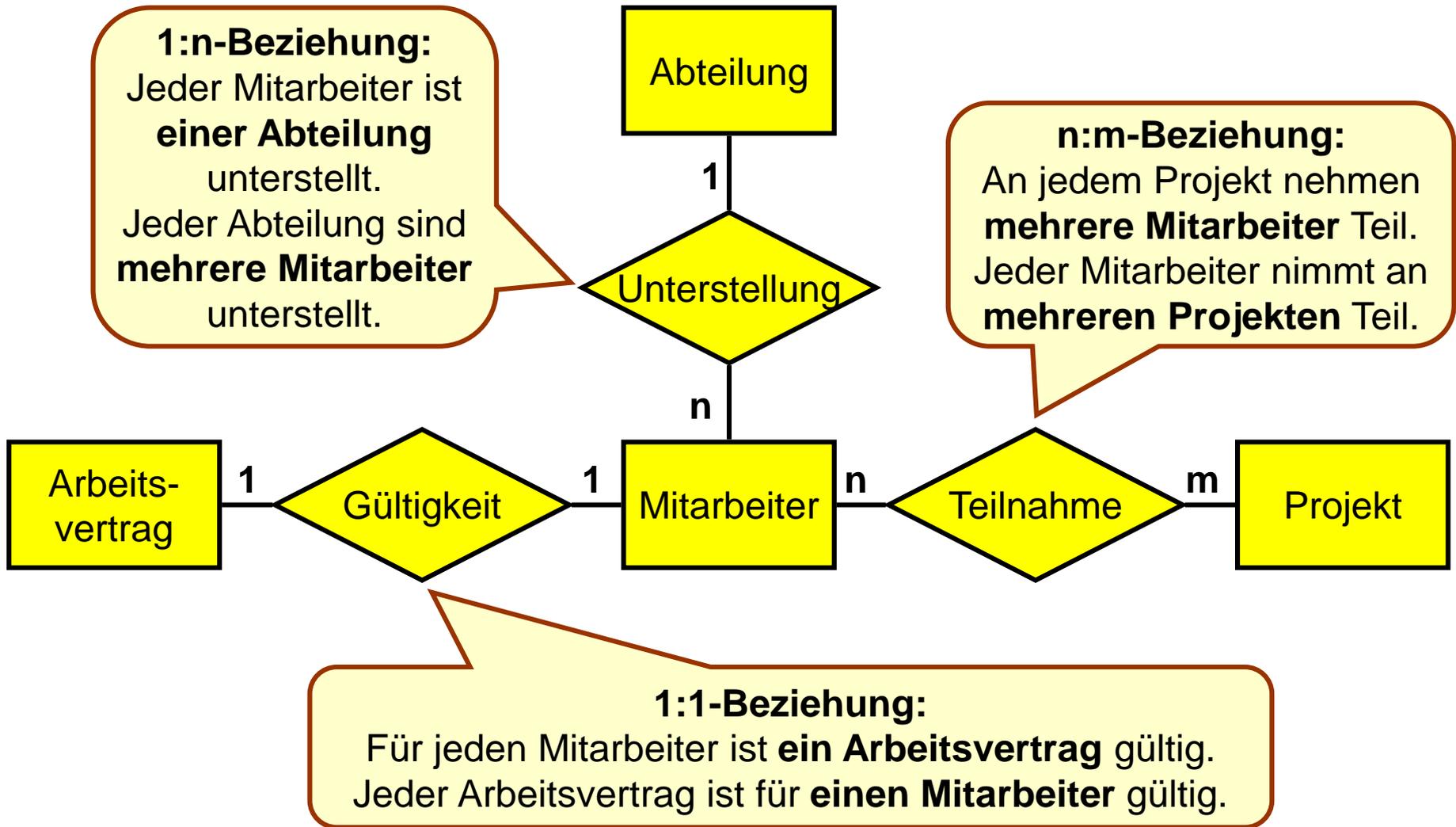
MÄCHTIGKEIT VON BEZIEHUNGSMENGEN (1)

Beziehungsmengen können sich darin unterscheiden, wie „viele“ Entitäten sie einander zuordnen können.

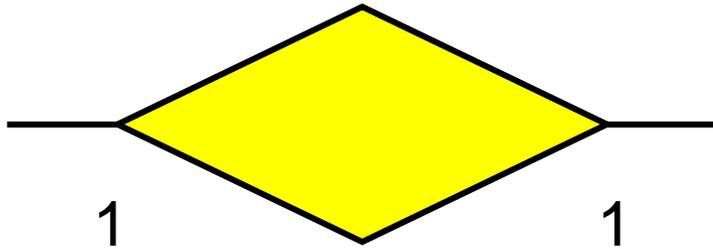


- Beispielsweise kann für die Beziehungsmenge „Unterstellung“ festgelegt werden, dass jeder Mitarbeiter **einer** Abteilung zugeordnet ist, jede Abteilung jedoch **mehrere** Mitarbeiter besitzt.
- Diese Eigenschaft der Beziehungsmenge wird als **Mächtigkeit** (alternative Namen: **Kardinalität, Assoziationstyp**) bezeichnet.
- Die Mächtigkeit wird mit Hilfe von Zahlen (**1** für „ein(e)“) und Symbolen (**n, m** oder ***** für „mehrere“) notiert.

MÄCHTIGKEIT VON BEZIEHUNGSMENGEN (2)

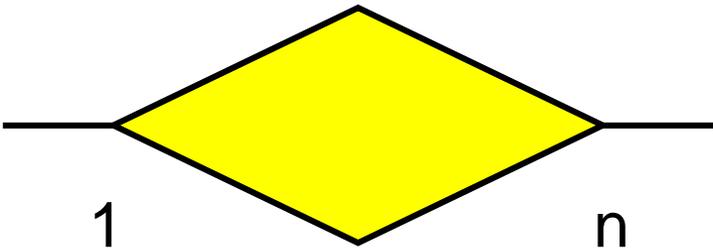


MÄCHTIGKEIT VON BEZIEHUNGSMENGEN (3)



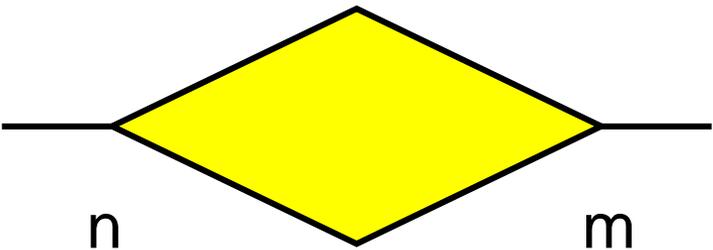
1:1-Beziehung

einfach-einfache Beziehung
(engl.: *one-to-one relationship*)



1:n-Beziehung

einfach-komplexe Beziehung
(engl.: *one-to-many relationship*)



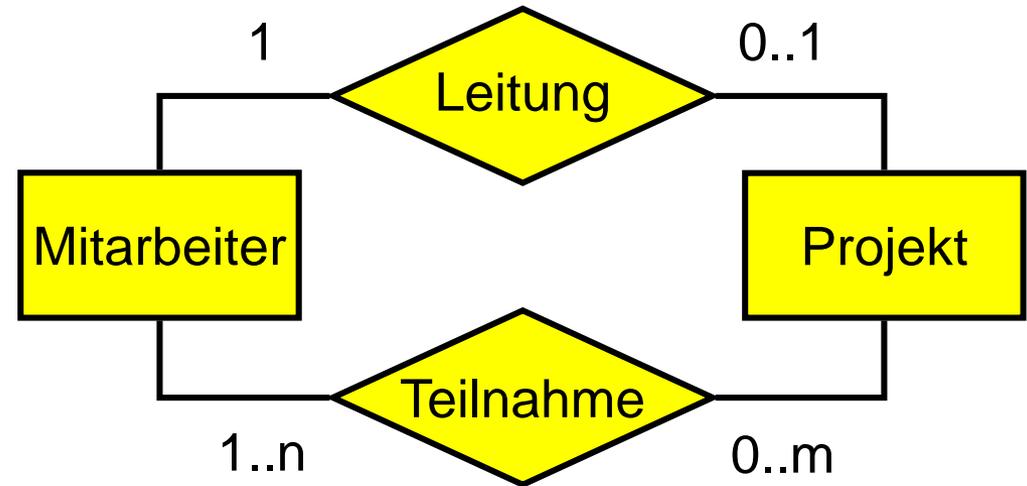
n:m-Beziehung

komplex-komplexe Beziehung
(*many-to-many relationship*)

INTERVALLDARSTELLUNG FÜR DIE MÄCHTIGKEIT

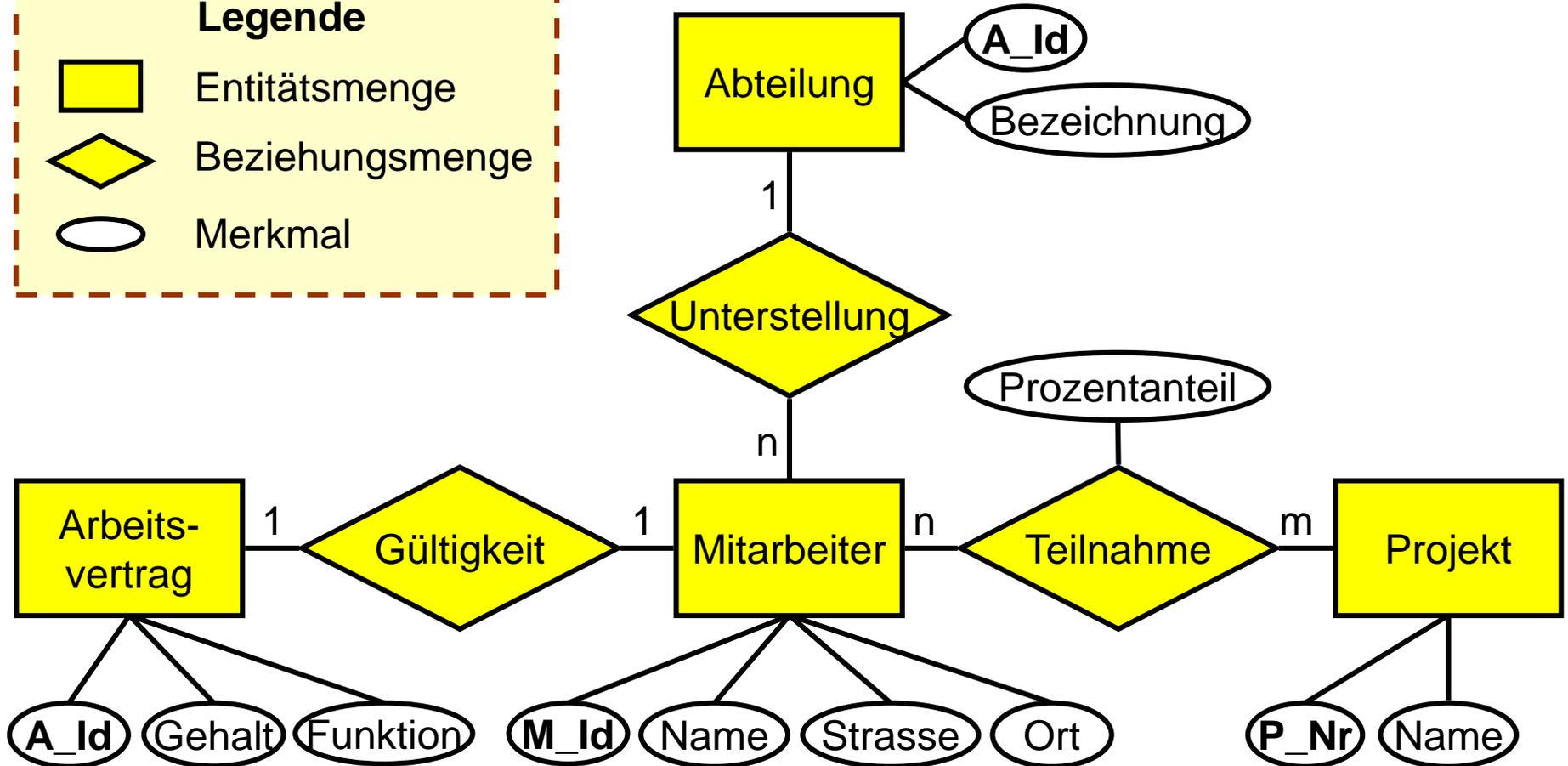
In der so genannten Intervalldarstellung lässt sich die Mächtigkeit von Beziehungsmengen durch Angabe von Untergrenzen noch präziser festlegen.

Beispiel:

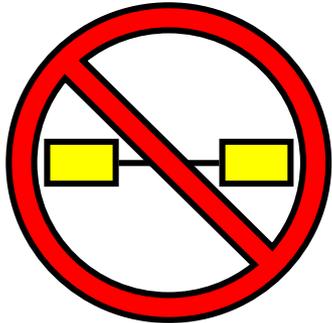


- Jeder Mitarbeiter kann ein Projekt leiten, muss aber nicht: Mächtigkeit **0..1**, d.h. zwischen 0 und 1
- Jeder Mitarbeiter kann an mehreren Projekten teilnehmen, muss aber nicht: Mächtigkeit **0..m**, d.h. zwischen 0 und m
- An jedem Projekt nimmt mindestens ein Mitarbeiter teil: Mächtigkeit **1..n**, d.h. zwischen 1 und n

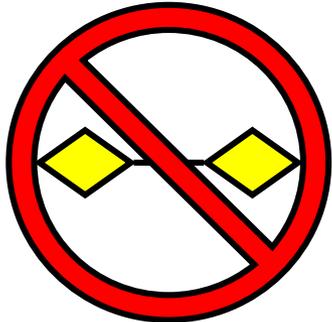
ENTITÄTEN-BEZIEHUNGSMODELL: DARSTELLUNG MIT MERKMALEN



Entitäten-Beziehungsmodelle besitzen eine „Syntax“:



- Zwei Entitätsmengen (Rechtecke) dürfen nie direkt nebeneinander liegen; sie dürfen nur über eine Beziehungsmenge (Raute) miteinander verbunden werden.



- Zwei Beziehungsmengen (Rauten) dürfen nie direkt nebeneinander liegen; dazwischen muss immer eine Entitätsmenge (ein Rechteck) liegen.



- Merkmale (Ellipsen) können nur direkt entweder an eine einzige Entitätsmenge (Rechteck) oder eine einzige Beziehungsmenge (Raute) angehängt werden.

AUFBAU VON DATENBANKEN:

3. DATENDEFINITION

Als nächster Schritt muss das Entitäten-Beziehungsmodell in Form von Tabellen umgesetzt werden.

Dies ist der Vorgang der **Datendefinition**. Darunter verstehen wir die Festlegung des sogenannten **Datenbankschemas**, das die **Struktur** einer Datenbank beschreibt:

- Namen der Tabellen und ihrer Felder
- allgemeine Eigenschaften zu den Tabellen und Feldern (z.B. Wertebereiche, Beziehungen, Integritätsbedingungen)

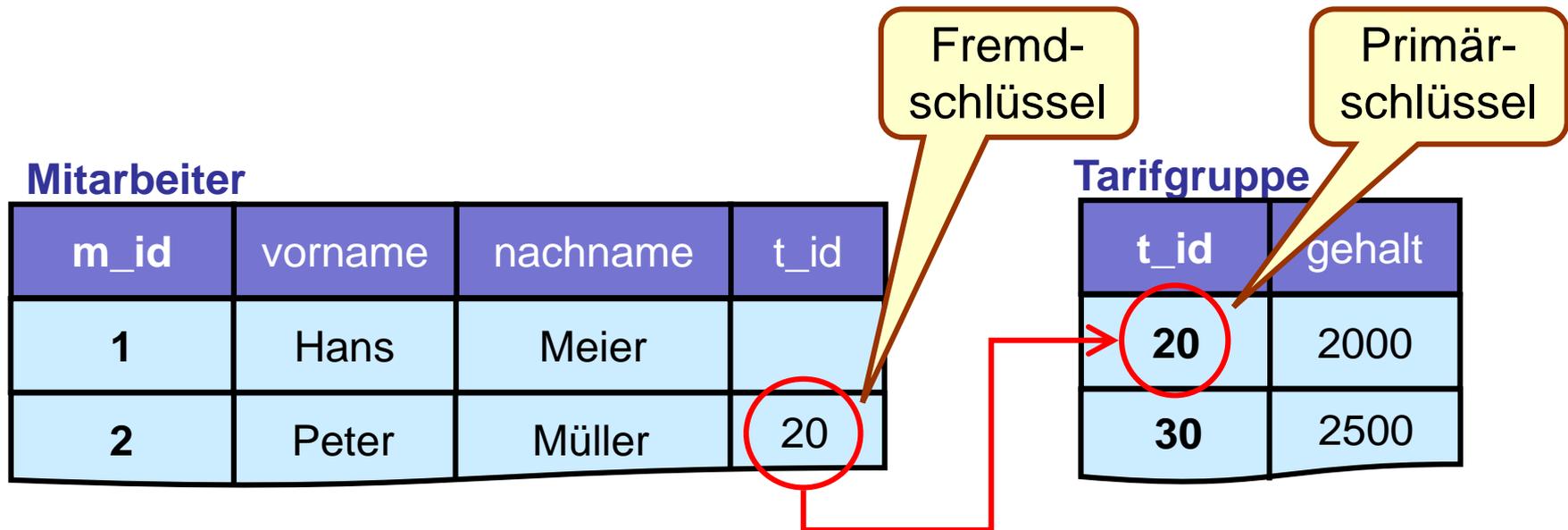
*Der konkrete **Inhalt** der Datenbank ist kein Gegenstand der Datendefinition. Der Inhalt ist Gegenstand des 4. Schritts, der **Datenmanipulation** genannt wird. Die Datenmanipulation wird in einem separaten Vorlesungsabschnitt behandelt.*

PRIMÄRSCHLÜSSEL

- Jede ordnungsgemäß definierte Tabelle besitzt einen so genannten **Primärschlüssel**.
- Im einfachsten Fall ist das ein Feld der Tabelle, dessen Werte die Datensätze der Tabelle **eindeutig** bestimmt.
 - ⇒ Oft verwendet: so genannte **künstliche** Primärschlüssel, die in der Realität keine Entsprechung haben, z.B. automatisch hochgezählte Nummern („**Autoinkrement**“).
 - ⇒ Beispiel: Matrikelnummer in der Tabelle Studierende
- Als Primärschlüssel können aber auch **Feldkombinationen** dienen, wenn durch diese jeder Datensatz in der Tabelle eindeutig bestimmt ist. Diese Feldkombinationen sollten allerdings minimal sein, d.h. es kann keines der Felder weggelassen werden, ohne die Eindeutigkeit aufzugeben.
 - ⇒ Beispiel: Kombination aus Ortsvorwahl und Rufnummer in einer Tabelle für Telefonanschlüsse.

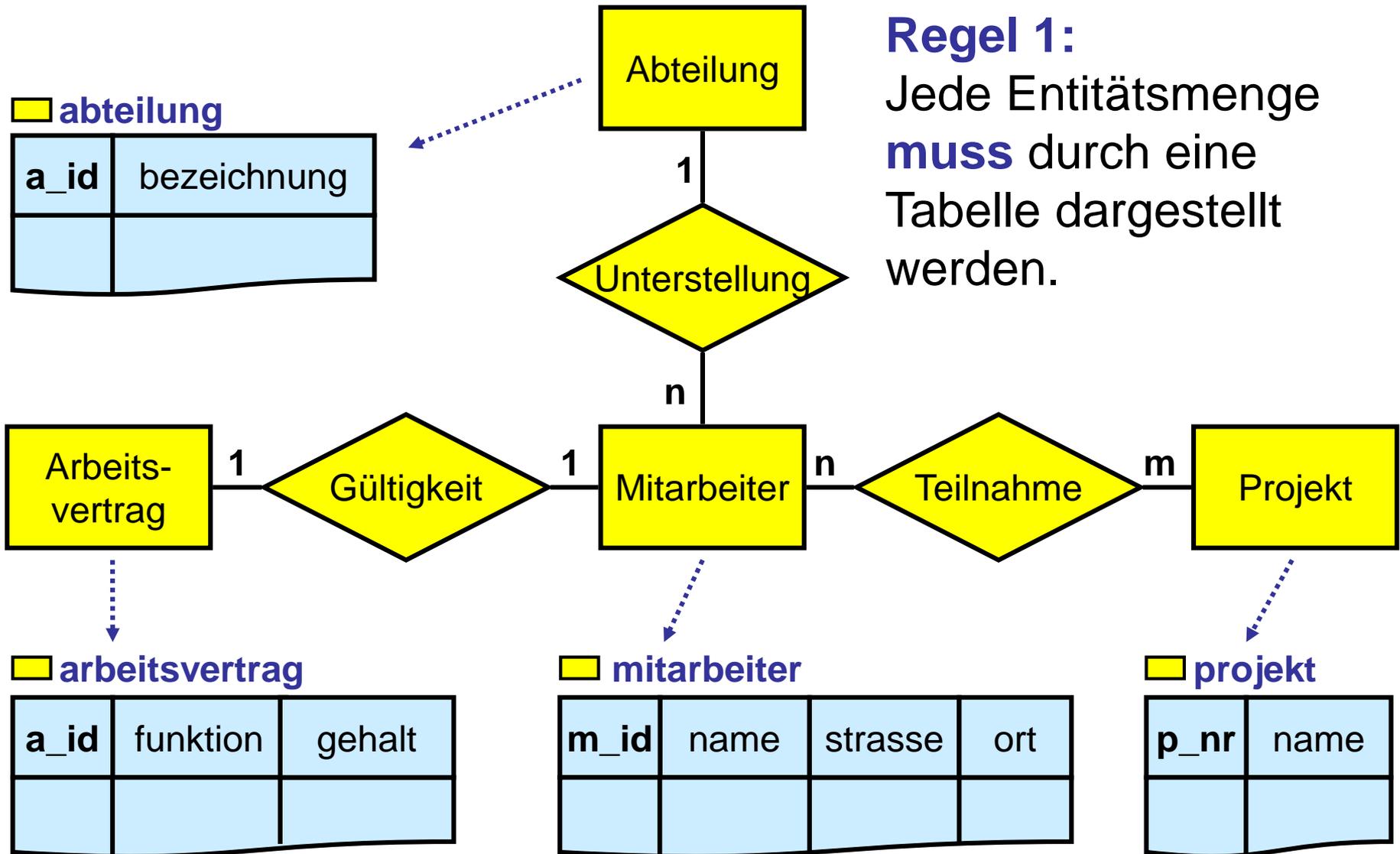
FREMSCHLÜSSEL

Die Werte von Primärschlüsseln können auch in weiteren Feldern auftauchen; solche Felder nennt man **Fremdschlüssel**. Sie werden verwendet, um von einem Datensatz auf einen anderen Datensatz zu verweisen. So können in Tabellen Beziehungen dargestellt werden

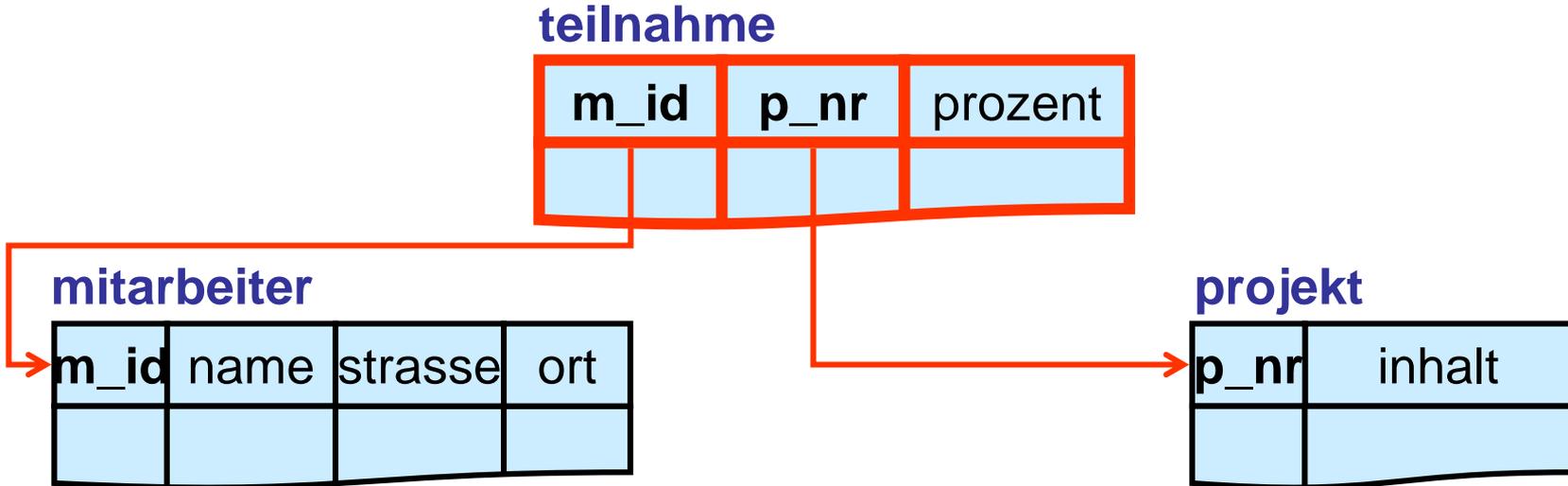


- **Regel 1:** Jede **Entitätsmenge muss** durch eine eigenständige Tabelle dargestellt werden:
 - ⇒ Die Merkmale der Entitätsmenge werden zu Feldern der Tabelle
 - ⇒ Es gibt einen Primärschlüssel, in der Regel ist das der Identifikationsschlüssel der Entitätsmenge
- **Regel 2:** Jede **Beziehungsmenge kann** durch eine eigenständige Tabelle dargestellt werden, doch ist das nicht in jedem Fall nötig.
 - ⇒ Genaueres sagen die **Regeln 3, 4 und 5** (siehe folgende Folien)

REGEL 1: MUSS-REGEL FÜR ENTITÄTSMENGEN



REGEL 3 FÜR KOMPLEX-KOMPLEXE BEZIEHUNGEN



Regel 3: Komplex-komplexe Beziehungsmengen müssen als **eigenständige Tabellen** definiert werden. Primärschlüssel der Beziehungsmengentabelle ist häufig die Kombination der beiden Fremdschlüssel oder ein künstlicher Schlüssel.

REGEL 4 FÜR EINFACH-KOMPLEXE BEZIEHUNGEN



mitarbeiter

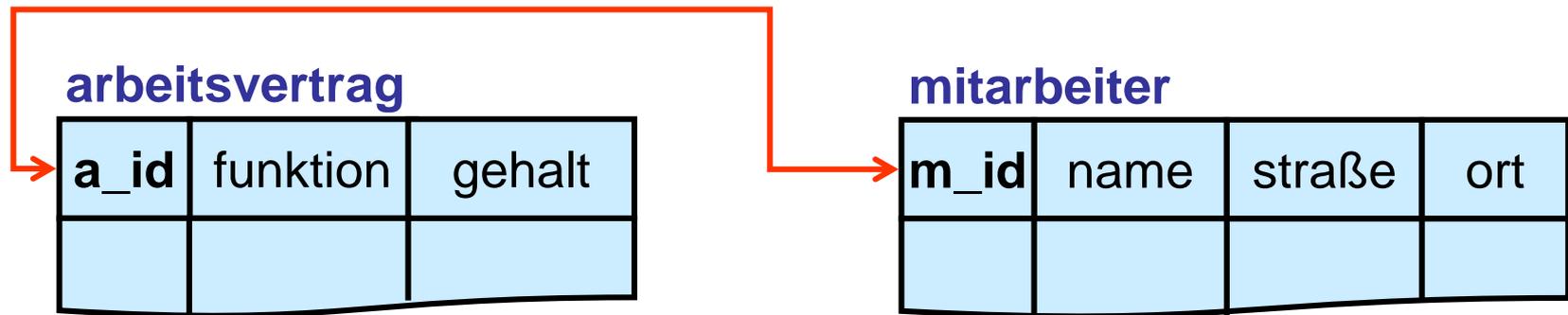
m_id	name	strasse	ort	a_id

abteilung

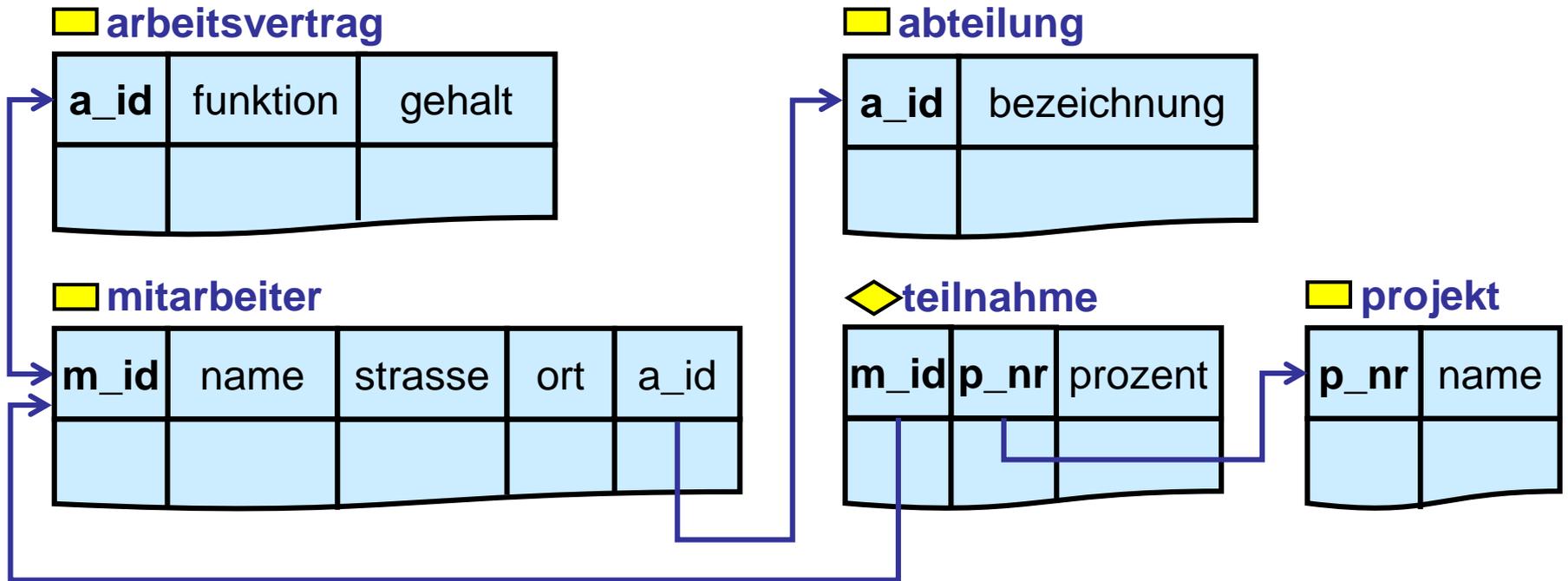
a_id	bezeichnung

Regel 4: Einfach-komplexe Beziehungsmengen können ohne eigenständige Beziehungsmengentabelle definiert werden. Der Entitätentabelle auf der „komplexen“ Seite der Beziehung (markiert durch „n“) wird ein Fremdschlüssel auf die andere Entitätentabelle zusammen mit eventuellen weiteren Merkmalen der Beziehungsmenge hinzugefügt.

REGEL 5 FÜR EINFACH-EINFACHE BEZIEHUNGEN



Regel 5: Einfach-einfache Beziehungsmengen können **ohne eigenständige Beziehungsmengentabelle** definiert werden. Zusammengehörige Datensätze werden entweder durch identische Primärschlüssel gekennzeichnet (wie hier) oder eine der Tabellen erhält einen Fremdschlüssel, der auf den Primärschlüssel in der anderen Tabelle verweist (wie bei einfach-komplexen Beziehungen, siehe letzte Folie).



Die fettgedruckten Merkmale bzw. Merkmalskombinationen (**a_id**, **m_id**, **p_nr**) sind Primärschlüssel.

Die Merkmale, von denen Pfeile ausgehen, sind Fremdschlüssel. Die Pfeile enden mit ihren Spitzen bei den Primärschlüsseln.

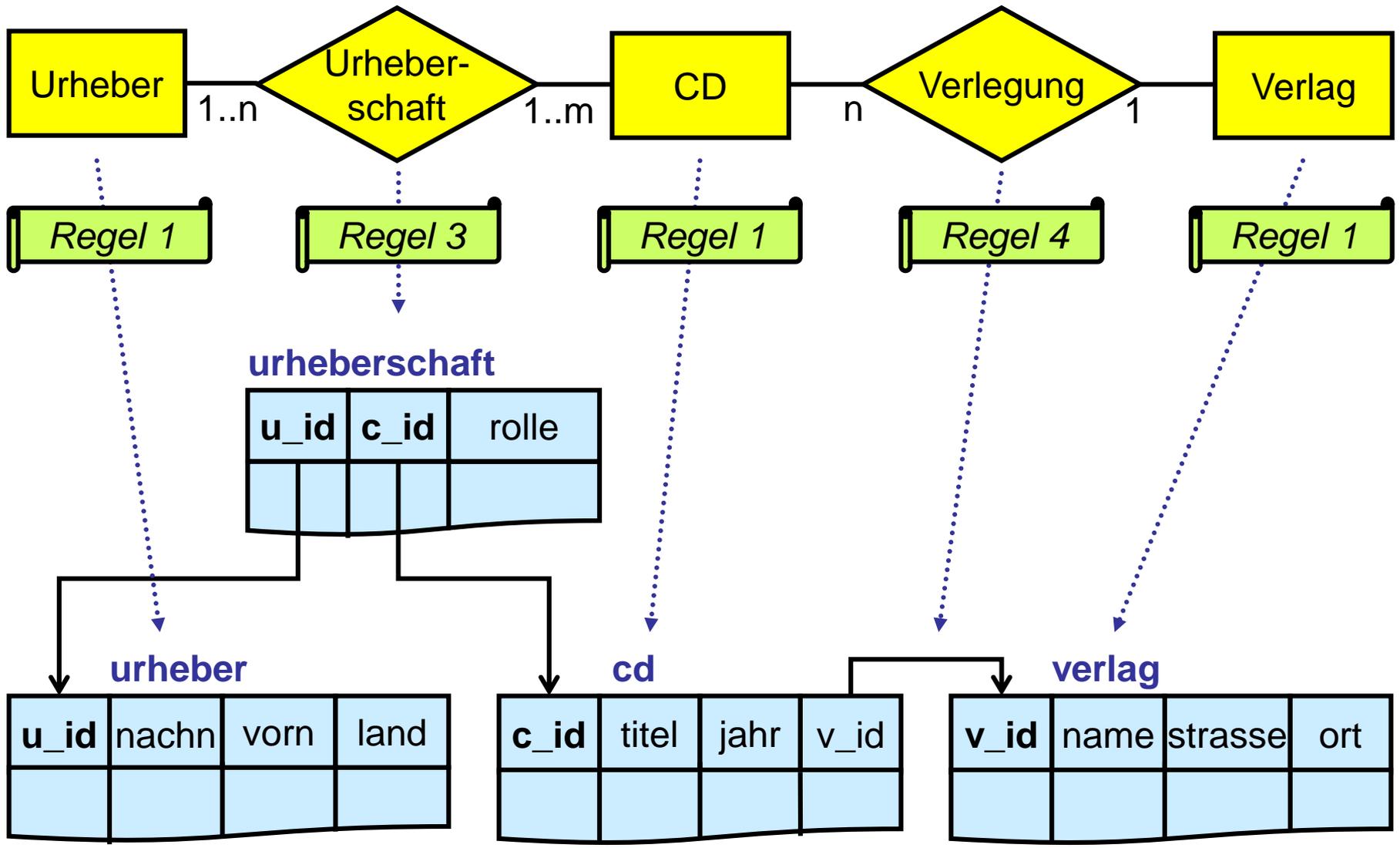
m_id und **p_nr** in **teilnahme** sind für sich Fremdschlüssel und in der Kombination Primärschlüssel. **a_id** in Arbeitsvertrag und **m_id** in Mitarbeiter sind zugleich Primär- und Fremdschlüssel.

- **Einfach-einfache** und **einfach-komplexe Beziehungsmengen** lassen sich mit Hilfe von Fremdschlüsselfeldern in einer der verknüpften Entitätstabellen darstellen.
 - ⇒ Beispiele: **m_id** in Arbeitsvertrag, **a_id** in Mitarbeiter
- Für **komplex-komplexe Beziehungsmengen** benötigt man eine eigene Tabelle, die zwei Fremdschlüsselfelder enthält.
 - ⇒ Beispiel: Beziehungsmenge Teilnahme als Tabelle mit Fremdschlüsselfeldern **m_id** und **p_nr**
- Fremdschlüsselfelder haben denselben Felddatentyp wie die zugehörigen Primärschlüsselfelder

Erstellen Sie für das nachfolgend beschriebene Beispiel einer CD-Sammlung selbst ein Entitäten-Beziehungsmodell und überführen Sie dieses in eine Tabellendarstellung:

- Jede CD besitzt einen eindeutigen Identifikator, einen Titel, sowie ein Erscheinungsjahr
- Es gibt Urheber, die durch Nachnamen, Vornamen und ein Heimatland genauer definiert sind
- CDs haben einen oder mehrere Urheber und umgekehrt. Die Urheberschaft kann durch eine Rolle (z.B. Komponist, Interpret) genauer definiert sein.
- Jeder CD ist genau ein Verlag zugeordnet, der durch einen Namen und die üblichen Adressangaben genauer definiert ist.

ÜBUNGSBEISPIEL CD-SAMMLUNG: LÖSUNG



DEFINITION VON TABELLEN: STRUKTUR

The screenshot shows the phpMyAdmin interface for the 'firma_mitarbeiter' table. The table structure is as follows:

#	Name	Typ	Kollation	Attribute	Null	Standard	Extra
1	<u>m_id</u>	int(10)		UNSIGNED	Nein	kein(e)	AUTO_INCREMENT
2	nachname	varchar(80)	utf8_unicode_ci		Nein	kein(e)	
3	strasse	varchar(80)	utf8_unicode_ci		Nein	kein(e)	
4	ort	varchar(80)	utf8_unicode_ci		Nein	kein(e)	
5	a_id	int(10)		UNSIGNED	Nein	kein(e)	

Annotations:

- Primär-schlüssel**: Points to the m_id column.
- Merkmale**: Points to the column headers.
- Einfach-komplexe Beziehung als Fremdschlüssel, hier auf Tabelle Abteilung**: Points to the a_id column.

INDIZIERUNG VON FELDERN

- Ein Feld einer Tabelle kann auf Wunsch **indiziert** werden, d.h. die Datenbank besitzt dann für das Feld einen **Index** = ein sortiertes Verzeichnis der Datenwerte und der zugeordneten Datensätze
- Für die Benutzer ist der Index unsichtbar. Die Benutzung der Datenbank erfolgt mit und ohne Index auf identische Weise.
- Vorteil: Ein Index beschleunigt die Suche nach Datensätzen mit bestimmten Datenwerten im indizierten Feld.
- Nachteil (meist nicht gravierend): Das Einfügen neuer Datensätze in die Tabelle dauert etwas länger und es wird etwas mehr Speicherplatz auf der Festplatte benötigt
- Analogie: Stichwortverzeichnis in einem Buch (oft auch Index genannt).

REFERENZIELLE INTEGRITÄT

- Fremdschlüssel dienen dazu, um auf andere Datensätze zu verweisen.
- Die Werte der Fremdschlüssel dienen dabei als Verweise, auch „Referenzen“ genannt.
- Ein Verweis auf einen Datensatz wird hergestellt, indem dessen Primärschlüsselwert in das Fremdschlüsselfeld eingetragen wird.
- Diese Verweise sollen sinnvollerweise nicht ins Leere zeigen, d.h. **jeder Wert eines Fremdschlüssels sollte als Primärschlüsselwert in der referenzierten Tabelle (d.h. der Tabelle, auf die verwiesen wird) vorkommen.**
 - ⇒ Diese Eigenschaft nennt man **referenzielle Integrität.**

REFERENZIELLE INTEGRITÄT MIT MYSQL (1)

MYSQL kann die referenzielle Integrität überprüfen. Dazu im Tab Struktur ① ...

The screenshot shows the 'Struktur' (Structure) tab of a MySQL table. The table has five columns: #1 m_id (int(10), UNSIGNED, AUTO_INCREMENT), #2 nachname (varchar(80)), #3 strasse (varchar(80)), #4 ort (varchar(80)), and #5 a_id (int(10), UNSIGNED). The 'a_id' column is selected. Below the table list, there are options for 'Primärschlüssel' (Primary Key), 'Unique', and 'Index'. The 'Index' option is selected. At the bottom, there are options for 'Beziehungsübersicht' (Relationship Overview) and 'Struktur analysieren' (Analyze Structure).

② Fremdschlüssel
③ mit Index versehen

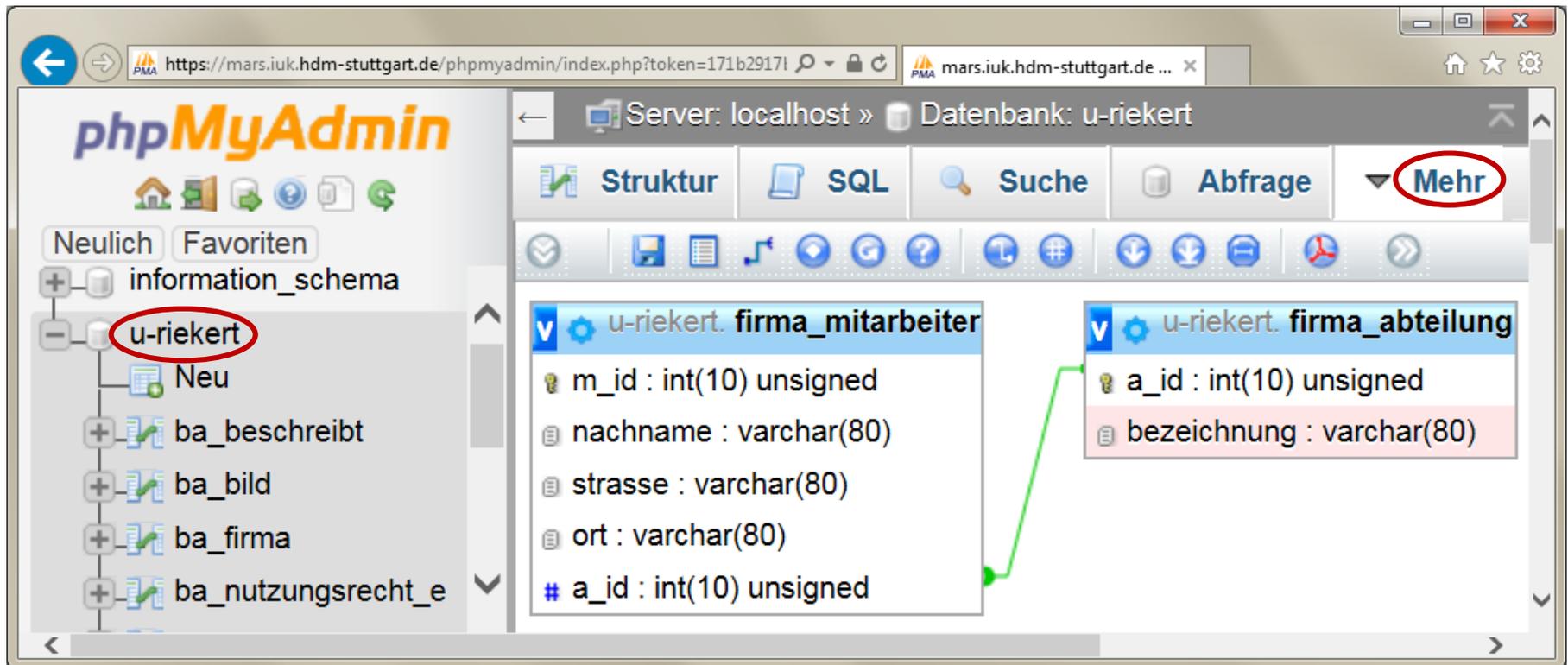
④ Anschließend in Beziehungsübersicht / „Beschränkung für ausw. Schlüssel“

durch Fremd-schlüssel referenziertes
⑤ Feld eintragen

The screenshot shows the 'Tabellenverknüpfungen' (Table Relationships) dialog box. It is configured for a foreign key relationship. The 'Spalte' (Column) is 'a_id'. The 'Interne Beziehung' (Internal Relationship) is 'u-riekert'. The 'Beschränkung für auswärtige Schlüssel (INNODB)' (Constraint for Foreign Keys) is 'firma_abteilung'. The 'a_id' field is selected in the dropdown menu. The 'Constraint-Name' field is empty. The 'ON DELETE' and 'ON UPDATE' actions are both set to 'RESTRICT'.

REFERENZIELLE INTEGRITÄT MIT MYSQL (2)

Beziehungen zwischen Fremd- und Primärschlüsseln lassen sich auch im „Designer“ anzeigen und bearbeiten (dazu Datenbank anwählen – Mehr – Designer).



REFERENZEN AUF DATENSÄTZE MIT HILFE VON FREMDSCHLÜSSELN

Tabelle **firma_abteilung**

a_id	bezeichnung
1	Einkauf
2	Produktion
3	Vertrieb

Primärschlüssel

Fremdschlüssel

Tabelle **firma_mitarbeiter**

m_id	nachname	strasse	ort	a_id
2	Huber	Türlenstr. 32	Stuttgart	2
1	Maier	Bahnhofstr. 3	Neu-Ulm	3
3	Weber	Nobelstr. 10	Stuttgart	3

Bei der Pflege von Tabellen ist es oft schwer, die referenzielle Integrität aufrecht zu erhalten.

Dies gilt vor allem, wenn die Fremdschlüssel keine sprechenden Schlüsselwerte (z.B. Abkürzungen) enthalten sondern künstliche Schlüsselwerte (z.B. automatisch generierte Nummern).

Hier bietet PHPMYAdmin Unterstützung:

- PHPMYAdmin kann ein Menü erlaubter Fremdschlüsselwerte erzeugen.
- In diesem Menü können außer den (künstlichen) Primärschlüsselwerten (z.B. a_id) auch noch die Werte eines weiteren Feldes (z.B. bezeichnung etc.) angezeigt werden.

PFLEGE VON REFERENZEN IN TABELLEN (2)

Zur Unterstützung der Pflege von Referenzen sollte man in PHPMYAdmin

- die referenzielle Integrität einrichten (s.o.)
- in der Beziehungsübersicht der referenzierten Tabelle (Struktur – Beziehungsübersicht – ganz unten) die anzuzeigende Spalte auswählen (Speichern nicht vergessen!).

Anzuzeigende Spalte auswählen:

Dann wird beim Bearbeiten des Fremdschlüssels zusätzlich auch der Wert der anzuzeigenden Spalte (meist ein Name oder eine Bezeichnung) angezeigt.

- Einkauf - 1
 - Produktion - 2
 - Vertrieb - 4
-
- 1 - Einkauf
 - 2 - Produktion
 - 4 - Vertrieb

a_id int(10) unsigned

- **Structured Query Language (SQL):** in relationalen Datenbanksystemen verwendete formale Sprache
 - ⇒ zur Datendefinition (Definition von Tabellen)
 - ⇒ und zur Datenmanipulation (Definition von Abfragen)
- Hier wird nur auf die **Datenmanipulation** mit SQL (SQL als **Abfragesprache**) eingegangen, da man dies für die Entwicklung von datenbankbasierten Webapplikationen braucht.
- Die Datendefinition (Definition von Tabellen) erfolgt in der Regel nicht zur Laufzeit der Webapplikation und kann deshalb ohne SQL-Kenntnisse vorab mit einem Tool wie PHPmyAdmin vorgenommen werden.

ARTEN VON ABFRAGEN

Es lassen sich folgende Arten von Abfragen unterscheiden:

- **Auswahlabfragen (SQL-Befehl SELECT):**
 - ⇒ Daten aus ausgewählten Feldern einer oder mehrerer Tabellen werden nach bestimmten Kriterien selektiert.
 - ⇒ Das Ergebnis ist wieder eine Tabelle, die aber nur vorübergehende Existenz besitzt.
- **Aktionsabfragen:**

Datenbankinhalte werden nach bestimmten Kriterien verändert. Wir unterscheiden:

 - ⇒ **Aktualisierungsabfragen (SQL-Befehl UPDATE):**
Ändern von Datenwerten vorhandener Datensätze
 - ⇒ **Anfügeabfragen (SQL-Befehl INSERT):**
Anfügen von neuen Datensätzen an vorhandene Tabellen (oder Einfügen, was dasselbe ist)
 - ⇒ **Löschabfragen (SQL-Befehl DELETE):**
Löschen von Datensätzen in Tabellen

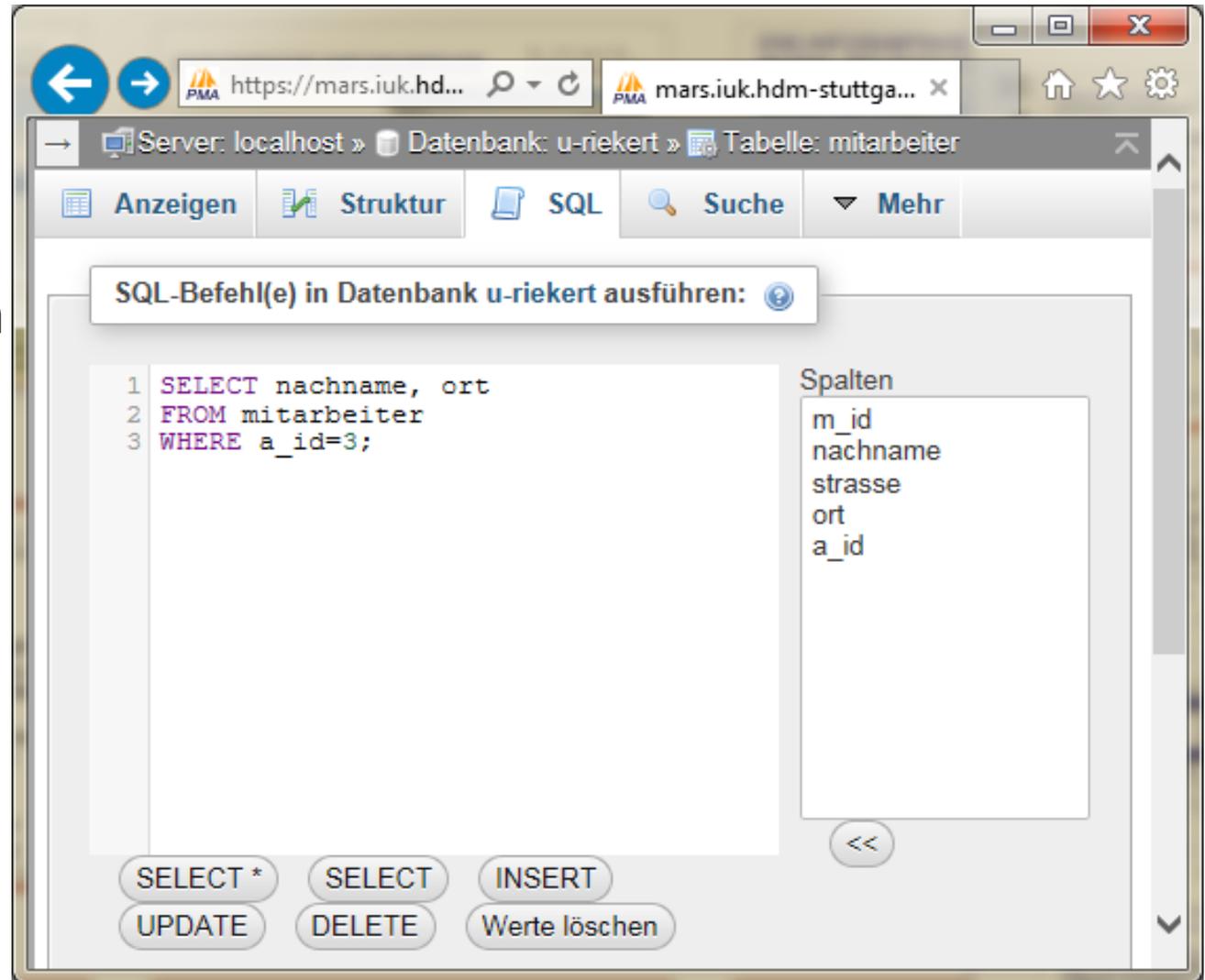
Unterschiedliche SQL-Anweisungen ermöglichen verschiedene Arten von Abfragen zur Datenmanipulation:

SQL-Anweisung	Zweck
SELECT	Auswahl- oder Selektionsabfragen Abrufen von Daten
UPDATE	Aktualisierungsabfragen Ändern von Daten
INSERT	Anfügeabfragen Eintragen von neuen Daten
DELETE	Löschabfragen Löschen von Daten
SELECT ... INTO	Tabellenerstellungsabfragen

SQL-ABFRAGEN IN PHPMYADMIN

SQL-Abfragen können in der SQL-Ansicht von PHPmyAdmin eingegeben und getestet werden.

Dafür stehen verschiedene Edit-Funktionen zur Verfügung.



ERGEBNIS EINER SQL-ABFRAGE

The screenshot shows a web browser window with the URL `https://mars.iuk.hdm-stuttgart.de`. The browser's address bar and tabs are visible. Below the browser, the phpMyAdmin interface is shown, displaying the following information:

- Server: localhost » Datenbank: u-riekert » Tabelle: mitarbeiter
- Navigation tabs: Anzeigen, Struktur, SQL, Suche, Einfügen, Mehr
- SQL query: `SELECT nachname, ort FROM mitarbeiter WHERE a_id=3`
- Buttons: Messen [Inline] [Bearbeiten] [SQL erklären] [PHP-Code erzeugen] [Aktualisieren]
- Controls: Anzahl der Datensätze: 25 (dropdown), Zeilen filtern: Diese Tabelle durchsuch
- Sortierung: Nach Schlüssel sortieren: keine (dropdown)
- Options: + Optionen
- Result table:

nachname	ort
Maier	Neu-Ulm
Weber	Stuttgart

AUSWAHLABFRAGEN IN SQL

- Auswahlabfragen in SQL werden mit SELECT-Anweisungen festgelegt.
- Es gibt verschiedene Arten von Auswahlabfragen, z.B.:
 - ⇒ **Projektion**: Auswahl von Tabellenspalten
 - ⇒ **Selektion**: Auswahl von Tabellenzeilen
 - ⇒ **Joins** (Verbünde): Vereinigung der Spalten zweier Tabellen unter bestimmten Bedingungen
 - ⇒ **Union** (Vereinigung): Vereinigung der Zeilen zweier Tabellen
 - ⇒ Diese Arten können auch kombiniert werden.
- Es gibt verschiedene Abfrageoptionen:
 - ⇒ **Sortieren, Gruppieren** von Datensätzen
 - ⇒ **Umbenennen** von Feldern

PROJEKTION

SELECT nachname, Ort
FROM mitarbeiter;



Abfrage

Ergebnis

The screenshot shows a web browser window with a database management interface. The address bar shows 'https://mars.iuk.h...'. The page title is 'Server: localhost » Datenbank: u-riekert » Tabelle: mitarbeiter'. The interface includes buttons for 'Anzeigen', 'Struktur', 'SQL', 'Suche', and 'Mehr'. The SQL command area contains the following query:

```
1 SELECT nachname, ort
2 FROM mitarbeiter
```

To the right of the query, a list of columns is shown:

Spalten
m_id
nachname
strasse
ort
a_id

Below the query, a table of results is displayed:

nachname	ort
Maier	Neu-Ulm
Huber	Stuttgart
Weber	Stuttgart

ABFRAGEOPTIONEN BEI DER PROJEKTION

- Alle Spalten auswählen:
`SELECT * FROM mitarbeiter;`

m_id	nachname	strasse	ort	a_id
1	Maier	Bahnhofstr. 3	Neu-Ulm	3
2	Huber	Türlestr. 32	Stuttgart	2
3	Weber	Nobelstr. 10	Stuttgart	3

- Nur unterschiedliche Datensätze:
`SELECT DISTINCT ort FROM mitarbeiter;`

ort
Neu-Ulm
Stuttgart

- Datensätze nach Spalte sortieren
`SELECT * FROM mitarbeiter
ORDER BY nachname;`

m_id	nachname	strasse	ort	a_id
2	Huber	Türlestr. 32	Stuttgart	2
1	Maier	Bahnhofstr. 3	Neu-Ulm	3
3	Weber	Nobelstr. 10	Stuttgart	3

Datensätze lassen sich mit der Option `ORDER BY` numerisch und alphabetisch sortieren. Absteigend sortiert wird durch den Zusatz `DESC`.

`SELECT * FROM mitarbeiter ORDER BY gehalt DESC, nachname`

Die Mitarbeiter werden nach Gehalt absteigend sortiert, innerhalb einer Gehaltsgruppe alphabetisch nach dem Namen.

(Vorausgesetzt ist, dass die Tabelle `mitarbeiter` ein Feld namens `gehalt` besitzt.)

SELEKTION



```
SELECT * FROM mitarbeiter WHERE ort = "Stuttgart";
```

Abfrage

Ergebnis

m_id	nachname	strasse	ort	a_id
1				
2	Huber	Türlenstr. 32	Stuttgart	2
3	Weber	Nobelstr. 10	Stuttgart	3

SELEKTIONSBEDINGUNGEN IN DER „WHERE“-KLAUSEL

- Alle Mitarbeiter aus Stuttgart
`SELECT * FROM mitarbeiter WHERE ort="Stuttgart";`
- Alle Mitarbeiter, die mehr als 5000 verdienen
`SELECT * FROM mitarbeiter WHERE gehalt > 5000;`
- Alle, deren Name alphabetisch sortiert nach „Maier“ kommt:
`SELECT * FROM mitarbeiter WHERE nachname > "Maier";`
- Bedingungen lassen sich auch kombinieren:
... `WHERE gehalt > 5000 AND NOT ort="Stuttgart";`

= gleich

<> ungleich

AND und

< kleiner

> größer

OR oder

<= kleiner oder gleich

>= größer oder gleich

NOT nicht

- Alle, deren Ort mit „Neu“ beginnt:

```
SELECT * FROM mitarbeiter  
WHERE ort LIKE "Neu%";
```

m_id	nachname	strasse	ort	a_id
1	Maier	Bahnhofstr. 3	Neu-Ulm	3
4	Hund	Schwarzwaldstr. 4	Neustadt	1

- Alle, deren Nachname mit „ber“ endet:

```
SELECT * FROM mitarbeiter  
WHERE nachname LIKE "%ber";
```

m_id	nachname	strasse	ort	a_id
2	Huber	Türlenstr. 32	Stuttgart	2
3	Weber	Nobelstr. 10	Stuttgart	3

AKTIONSABFRAGEN

Aktionsabfragen verändern Datenbankinhalte (im Gegensatz zu Auswahlabfragen)

Anfügeabfragen:

```
INSERT INTO mitarbeiter (name, ort, strasse)  
VALUES („Hacker", "Stuttgart", „Pfaffenwaldring");
```

Aktualisierungsabfragen:

```
UPDATE studierende SET hochschule = "HdM"  
WHERE hochschule = "HBI";
```

Löschabfragen:

```
DELETE FROM studierende WHERE jahrgang = 2005;
```

Tabellenerstellungsabfrage:

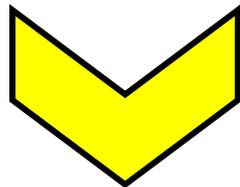
```
SELECT ((irgendeine Abfrage)) INTO tabellenname;
```

JOIN

			Blue wavy
			Green checkered
			Red brick
			Pink vertical

Red brick			
Blue wavy			
Red brick			
Green checkered			

Ausgangstabellen



Bedingung

			Red brick	Red brick			
			Blue wavy	Blue wavy			
			Red brick	Red brick			
			Green checkered	Green checkered			

Ergebnis des Join

Der Join verbindet zwei einzelne Tabellen zu einer großen Tabelle unter Berücksichtigung bestimmter Bedingungen. Beim „Equivalent Join“ oder „Equi-Join“ werden Datensätze zusammengefügt, bei denen die Werte ausgewählter Felder miteinander übereinstimmen.

INNER UND OUTER JOINS AUSGANGSTABELLEN

Ausgangstabellen

mitarbeiter

m_id	name	strasse
1	Maier	Bahnhofstr. 3
2	Huber	Königstr. 2
3	Schaufler	Marktplatz 6
4	Schreiber	Schloßstr. 20
5	König	Schillerstr. 29
6	Kerner	Kelterstr. 51
7	Einstein	Planckstr. 6
8	Walz	Industriestr. 4
9	Braun	Waldstr. 4
10	Filzer	Kohlstr. 45
11	Graf	Goethestr. 9
12	Kaiser	Pfarrstr. 40
13	Hacker	Zusestr. 200

abteilung

a_id	bezeichnung	abteilungsleiter_id
1	Einkauf	12
2	Produktion	5
3	Entwicklung	11
4	EDV	13
5	E-Commerce	

ort	abteilung	abteilungsleiter_id
Boblingen		9
Ludwigsburg		2
Esslingen		3
Karlsruhe		2
Mannheim		1
Leonberg		2
Filderstadt		3
Sindelfingen		3
Stuttgart		1
München		4

INNER UND OUTER JOINS

Ziel: Wir möchten die Tabellen `mitarbeiter` und `abteilung` durch einen Equivalent Join über die `abteilungsleiter_id` verknüpfen.

Probleme:

- Es gibt Mitarbeiter, die keine Abteilungen leiten. Sollen diese im Ergebnis mit aufgeführt werden?
- Es gibt Abteilungen, für die der Abteilungsleiter nicht festgelegt wurde. Gehören die auch zum Ergebnis?

Je nach der Antwort auf diese Frage muss man einen Inner oder einen Outer Join verwenden.

- Der „**Left Outer Join**“ (oder kurz: „**Left Join**“) schließt alle Datensätze der linken (ersten) Tabelle ein, der „**Right Outer Join**“ (oder kurz: „**Right Join**“) alle der rechten.
- Der „**Inner Join**“ enthält nur die Datensätze, bei denen in beiden Tabellen die betreffenden Felder existieren.

INNER UND OUTER JOINS

ERGEBNISSE

Inner Join

name	bezeichnung
Produktion	König
Entwicklung	Graf
Einkauf	Kaiser
EDV	Hacker

Left (Outer) Join

name	bezeichnung
Produktion	König
Entwicklung	Graf
Einkauf	Kaiser
EDV	Hacker
E-Commerce	

Right (Outer) Join

name	bezeichnung
	Maier
	Huber
	Schaufler
	Schreiber
Produktion	König
	Kerner
	Einstein
	Walz
	Braun
	Filzer
Entwicklung	Graf
Einkauf	Kaiser
EDV	Hacker

INNER JOIN

```
SELECT abteilung.bezeichnung, mitarbeiter.name  
FROM abteilung INNER JOIN mitarbeiter  
ON abteilung.m_id_abteilungsleiter = mitarbeiter.m_id;
```

Der „Inner Join“ enthält nur die Datensätze, bei denen in beiden Tabellen die betreffenden Felder existieren.

name	bezeichnung
Produktion	König
Entwicklung	Graf
Einkauf	Kaiser
EDV	Hacker

LEFT JOIN

```
SELECT abteilung.bezeichnung, mitarbeiter.name  
FROM abteilung LEFT JOIN mitarbeiter  
ON abteilung.m_id_abteilungsleiter = mitarbeiter.m_id;
```

Der „Left Outer Join“
(oder kurz: „Left Join“) schließt
alle Datensätze der linken
(ersten) Tabelle ein
und von der rechten Tabelle
alle passenden.

name	bezeichnung
Produktion	König
Entwicklung	Graf
Einkauf	Kaiser
EDV	Hacker
E-Commerce	

RIGHT JOIN

```
SELECT abteilung.bezeichnung, mitarbeiter.name
FROM abteilung RIGHT JOIN mitarbeiter
ON abteilung.m_id_abteilungsleiter = mitarbeiter.m_id;
```

Der „Right Outer Join“ (oder kurz: „Right Join“) schließt alle Datensätze der rechten (zweiten) Tabelle ein und von der linken Tabelle alle passenden.

name	bezeichnung
	Maier
	Huber
	Schaufler
	Schreiber
Produktion	König
	Kerner
	Einstein
	Walz
	Braun
	Filzer
Entwicklung	Graf
Einkauf	Kaiser
EDV	Hacker

- PHP: Skriptsprache zur serverseitigen Erzeugung von dynamischen Webseiten
- PHP ist eingebettet in HTML
 - ⇒ Dateinamenserweiterung .php statt .html
 - ⇒ Die festen Anteile einer Webseite werden weiterhin durch die bekannten HTML-Befehle erzeugt
 - ⇒ Die variablen Anteile werden durch die Sprachelemente der PHP-Skriptsprache erzeugt.
 - ⇒ Kennzeichnung der PHP-Skriptelemente durch besondere Tags
- PHP-Syntax ähnlich C/C++, Perl, Java oder Javascript
- Zentrale Website: <http://www.php.net>

S. Münz: *Webseiten professionell erstellen*. 3. überarb. Aufl. Addison Wesley, 2008 (2. Aufl. als Download abrufbar:

<http://www.stefan-muenz.de/webseiten-professionell/professionelle-websites-auflage-2.pdf>)

PHP-Dokumentationsgruppe: *PHP-Handbuch*.

<http://php.net/manual/de/>

MySQL Documentation. <http://dev.mysql.com/doc/>

TAGS ZUR EINBETTUNG VON PHP-CODE IN HTML-CODE

- Mit einem Script-Tag:
 - ⇒ `<script language="php"> echo "Hallo";
</script>`
- XML-konforme Einbettung:
 - ⇒ `<?php echo "Hallo"; ?>`

Weniger zu empfehlen:

- SGML-konforme Einbettung
(erfordert Konfigurationseinstellung *short_open_tag*)
 - ⇒ `<? echo "Hallo"; ?>`
- Tags im Stil von Microsofts ASP
(erfordert Konfigurationseinstellung *asp_tags*)
 - ⇒ `<% echo "Hallo"; %>`
 - ⇒ `<%= $variable>` als Abkürzung für
`<% echo $variable %>`

EIN HELLO WORLD SCRIPT IN PHP

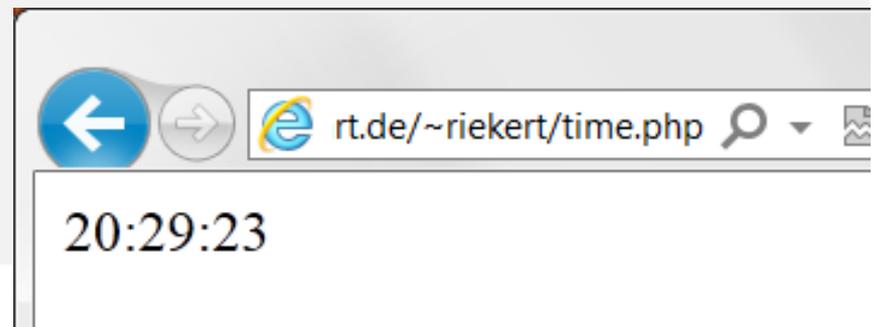
```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Beispiel</title>
  </head>
  <body>
    <?php
      echo "Hallo, ich bin ein PHP-Skript!";
    ?>
  </body>
</html>
```



DIE ERSTE „DYNAMISCHE“ WEBSEITE

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Uhrzeit</title>
  </head>

  <body>
    <?php
      echo date("H:i:s");
    ?>
  </body>
</html>
```



- Von einem Internetbrowser aus wird eine URL mit Dateinamensendung `.php` abgerufen.
- Der Webserver ist so konfiguriert, dass er angeforderte Dateien mit Endung `.php` an den PHP-Interpreter übergibt.
- Der PHP-Interpreter verarbeitet die PHP-Dateien, indem er
 - ⇒ den eingebetteten PHP-Code ausführt und
 - ⇒ aus der PHP-Datei eine HTML-Seite erzeugt.
- Die erzeugte HTML-Seite sieht aus wie die PHP-Datei, nur sind die Teile innerhalb der PHP-Tags durch Text ersetzt, der durch eingebettete PHP-Ausgabeansweisungen generiert wurde, z.B. durch den Befehl `echo`.
- Diese HTML-Seite wird an den Webserver übergeben, der diese dann an den Internetbrowser zur Anzeige überträgt.

Algorithmus: Arbeitsanleitung zum Lösen eines Problems oder einer Aufgabe, die so präzise formuliert ist, dass sie im Prinzip auch von einem Computer ausgeführt werden kann.

Programmablaufpläne (Flussdiagramme) und **Struktogramme** (Nassi-Shneidermann-Diagramme) dienen zur graphischen Darstellung von Algorithmen.

Programmiersprachen dienen zur Formulierung von Algorithmen.

Ein in einer Programmiersprache formulierter Algorithmus heißt **Programm**.

In Form von Programmen können Algorithmen durch einen **Computer** ausgeführt werden.

AUFGABE: DER QUADRATZAHLTTEST

Aufgabe:

Es ist ein Programm zu schreiben, das testet, ob eine vom Benutzer eingegebene Zahl eine Quadratzahl ist.

Beispiel:

Benutzer:	9
System:	Quadratzahl
Benutzer:	3
System:	keine Quadratzahl

EIN ALGORITHMUS ZUR LÖSUNG DER AUFGABE

Algorithmus

Parameter n übernehmen.

Der Reihe nach alle
Quadratzahlen q bilden:
 $q = 0, 1, 4$, usw.

Weitermachen, solange q
< n (kleiner als n) ist.
Aufhören, wenn das nicht
mehr der Fall ist.

Wenn nun $n = q$ ist,
war n eine Quadratzahl,
sonst nicht.

Beispiel 1

$$n = 9$$

$$q = 0*0 = 0$$

$$q = 1*1 = 1$$

$$q = 2*2 = 4$$

$$q = 3*3 = 9$$

Jetzt gilt
 $q < n$ nicht
mehr

$n = q$?
Ja! \Rightarrow n ist
Quadratzahl

Beispiel 2

$$n = 3$$

$$q = 0*0 = 0$$

$$q = 1*1 = 1$$

$$q = 2*2 = 4$$

Jetzt gilt
 $q < n$ nicht
mehr

$n = q$?
Nein! \Rightarrow n ist
keine Q'zahl.

UMSETZUNG DES ALGORITHMUS IN EIN PROGRAMM

Algorithmus

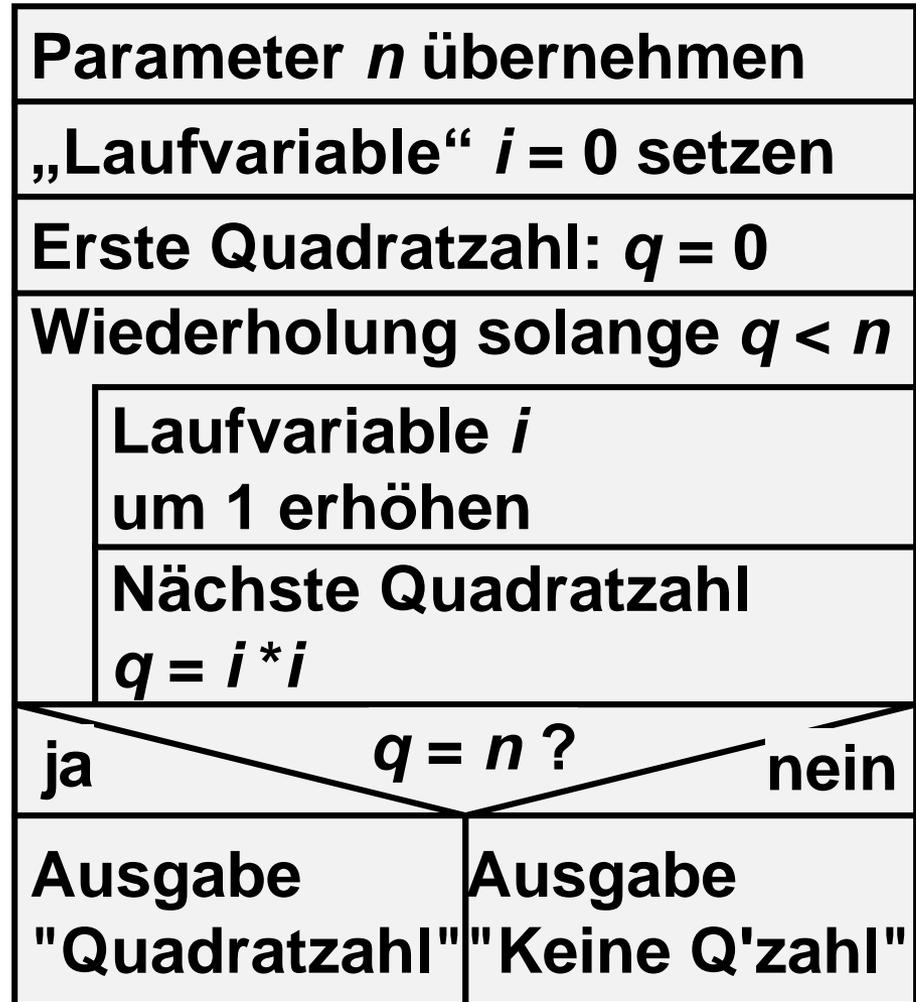
Parameter n übernehmen.

Der Reihe nach alle
Quadratzahlen q bilden:
 $q = 0, 1, 4, \text{ usw.}$

Weitermachen, solange q
< n (kleiner als n) ist.
Aufhören, wenn das nicht
mehr der Fall ist.

Wenn nun $n = q$ ist,
war n eine Quadratzahl,
sonst nicht.

Struktogramm

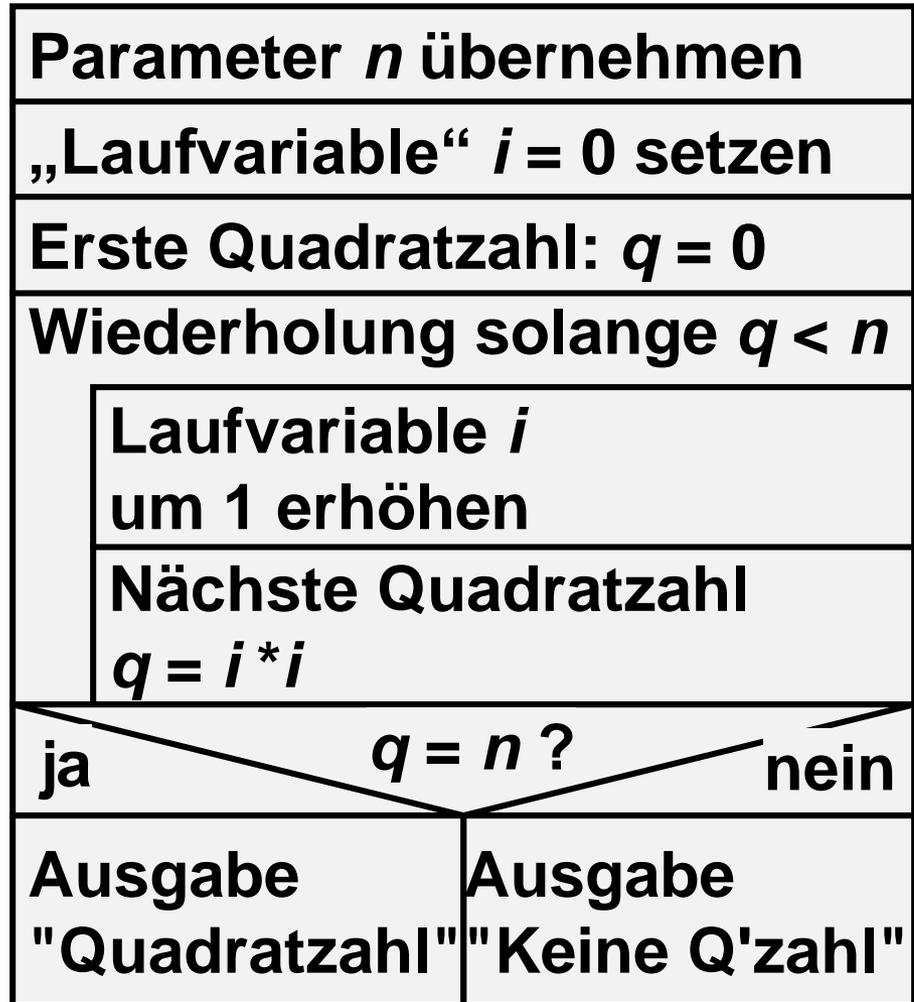


Struktogramm

 Sequenz
von Anweisungen

 Schleife

 Bedingte
Anweisung



UMSETZUNG DES STRUKTOGRAMMS IN EIN PHP-PROGRAMM

PHP-Programm

```
$n = $_GET["n"];
```

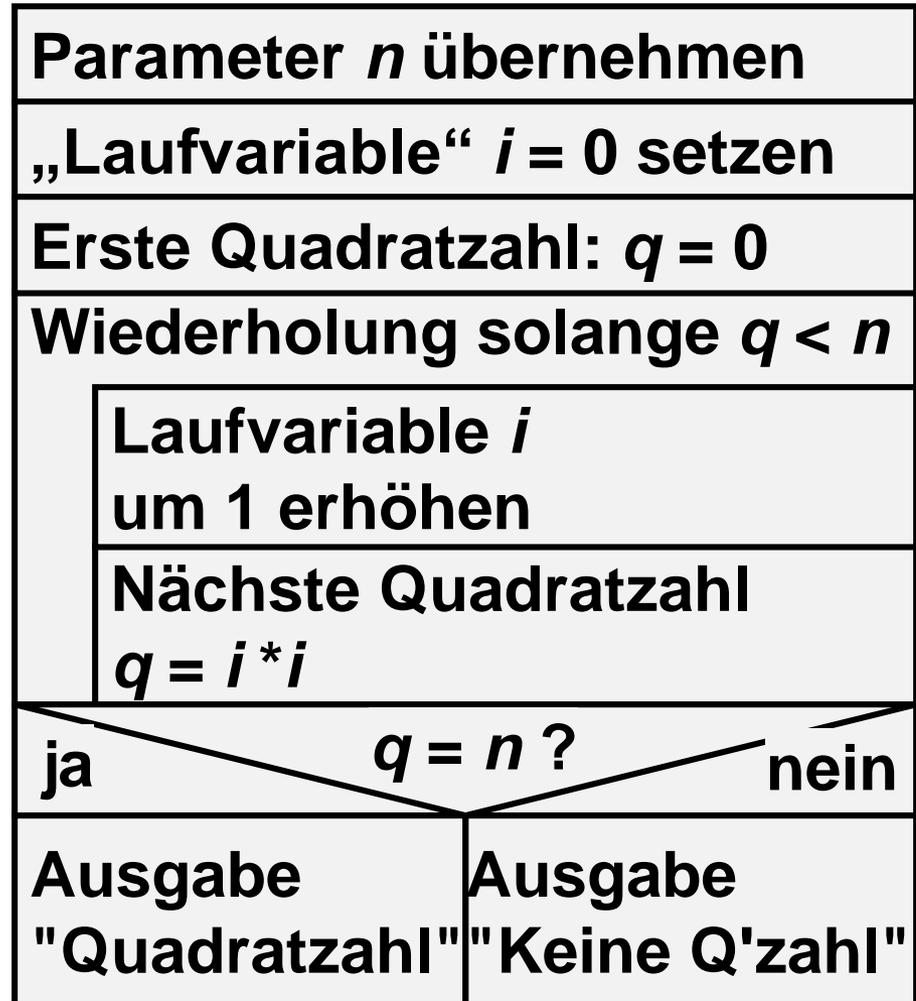
```
$i = 0;
```

```
$q = 0;
```

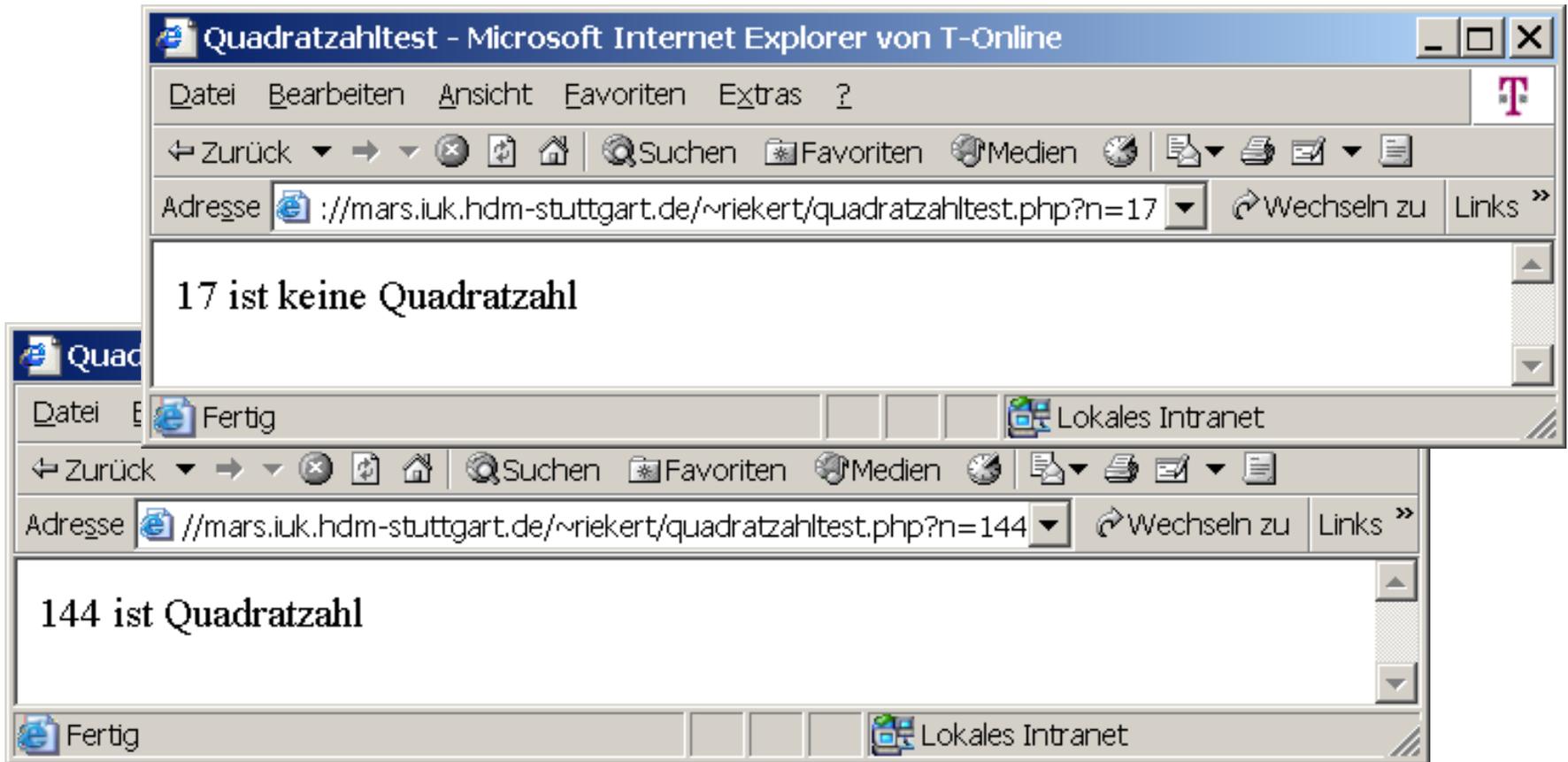
```
while ($q < $n) {  
    $i = $i + 1;  
    $q = $i * $i;  
};
```

```
if ($q == $n) {  
    echo "Quadratzahl";  
}  
else {  
    echo "Keine Q'zahl";  
};
```

Struktogramm



PARAMETERÜBERGABE VIA URL



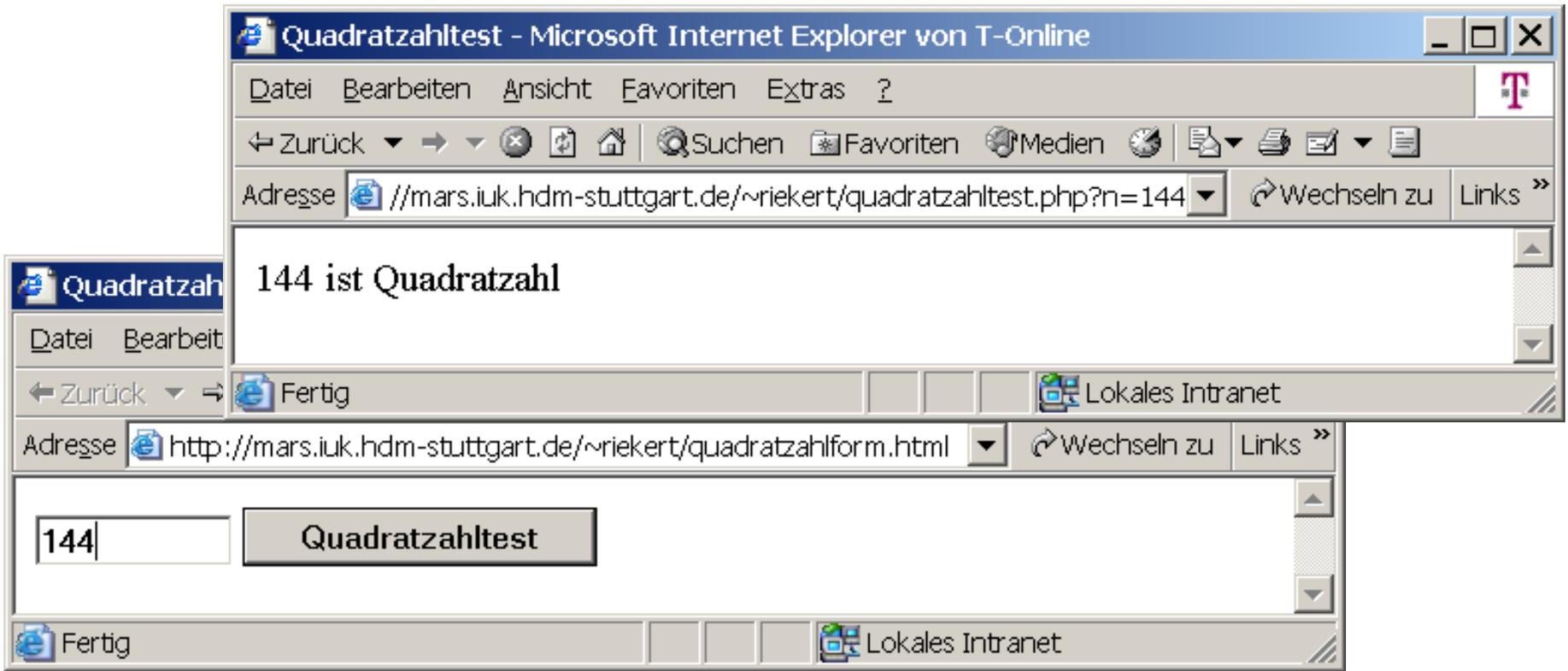
Übergabe des Parameters n an das Skript über die URL:
Aufruf des Skripts über sogenannte GET-Methode.
Zugriff auf den Parameter im PHP-Skript durch **`$_GET["n"]`**

FORMULAR ZUR PARAMETERÜBERGABE

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8" />
  <title>Quadratzahltest</title>
</head>
<body>
  <form action="quadratzahltest.php" method="get">
    <input type="text" name="n" size=10 />
    <input type="submit" value="Quadratzahltest" />
  </form>
</body>
</html>
```



AUFRUF EINES SKRIPTS VIA FORMULAR



Die Option `method="get"` bewirkt eine Parameterübergabe in der URL, doch das geht nur bei kurzen Parametern (< 128 Zeichen). Die Option `method="post"` hat keine solche Beschränkung, dann werden aber die Parameter nicht mehr in der URL sichtbar und der Zugriff im PHP-Skript erfolgt dann durch `$_POST["n"]`

ADDITION: FORMULAR MIT MEHREREN PARAMETERN

```
<!doctype html>
```

```
<html>
```

```
<head>
```

```
  <meta charset="utf-8" />
```

```
  <title>Addition</title>
```

```
</head>
```

```
<body>
```

```
  <form action="add.php" method="get">
```

```
    <input type="text" name="a" size="5" /> +
```

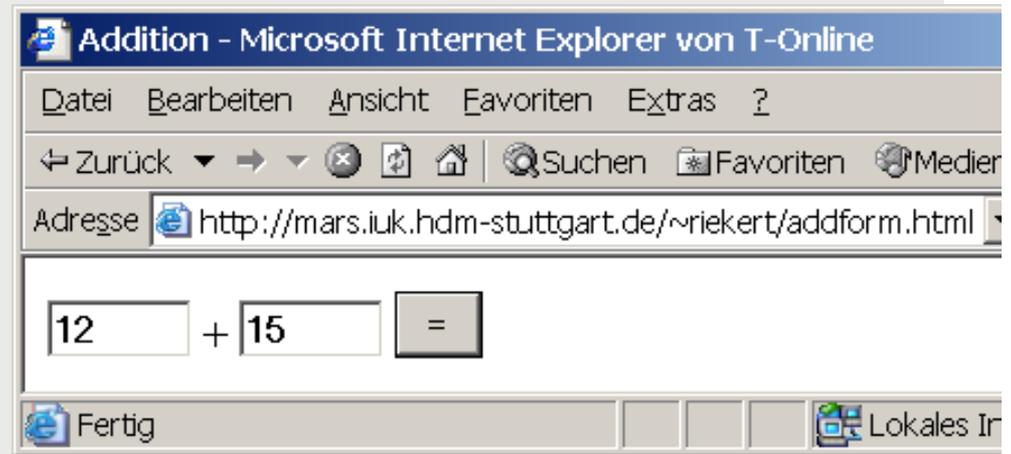
```
    <input type="text" name="b" size="5" />
```

```
    <input type="submit" value=" = " />
```

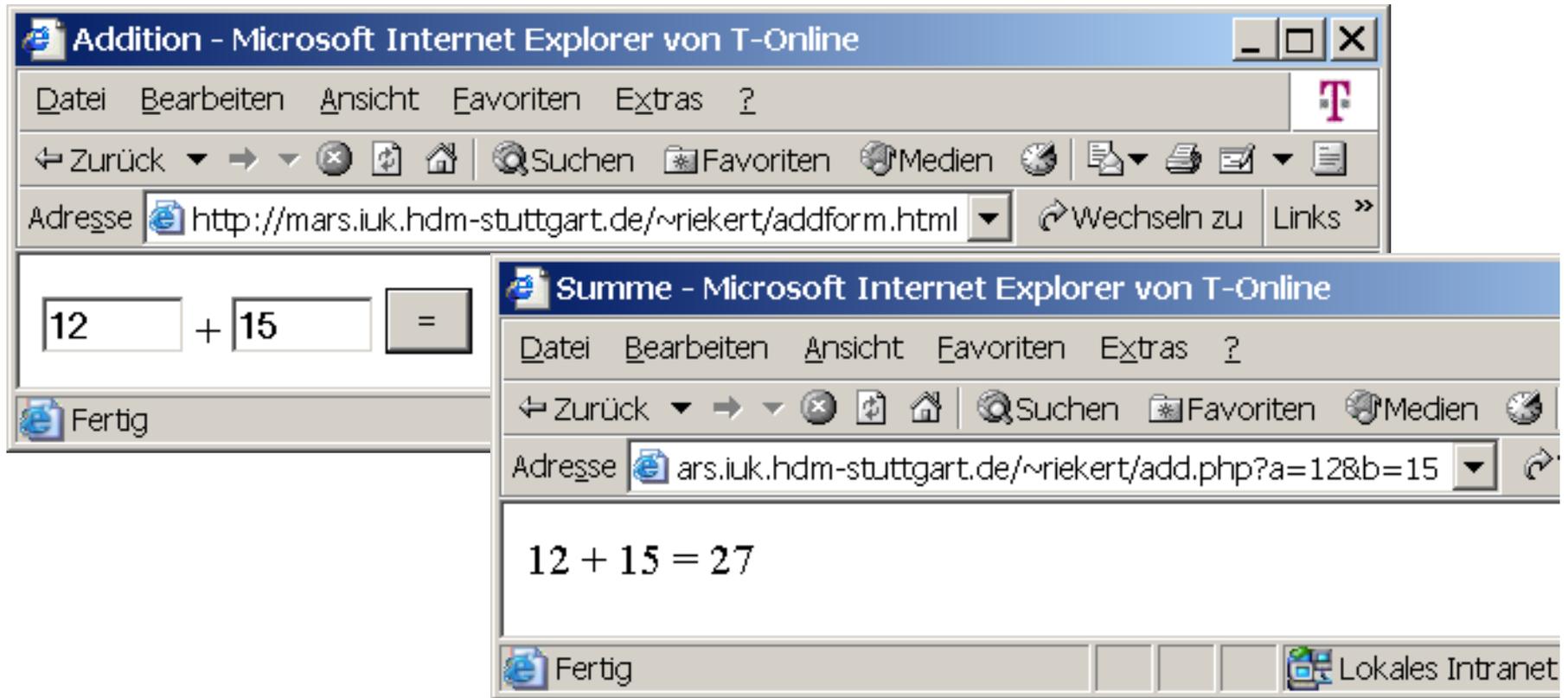
```
  </form>
```

```
</body>
```

```
</html>
```



ADDITION: ADDITION ZWEIER ZAHLEN DURCH PHP-SKRIPT



Multiple Parameter werden mit dem &-Zeichen verkettet.
Übergabe in der URL durch die Get-Methode:

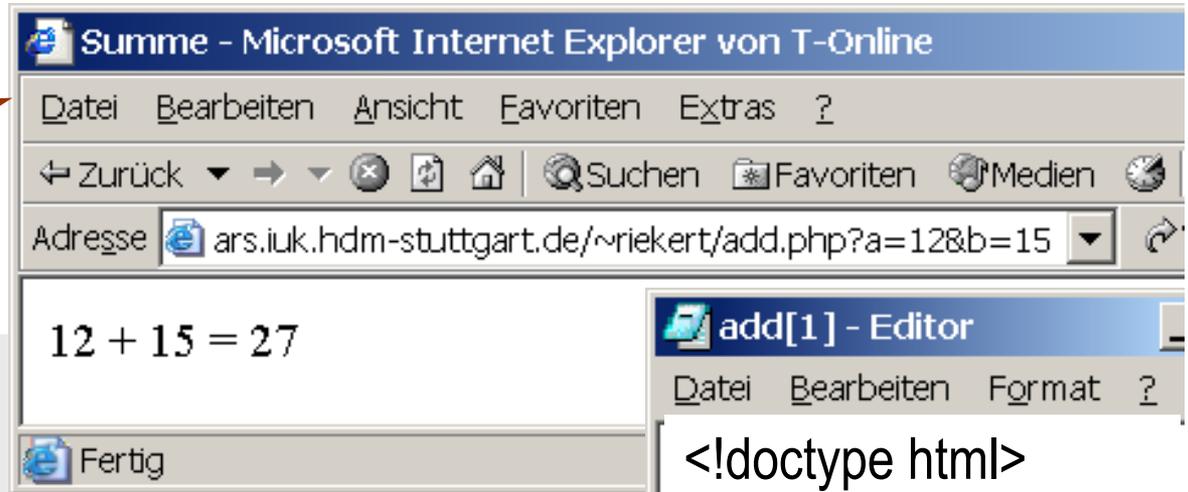
<http://domain/verzeichnis/skript.php?par1=wert1&par2=wert2> ...

ADDITION: DAS PHP-SKRIPT

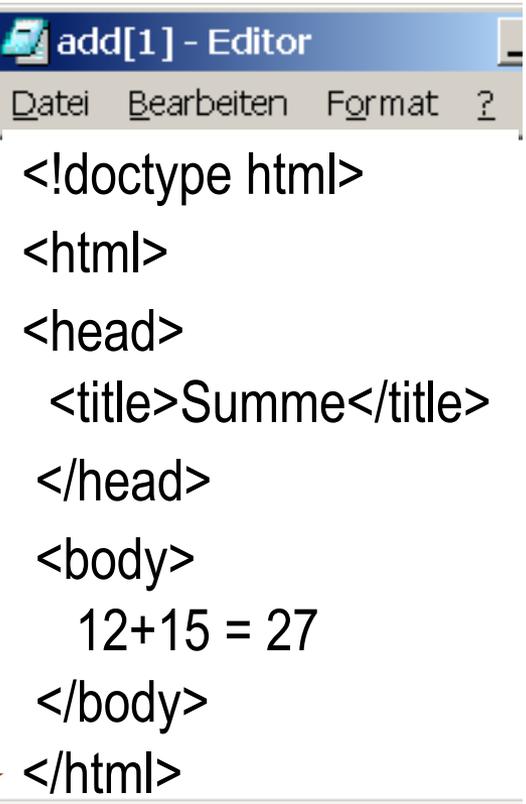
Ausgabe des Skripts

Das PHP-Skript

```
<!doctype html>
<html>
<head>
<title>Summe</title>
</head>
<body>
<?php
    $a=$_GET["a"]; $b=$_GET["b"];
    echo $a . " + " . $b . " = " . ($a+$b);
?>
</body>
</html>
```



Seitenquelltext der Ausgabe des Skripts



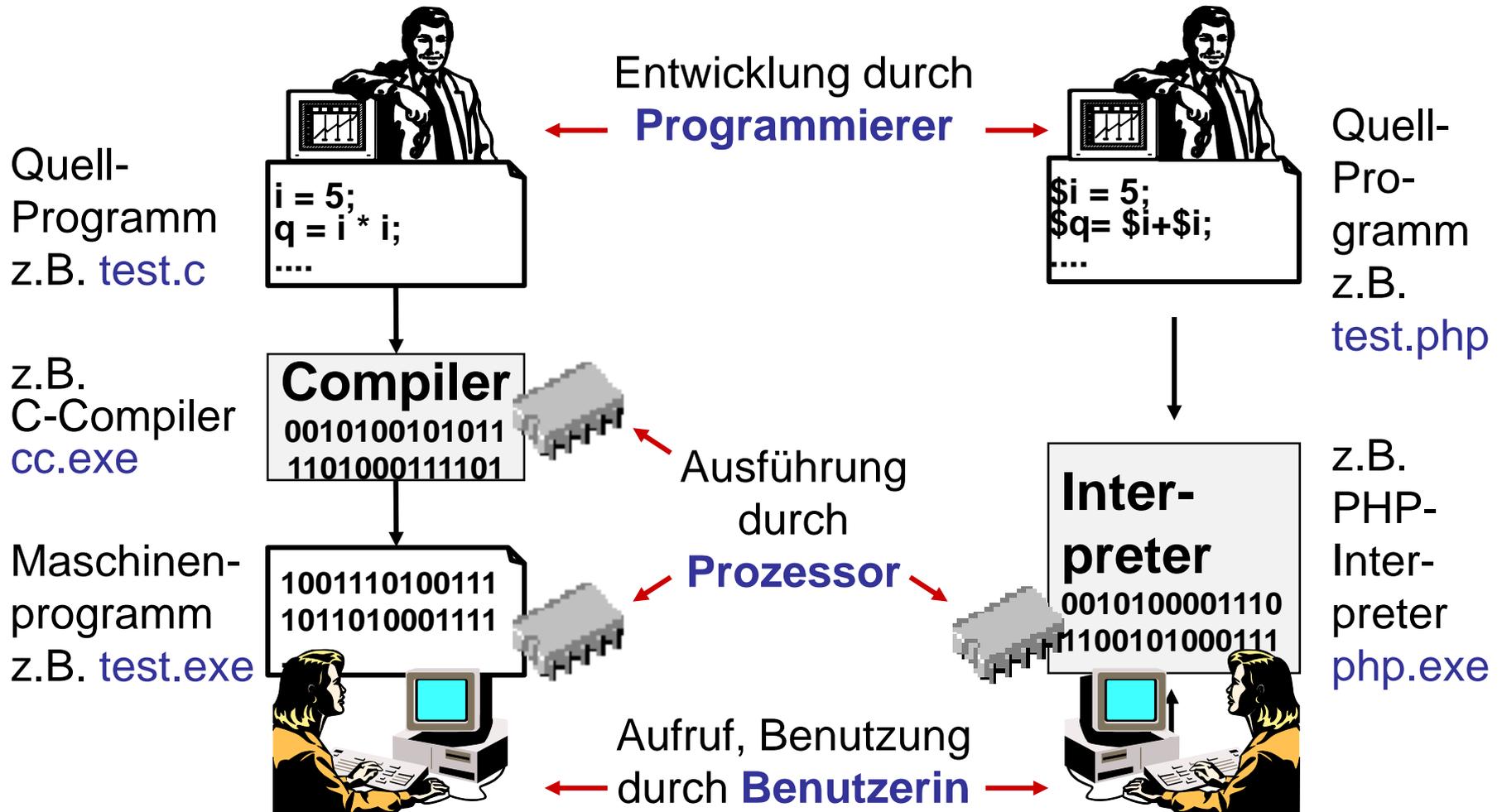
- Ein Prozessor (z.B. der Pentium-Prozessor) kann nur sogenannte **Maschinenprogramme** ausführen, diese bestehen aus einer Folge von Zahlencodes.
- Menschen schreiben Maschinenprogramme in einer sog. **Assemblersprache**, dabei sind die Zahlencodes durch Namen von Maschinenbefehlen ersetzt.
- Meist schreibt man Programme aber in **höheren Programmiersprachen** (z.B. C, Java, PHP)
 - ⇒ Sie werden dann entweder mit einem **Compiler** in Maschinenprogramme übersetzt und anschließend durch den Prozessor ausgeführt
 - ⇒ oder durch einen sog. **Interpreter** ausgeführt.

Compiler und Interpreter sind selbst Programme.

COMPILERSPRACHEN UND INTERPRETERSPRACHEN

Compilersprachen

Interpretersprachen



- Der Compiler wird nur bei der Programmentwicklung gebraucht, im Betrieb läuft das übersetzte Programm ab.
- Deshalb hat der Compiler Zeit für aufwendige Programmüberprüfungen und Optimierungen.
- Compiler überprüfen Programme hinsichtlich „Vokabular“ und „Grammatik“, so dass viele Programmierfehler bereits bei der Compilierung entdeckt werden können.
- Durch Compiler übersetzte Programme sind Maschinenprogramme, die in der Regel sehr schnell und effizient ablaufen.
- Die Programmentwicklung mit Compilersprachen ist etwas mühevoll, da ein Programm nach jeder Änderung neu compiliert werden muss.
- Wichtigstes Beispiel für Compilersprachen: C/C++

- Interpreterprogramme funktionieren nicht für sich alleine, sie benötigen zur Ausführung einen Interpreter.
- Da der Interpreter zur Laufzeit des Programms aktiv ist, hat er wenig Zeit für aufwendige Prüfungen. Fehlerhafte Programme „stürzen“ oft mit einer kurzen Meldung „ab“.
- Es gibt jedoch Programmentwicklungsumgebungen mit Editoren, die Syntaxüberprüfungen vornehmen.
- Interpretierte Programme sind deutlich langsamer als compilierte, was mit den heutigen schnellen Computern allerdings kein großes Problem mehr darstellt.
- Die Programmentwicklung ist erleichtert, da Programme nach Änderungen sofort wieder gestartet werden können.
- Beispiele für Interpretersprachen: Visual Basic for Applications (VBA), Javascript, Perl, **PHP**

ANWEISUNGEN

PHP-Scripts werden aus Anweisungen gebildet, die mit „;“ verkettet werden.

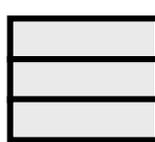
Elementare Anweisungen sind:

 **Zuweisungen**, z.B.: `$i = $i + 1`

 **Befehle**, z.B.: `echo $i`

 **Funktionsaufrufe**, z.B.: `TitelAusgeben("Hi")`

Zusammengesetzte Anweisungen (Kontrollstrukturen) sind:

 **Sequenzen (Blöcke)** von Anweisungen, verkettet mit „;“, geklammert durch „{ }“
z.B.: `{ $i = $i+1; $s = $s+$i; }`

 **Bedingte Anweisungen**, z.B.: `if`-Anweisung
`if ($x>0) { $y=$x; } else { $y=-$x; }`

 **Schleifen**, z.B.: `while`-Anweisung
`while ($i<10) { $erg = $erg+$i; $i=$i+1; }`

Bedingte Anweisung:

```
if (Bedingung)  
    {Anweisung; Anweisung; ...}  
else  
    {Anweisung; Anweisung; ...}
```

Bedingung	
ja	nein
Anw.	Anw.
Anw.	Anw.

Der Else-Zweig kann auch entfallen, sofern er nicht gebraucht wird.

Beispiele:

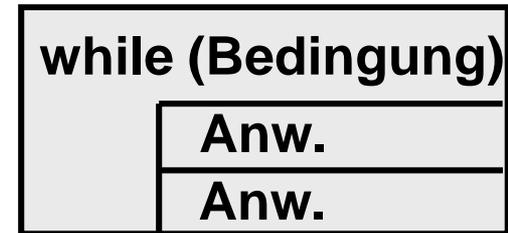
```
if ($x>0) {$y=$x;} else {$y=-$x;}
```

```
if ($x<0) {$x=-$x;}
```

SCHLEIFEN

Schleife:

```
while (Bedingung)  
    {Anweisung; Anweisung; ...}
```



Beispiel:

```
$ergebnis = 0;  
$i = 1;  
while ($i < 10) {  
    $ergebnis = $ergebnis + $i;  
    $i = $i + 1;  
}
```

- In Variablen können einfache Werte gespeichert werden.
- Auch zusammengesetzte Daten wie Arrays und Hashes (assoziative Arrays) oder Objekte werden in normalen Variablen gespeichert.
- In PHP sind die Werte typisiert und nicht die Variablen (anders als in Perl oder Java)
- Variablen müssen nicht deklariert werden; die Namen aller Variablen werden mit dem Dollarzeichen (\$) markiert, damit sie der Interpreter automatisch erkennt.

```
$hausnummer = 32;  
$strasse = "Wolframstrasse";  
$woche = array("mo", "di", "mi", "do", "fr", "sa");
```

In PHP gibt es folgende Typen von Werten:

- **Skalartypen** (einfache Datentypen)

- ⇒ **int** (ganze Zahlen)

- ⇒ **double** (Dezimalbrüche)

- ⇒ **string** (Texte)

- ⇒ **boolean** (Wahrheitswerte)

```
$i = 100;  
$pi = 3.1415;  
$c = 2.998E8;  
$fh = "HdM";  
$b = ($i < $n);  
$f = FALSE;
```

- **Strukturierte Typen** (zusammengesetzte Datentypen)

- ⇒ **Array** (Listen von Werten)

Zugriff über Elementnummer
oder in „assoziativem Array“
über textuellen Zugriffsschlüssel

```
$torwart  
= $spieler[1];  
$tel["Riekert"]  
= "0731-36456"
```

- ⇒ **Object** (Objekte für die objektorient. Programmierung)

STRINGLITERALE (1)

Stringliterale dienen zur Angabe fester Strings (Zeichenketten). Sie können mit "doppelten" oder 'einfachen' Anführungszeichen dargestellt werden.

In "doppelten" Anführungszeichen lassen sich Sonderzeichen mit Hilfe des Gegenschrägstriches (Backslash) angeben.

- \"** Anführungszeichen
- ** Gegenschrägstrich (Backslash)
- \n** Zeilenvorschub (Newline)
- \\$** Dollarzeichen
- \033** Sonderzeichen mit Oktalcode 33 (ESC)
- \x7F** Sonderzeichen mit Hexcode 7F (DEL)

```
echo "Der \"zwanghafte\nProgrammierer\"";  
⇒ Der "zwanghafte  
Programmierer"
```

STRINGLITERALE (2)

- In Stringliteralen, die durch "doppelte" Anführungszeichen eingeschlossen sind, können auch Werte von Variablen eingefügt werden (In Perl nennt man das „Interpolation“):

```
$faktor1 = 3; $faktor2 = 4;  
echo "Berechne $faktor1 + $faktor2 !";  
Ergebnis: Berechne 3 + 4 !
```

- In Stringliteralen, die durch 'einfache' Anführungszeichen eingeschlossen sind, werden alle Zeichen unverändert übernommen, mit Ausnahme des Gegenschrägstriches und des einfachen Anführungszeichen. Diese müssen folgendermaßen dargestellt werden:

\ ' **Anführungszeichen**

\\ **Gegenschrägstrich (Backslash)**

Zahlenwerte lassen sich mit Hilfe von Rechenoperatoren (Grundrechenarten: $+$ $-$ $*$ $/$, Divisionsrestbildung: $\%$, Potenzbildung: $**$) zu Ausdrücken kombinieren:

```
$umfang = 2 * ($laenge + $breite);  
$neunerrest = $wert % 9;  
$xquadrat = $x ** 2;
```

Strings lassen sich mit Hilfe des Verkettungsoperators `.` zusammenfügen:

```
$text = "Gesamtpreis = " . 2*3;  
$text .= " Euro"; // Anfüegen an die Variable  
echo $text;
```

⇒ Gesamtpreis = 6 Euro

- Arithmetische Operatoren verknüpfen Zahlen miteinander, Stringoperatoren verknüpfen Strings.
- Erforderlichenfalls wird automatisch gewandelt
 - ⇒ Für Stringoperatoren werden Zahlen in gleichlautende Strings gewandelt
 - ⇒ Für arithmetische Operatoren werden Strings automatisch in Zahlen gewandelt. Wenn diese mit Zahlen beginnen, so werden diese als Ergebnis genommen. Falls nicht, wird 0 als Wert angenommen!
- Oder man verwendet Typwandlungsoperatoren:
(int), (double), (string), (bool), (array), (object), z.B.
`echo (int) (5 / 2) ;`
⇒ 2

VERGLEICH VON WERTEN

Für Zahlen und Strings werden dieselben Vergleichsoperatoren verwendet:

```
if ($hausnummer == 34)
    {echo "Nebengebäude" }

if ($str == 'Wolframstrasse')
    {$hausmeister = "Rueber" }
```

Falls die Datentypen ungleich sind, werden diese automatisch gewandelt, und zwar in Richtung Zahlen! Ein String, der nicht mit Ziffern beginnt, wird dann als 0 interpretiert!

==	gleich
!=	nicht gleich
<	kleiner als
<=	kleiner oder gleich
>	größer als
>=	größer oder gleich

Achtung! Nicht verwechseln:

== Vergleich
= Zuweisung

DER DATENTYP BOOLEAN

- Bedingungen in IF- und WHILE-Anweisungen wie z.B. ($\$i < \n) sind ebenfalls Ausdrücke. Ihr Ergebnis ist vom Datentyp **boolean** (ein Wahrheitswert).
- Es gibt nur zwei Werte vom Typ **boolean**: **TRUE** (wahr = trifft zu) und **FALSE** (falsch = trifft nicht zu)
- Andere Datentypen wie **int**, **double**, **string** oder **array** lassen sich in den Typ **boolean** konvertieren, z.B. mit dem Operator (**bool**) oder automatisch in einer WHILE- oder IF-Bedingung:
 - ⇒ Dabei werden die Zahlen 0, 0.0, der String "0", der leere String "", der spezielle Typ **NULL** inkl. nicht gesetzter Variablen sowie ein Array ohne Elemente in den Wert **FALSE** konvertiert.
 - ⇒ Alle anderen Zahlen, Strings und Arrays werden in den Wert **TRUE** konvertiert

ARRAYS ALS WERTE

Arrays (Listen) sind ein besonderer Datentyp, der mehrere Werte aufnehmen kann:

- Arrays werden mit der Funktion **array()** angelegt.
- Für den Zugriff auf einzelne Elemente verwendet man das **\$**-Zeichen sowie einen numerischen Index in **[eckigen Klammern]**, der ab 0 zählt.

```
$woche = array('mo', 'di', 'mi', 'do', 'fr', 'sa');  
echo $woche[1];
```

⇒ di

```
$woche[6] = 'so';
```

ASSOZIATIVE ARRAYS (HASHS)

Die Indizes von Arrays müssen nicht notwendigerweise Zahlen sein. Man kann hierfür auch beliebige Strings verwenden.

- Dadurch sind auch „assoziative Arrays“ möglich (Dieser Datentyp wurde erstmals in der Programmiersprache Perl unter dem Namen „Hash“ eingeführt)

```
$translate = array('gut' => 'good', 'schlecht' => 'bad');  
echo "I feel " . $translate['schlecht'] . "\n";
```

⇒ I feel bad

```
$translate['sehr'] = 'very';  
echo "PHP is" . $translate['sehr'] . $translate['gut'] . "\n";
```

⇒ PHP is very good

\$_GET[] und **\$_POST[]** sind ebenfalls assoziative Arrays.

FUNKTIONEN (1)

Eine Funktion ist ein benannter Programmcodeblock, der eine bestimmte Aufgabe übernimmt. Er kann dabei mit einem Satz ihm übergebener Werte, den so genannten Parametern, arbeiten und eventuell einen einzelnen Wert zurückgeben. Eine Funktion kann an mehreren Stellen in einem PHP-Skript aufgerufen werden.

```
function zins ($prozent, $kapital)
{
    $ergebnis = $prozent * $kapital / 100.0;
    return $ergebnis;
}

$k0 = 10000.00;
$k1 = $k0 + zins (2.0 , $k0); // = 10200.00
$k2 = $k1 + zins (2.0 , $k1); // = 10404.00
```

Definition einer Funktion

```
function funktionsname ($par1, $par2, ...)
{
    Anweisungsblock;
    return Ausdruck;
}
```

Typischerweise geben Funktionen einen Wert zurück. Wird das Schlüsselwort **return** erreicht, wird die Kontrolle an die aufrufende Anweisung zurückgegeben und das Ergebnis des Ausdrucks hinter dem **return** wird als Wert der Funktion zurückgegeben.

Funktionen die keinen Wert per **return** zurückgeben, werden auch als Prozeduren bezeichnet.

PHP unterstützt die objektorientierte Programmierung

- Ein **Objekt** besitzt Eigenschaften (Daten) und Verhalten (Funktionen/Prozeduren).
 - ⇒ Die Daten werden als Attribute des Objekts bezeichnet.
 - ⇒ Die Funktionen/Prozeduren werden als Methoden des Objekts bezeichnet.
 - ⇒ Objekte werden genutzt durch Zugriff auf die Werte ihrer Attribute und Aufruf ihrer Methoden
- Eine **Klasse** fasst gleichartige Objekte, die so genannten Instanzen der Klasse zusammen.
 - ⇒ Die Klasse ist eine Art Schablone, die die Attribute und Methoden ihrer Instanzen festlegt.
 - ⇒ Klassen werden u.a. genutzt, um Instanzen zu bilden.

OBJEKTE: PROGRAMMBEISPIEL

```
class Konto {
    public $KtoNr;
    public $KtoInhaber;
    private $Ktostand = 0;
    private $Dispo = 0;

    public function einzahlen($Betrag) {
        $this->Ktostand += $Betrag; }
    public function abheben ($Betrag) {
        if ($this->Ktostand+$this->Dispo-$Betrag)>0) {
            $this->Ktostand -= $Betrag;
            return true; }
        else {
            echo "Konto nicht gedeckt!";
            return false; } }
}
...
```

OBJEKTE: PROGRAMMBEISPIEL (FORTSETZ.)

```
// Konten anlegen;  
$Kto1 = new Konto;  
$Kto1->KtoNr = "1234567";  
$Kto1->KtoInhaber = "Max Mustermann";  
  
$Kto2 = new Konto;  
$Kto2->KtoNr = "8910123";  
$Kto2->KtoInhaber = "Petra Musterfrau";  
  
// Überweisung tätigen  
$erfolg = $Kto1->abheben(900);  
if ($erfolg==true) {  
    $Kto2->einzahlen(900); }  
else {  
    echo "Überweisung konnte nicht ausgeführt werden!";}
```

DEKLARATION VON KLASSEN

Bevor ein Objekt angelegt werden kann, muss dessen Aufbau, quasi dessen "Konstruktionsplan", angegeben werden.

Dieser "Konstruktionsplan" wird Klasse genannt und wird mit dem Schlüsselwort **class** vereinbart.

```
class Klassenname
{
    public $eigenschaft = wert;

    public function Funktionsname ($argumente)
    {
        // PHP-Quelltext
    }
}
```



Die Sichtbarkeit von Attributen und Methoden wird spezielle Modifikatoren geregelt:

- **private** - Methoden und Attribute, die als private deklariert sind, dürfen nur innerhalb der Klasse genutzt werden. Eine Veränderung eines Attributes oder Aufrufen einer Methode von außen ist nicht zulässig.
- **public** - Methoden und Attribute, die als public deklariert werden, sind überall und für jeden sichtbar. Man kann also auch ein Attribut einer Klasse verändern, ohne eine Methode der Klasse aufzurufen.
- **protected** - Methoden und Attribute sind nach außen nicht sichtbar bzw. von außen erreichbar, jedoch in den abgeleiteten Klassen (Unterklassen) sind sie verfügbar (Wird gebraucht in Klassenhierarchien).

DEKLARATION VON METHODEN

Die meisten Methoden arbeiten nur mit den Daten innerhalb des aktuellen Objektes, obwohl dies PHP nicht erzwingt.

Für die als **private** deklarierten Attribute werden wird für den Zugriff von außen mit **public deklarierte** Methoden benötigt.

Innerhalb einer Methode enthält die Variable **\$this** eine Referenz auf das umgebende Objekt.

```
class Konto
{ private $Ktostand;

    public function einzahlen($Betrag)
    { $this->Ktostand += $Betrag; }
}
```

ERZEUGUNG VON OBJEKTEN DURCH INSTANZIIERUNG

- Aus einer Klasse kann ein konkretes Objekt erzeugt werden. Dies wird als **Instanziierung** bezeichnet.
- Die Instanziierung einer Klasse wird durch das Schlüsselwort **new** eingeleitet.

```
$mein_Konto = new Konto;
```

- Bei der Instanziierung kann der Name der entsprechenden Klasse auch durch eine String-Variable gegeben sein.

```
$klasse = 'Konto';
```

```
$mein_Zweitkonto = new $klasse;
```

- Variablen, die Objektreferenzen enthalten, können in gleicher Weise verwendet werden wie normale Variablen.

- Der Zugriff auf die Methoden bzw. Attribute eines Objekts erfolgt durch die **->** - Notation (**Qualifizierung**).

`$Objektname->Methode (Argumente) ;`

bzw.

`$Objektname->Attribut ;`

- **Beispiel**

```
$mein_Konto = new Konto ;
```

```
$mein_Konto->KtoNr = 0 ;
```

```
$mein_Konto->KtoInhaber = "Max Mustermann" ;
```

```
$mein_Konto->einzahlen(1000) ;
```

Als Anwendungsbeispiel soll ein „Schwarzes Brett“ entwickelt werden.

- Das schwarze Brett besteht aus einer Menge von „Aushängen“. Diese Aushänge haben einen Betreff, einen Text, den Namen eines Verfassers und ein Datum.
- Die Aushänge sollen in einer Datenbank verwaltet werden.
- Es soll eine Webapplikation entwickelt werden, die folgende Funktionen erfüllt:
 - ⇒ Anzeige einer Übersicht aller Aushänge
 - ⇒ Detailansicht eines einzelnen Aushangs
 - ⇒ Eingabe eines neuen Aushangs

Folgendes Vorgehen ist geplant

- Definition der Tabelle(n) und Füllen mit Anfangsdaten mit Hilfe von PHPmyAdmin
- Entwicklung von PHP-Skripten
 - ⇒ Anzeige einer Übersicht aller Aushänge (bb_liste.php)
 - ⇒ Detailansicht eines einzelnen Aushangs (bb_display.php)
 - ⇒ Eingabe eines neuen Aushangs (bb_eintrag.html, bb_eintrag.php)
- Die Reihenfolge ist bewusst so gewählt:
 - ⇒ Schwierigkeitsgrad langsam ansteigend, immer eine neue Schwierigkeit kommt hinzu.
 - ⇒ Software von Anfang an lauffähig, keine Durststrecken.

SCHWARZES BRETT: DATENDEFINITION IN MYSQL

- Erstellen Sie mit phpMyAdmin in MySQL eine Tabelle mit dem Namen **blackboard**, die als schwarzes Brett dient, mit folgenden Merkmalen:
 - ⇒ id = Nummer des Aushangs, Primärschlüssel
 - ⇒ betreff = Titel des Aushangs, max. 80 Zeichen
 - ⇒ name (maximal 20 Zeichen)
 - ⇒ datum = Datum der Eingabe
 - ⇒ text = eigentlicher Text des Aushangs
 - ⇒ pin = PIN oder Passwort
 - ⇒ aktiv = Aushang ist „aktiv“ geschaltet
- Tragen Sie mit phpMyAdmin erste Aushänge in die Datenbanktabelle ein.

} *Optional, für
Erweiterungen*

DEFINITION DER TABELLE BLACKBOARD

Server: localhost » Datenbank: u-riekert » Tabelle: blackboard

Anzeigen Struktur SQL Suche Einfügen Exportieren Importieren Mehr

#	Name	Typ	Kollation	Attribute	Null	Standard	Extra
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Nein	kein(e)	AUTO_INCREMENT
<input type="checkbox"/>	2 betreff	varchar(80)	utf8_unicode_ci		Nein		
<input type="checkbox"/>	3 name	varchar(20)	utf8_unicode_ci		Nein		
<input type="checkbox"/>	4 datum	timestamp		on update CURRENT_TIMESTAMP	Nein	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP
<input type="checkbox"/>	5 text	text	utf8_unicode_ci		Nein	kein(e)	
<input type="checkbox"/>	6 pin	int(10)		UNSIGNED ZEROFILL	Nein	0000000000	
<input type="checkbox"/>	7 aktiv	tinyint(3)		UNSIGNED	Nein	0	

Alle auswählen markierte: Anzeigen Bearbeiten Löschen Primärschlüssel Unique Index

- Standarddatentypen: **int** (ganze Zahl), **tinyint** (kleine ganze Zahl), **varchar** (Textstring) und **text** (umfangreicher Text)
- Für das Datum wird der Datentyp **timestamp** verwendet
 - ⇒ Beim Erzeugen/Ändern eines Datensatzes wird automatisch der aktuelle Wert von Datum/Uhrzeit eingetragen.

EIN EINFACHES SKRIPT ZUR AUSGABE ALLER „AUSHÄNGE“

```
...
<?php
$dbhost = "localhost";
$dbuser = "riekert";
$dbpass = "geheim";
$dbname = "u-riekert";

mysql_connect($dbhost, $dbuser, $dbpass);
mysql_select_db($dbname);

$abfrage = "SELECT * FROM blackboard";
$ergebnis = mysql_query($abfrage);

while ($row = mysql_fetch_assoc($ergebnis)) {
    echo $row['betreff'] . "<br/>\n";
}
?>
```

*PHP-Code zur Ausgabe
der „Betrefe“ aller Aushänge*

Schwarzes Brett

Blackboard Relaunch
nur mal so...
Grüße

ERLÄUTERUNG DES PHP-SCRIPTS ZUM DATENBANKZUGRIFF

```
$dbhost = "localhost";  
$dbuser = "riekert";  
$dbpass = "geheim";  
$dbname = "u-riekert";
```

Hostname (Servername)

Benutzername

Passwort

Datenbankname

Verbindung mit Datenbanksystem herstellen

```
mysql_connect($dbhost, $dbuser, $dbpass);  
mysql_select_db($dbname);
```

Datenbank auswählen

Abfrage definieren und in Variable \$abfrage abspeichern

```
$abfrage = "SELECT * FROM blackboard";  
$ergebnis = mysql_query($abfrage);
```

Ausführen der Abfrage

Nächste Ergebniszeile holen – solange eine kommt

```
while ($row = mysql_fetch_assoc($ergebnis)) {  
    echo $row['betreff'] . "<br/>\n";  
}
```

Zugriff auf Datenfeld als Element eines assoziativen Arrays

AUSGABE ALLER „AUSHÄNGE“: VERBESSERTES SCRIPT

bb_test.php (ohne umgebenden HTML-Rahmen)

```
<?php  
include_once("bb_userdata.php");
```

Das Setzen von \$dbhost, \$dbuser usw. wird in eine **Includedatei** ausgelagert.

```
if (mysql_connect($dbhost, $dbuser, $dbpass)  
    AND mysql_set_charset("utf8")  
    AND mysql_select_db($dbname)) {
```

UTF8-Zeichensatz verwenden!

Wenn alle drei mysql-Funktionen **TRUE** zurückliefern, geht's hier weiter.

```
$abfrage = "SELECT * FROM blackboard";  
$ergebnis = mysql_query($abfrage);
```

```
while ($row = mysql_fetch_assoc($ergebnis)) {  
    $betreff = $row['betreff'];  
    echo "$betreff <br/>\n";  
}  
}
```

Wert von **\$betreff** in String einfügen.

mysql_fetch_assoc() ergibt **FALSE**, wenn keine Zeile mehr da ist.

```
else {  
    echo "Keine Verbindung zu Datenbank möglich!";  
}  
?>
```

Fehlerausgabe, wenn eine der mysql-Funktionen **FALSE** ergab.

INHALT DER INCLUDEDATEI

```
include_once("bb_userdata.php");
```

Mit einer Include-Anweisung kann Code aus einer anderen Datei (hier `bb_userdata.php`) in eine PHP-Datei eingefügt werden.

bb_userdata.php

```
<?php
$dbhost = "localhost";
$dbuser = "riekert";
$dbpass = "geheim";
$dbname = "u-riekert";
?>
```

`localhost` = dieser Computer, auf dem auch PHP läuft.
Falls ein fremder Datenbankserver benutzt wird, muss hier dessen Domain Name oder IP-Adresse eingetragen werden.

Vorteil der Includedatei: Keine Redundanz.

Wenn sich die Daten ändern (z.B. Datenbankserver oder Passwort), genügt eine Änderung an einer zentralen Stelle.

Vertrauliche Daten (Passwort) liegen nur in einer Datei.

AUSGABE DER AUSHÄNGE ALS TABELLE – WIE GEHT DAS?

Aufgabe:

- Wie könnte man die Aushänge als Tabelle ausgeben?
- Wie könnte man den Betreff jeweils anklickbar machen?

Betreff	Name
Informationssysteme	riekert
Wellensittich	xy007
Schlüssel verloren	schussel

Lösungsansatz:

- Zunächst mal eine Tabelle mit HTML gestalten nach Anleitung (z.B. <http://webkompetenz.wikidot.com/html-handbuch:einfache-tabelle>) oder mit Hilfe eines HTML-Editors (z.B. Dreamweaver).
- Dann nach diesem Muster eine PHP-Seite entwickeln.
- Der Betreff sollte als Hyperlink ausgebildet werden:
http://mars.iuk.hdm-stuttgart.de/riekert/bb_display.php?id=1
http://mars.iuk.hdm-stuttgart.de/riekert/bb_display.php?id=2
usw. Allerdings muss die id durch PHP eingefügt werden.

ERSTELLEN VON TABELLEN MIT HTML

Id	Betreff	Name
1	WG	Paul
2	Skateboard	Susi

```
<body>
  <table>
    <tr>
      <th> Id </th> <th> Betreff </th> <th> Name </th>
    </tr>
    <tr>
      <td> 1 </td> <td> WG </td> <td> Paul </td>
    </tr>
    <tr>
      <td> 2 </td> <td> Skateboard </td> <td> Susi </td>
    </tr>
  </table>
</body>
```

s.a.: <http://webkompetenz.wikidot.com/html-handbuch:einfache-tabelle>

RÄNDER VON TABELLEN

In HTML5 muss man die Ränder einer Tabelle mit CSS definieren.

Am besten in einer CSS-Datei, die z.B. Folgendes enthält:

bb_style.css

```
table, th, td {  
    border: 1px solid gray;  
}
```

Die CSS-Datei wird im Kopfteil der HTML-Datei eingebunden

```
<head>  
<!-- andere Definitionen im HTML-Kopfbereich -->  
<link rel="stylesheet" type="text/css"  
      href="bb_style.css">  
</head>
```

Id	Betreff	Name
1	WG	Paul
2	Skateboard	Susi

AUSGABE DER AUSHÄNGE ALS TABELLE – ÄUSSERER RAHMEN

bb_liste.php (ohne umgebenden HTML-Rahmen)

```
<table>
  <tr> <th>Betreff</th> <th>Name</th> </tr>
<?php
include_once("bb_userdata.php");
if (mysql_connect($dbhost, $dbuser, $dbpass)
    AND mysql_set_charset("utf8")
    AND mysql_select_db($dbname)) {
$abfrage = "SELECT * FROM blackboard";
$ergebnis = mysql_query($abfrage);

... Schleife zur Ausgabe der Tabellenzeilen, siehe nächste Folie ...

}
else {echo "Keine Verbindung zu Datenbank moeglich!";}
?>
</table>
```

AUSGABE DER AUSHÄNGE ALS TABELLE – INNERE SCHLEIFE

bb_liste.php (Schleife im inneren Teil)

```
while ($row = mysql_fetch_assoc($ergebnis)) {  
    $id = $row['id'];  
    $betreff = $row['betreff'];  
    $name = $row['name'];  
    echo "<tr>  
        <td><a href='bb_display.php?id=$id'>  
            $betreff </a> </td>  
        <td>$name</td>  
    </tr>\n";  
}
```

Betreff	Name
Blackboard Relaunch	riekert
nur mal so...	HansDampf
Grüße	August

AUSGABE DER AUSHÄNGE ALS TABELLE – SCREENSHOT

The screenshot shows a web browser window titled "Schwarzes Brett" with the URL [http://mars.i...](http://mars.iuk.hdm-stuttgart.de/~riekert/). The main heading is "Schwarzes Brett". Below it is a table of forum posts:

Betreff	Name
Blackboard Relaunch	riekert
nur mal so.	HansDampf
Grüße	August

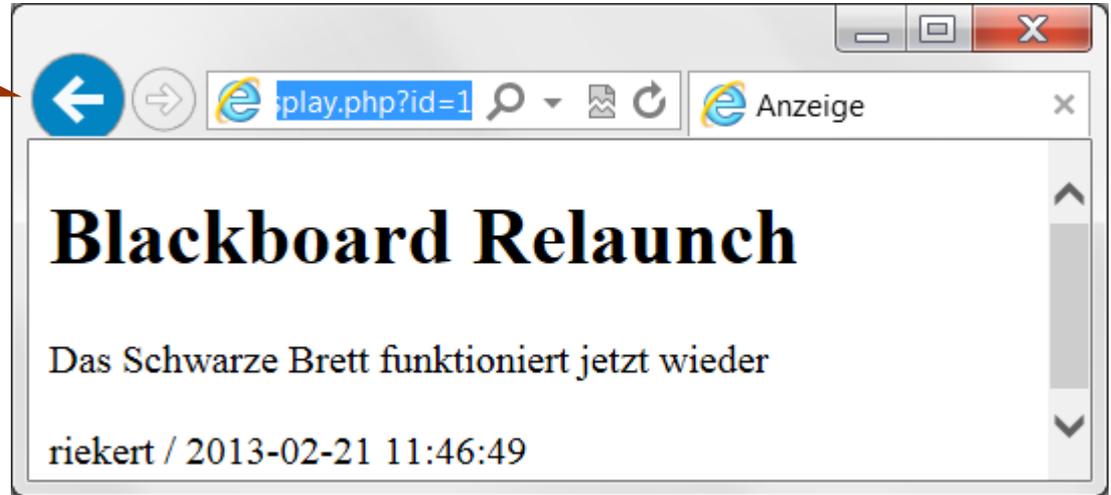
Below the table is a link [Neuer Aushang](#). At the bottom of the browser window, a text box contains the URL: http://mars.iuk.hdm-stuttgart.de/~riekert/bb_display.php?id=2.

Annotations:

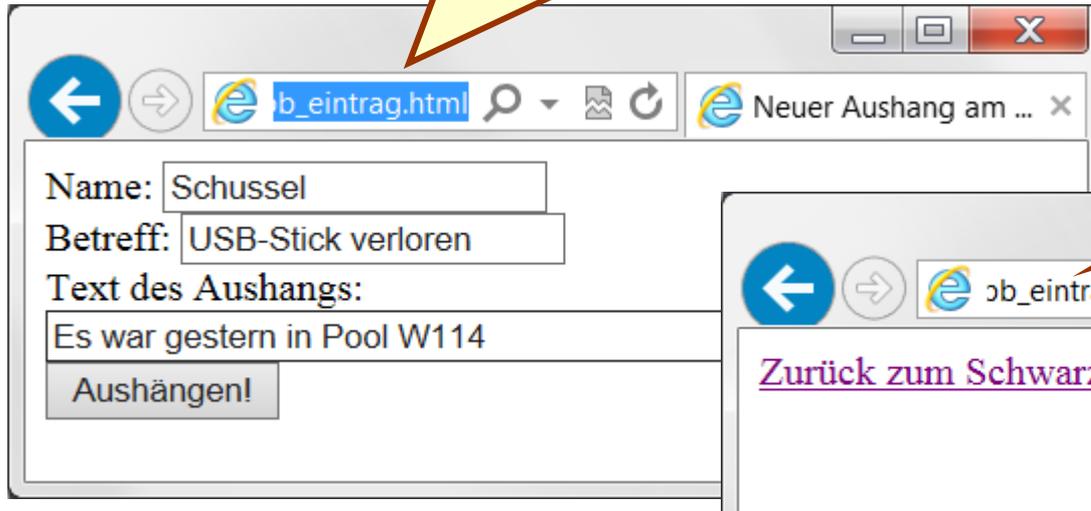
- A yellow callout box points to the "nur mal so." link with the text: "Wenn dieser Hyperlink angeklickt wird, landet man auf dieser URL".
- Another yellow callout box points to the "Blackboard Relaunch" link with the text: "Aufgabe: Schreiben Sie das Skript bb_display.php zur kompletten Anzeige eines einzelnen „Aushangs“".

FERTIGSTELLUNG: WEITERE WEBSEITEN UND SKRIPTE

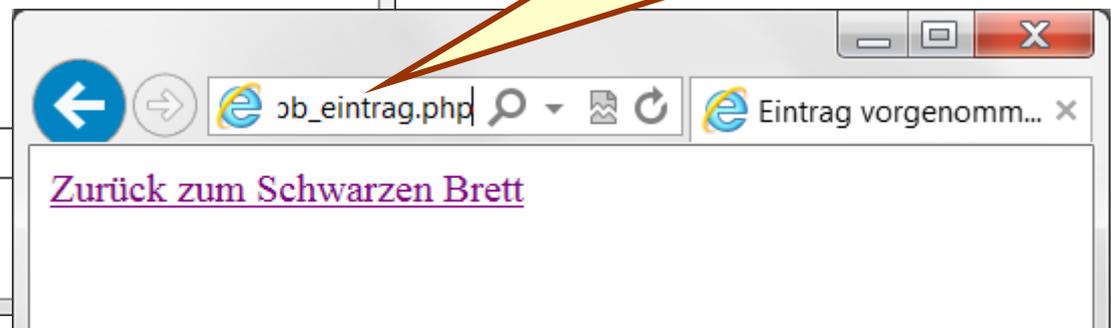
bb_display.php?id=...
Vollst. Anzeigen eines
einzelnen Aushangs



bb_eintrag.html
Formular zur Eingabe
eines Aushangs



bb_eintrag.php
Skript zur Verarbeitung
einer Aushangeingabe



ANZEIGEN EINES EINZELNEN AUSHANGS

*bb_display.php (ohne umgebenden HTML-Rahmen
und Verbindungsaufnahme zur Datenbank)*

```
$id=(int)$_GET["id"];  
$abfrage = "SELECT * FROM blackboard WHERE id=$id";  
$ergebnis = mysql_query($abfrage);  
if ($row = mysql_fetch_assoc($ergebnis)) {  
    $betreff = $row['betreff'];  
    $text = $row['text'];  
    $name = $row['name'];  
    $datum = $row['datum'];  
    echo "<h1>$betreff</h1>";  
    echo "$text<br/><br/>";  
    echo "$name / $datum";  
}  
else {  
    echo "Datensatz mit id=$id nicht gefunden!";  
}
```

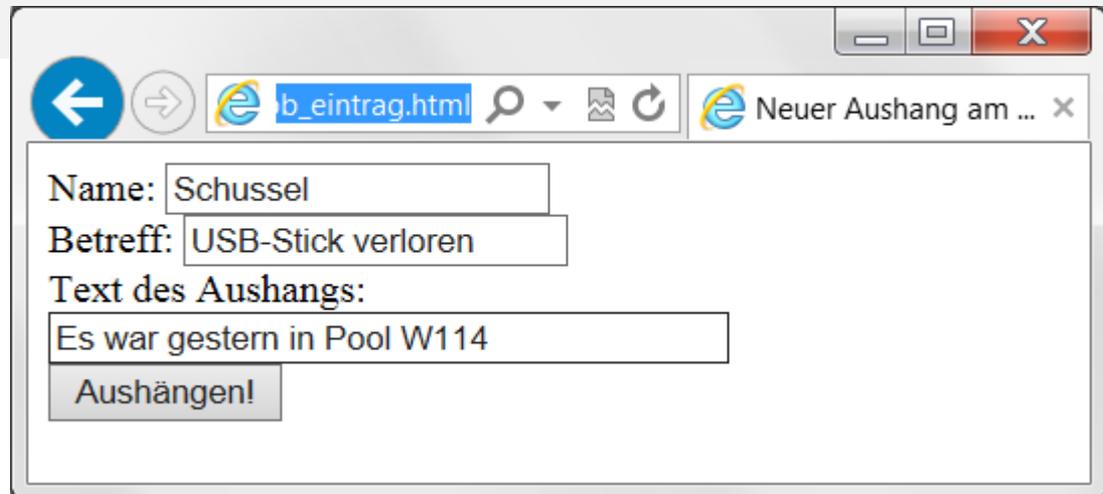
Übernahme der id aus der URL



FORMULAR ZUR EINGABE EINES AUSHANGS

bb_eintrag.html (ohne umgebenden HTML-Rahmen)

```
<form action="bb_eintrag.php" method="post">  
  Name:      <input type="text" name="name" /> <br />  
  Betreff:   <input type="text" name="betreff" /> <br />  
  Text des Aushangs: <br />  
  <input type="text" name="text"  
    size="40" maxlength="80" /> <br />  
  <input type="submit" value="Aushängen!" />  
</form>
```



The screenshot shows a web browser window with the address bar displaying "bb_eintrag.html". The page content is a form with the following fields and controls:

- Name: Schussel
- Betreff: USB-Stick verloren
- Text des Aushangs: Es war gestern in Pool W114
- Aushängen! (submit button)

SKRIPT ZUR VERARBEITUNG EINER AUSHANGEINGABE

bb_eintrag.php (ohne umgebenden HTML-Rahmen)

UTF-8-Eingaben übernehmen und als HTML-Sonderzeichen kodieren

```
$name= htmlspecialchars($_POST["name"], ENT_QUOTES, "UTF-8");  
$betreff= htmlspecialchars($_POST["betreff"], ENT_QUOTES,  
                            "UTF-8");  
$text= htmlspecialchars($_POST["text"], ENT_QUOTES, "UTF-8");
```

```
include_once("bb_userdata.php");  
if (mysql_connect($dbhost, $dbuser, $dbpass)  
    AND mysql_set_charset("utf8")  
    AND mysql_select_db($dbname)) {  
    $abfrage = "INSERT INTO blackboard (name, betreff, text)  
              VALUES ('$name', '$betreff', '$text')";  
    $ergebnis = mysql_query($abfrage);  
    echo "Aushang $betreff von $name eingetragen <br/>";  
}  
else { echo "Keine Verbindung zu Datenbank möglich!";}
```

Vorbereiten
einer Abfrage
mit Parametern

und ausführen

- Bei Neueingabe: automatische Erzeugung und Versand einer PIN an die Emailadresse name@hdm-stuttgart.de. Erst nach Eingabe dieser PIN kann der Aushang aktiv geschaltet werden
- Anzeigefunktion soll nur aktive Aushänge anzeigen
- Möglichkeit der Bearbeitung und des Löschens von Aushängen (erfordert Angabe der PIN)
- Feedbackformular: Versand einer Email an den Ersteller des Aushangs (name@hdm-stuttgart.de)
- Suchfunktionen, Sortierfunktionen
- Verwendung einer weiteren Tabelle, z.B. für Kategorien oder Autoren der Aushänge.

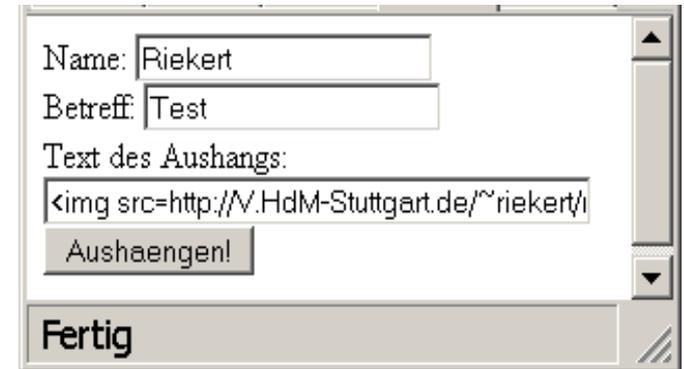
Auf den folgenden Seiten sind vertiefende Themen zu Webapplikationen wiedergegeben:

- Sicherheitsaspekte von Webapplikationen
- Schlüsselbeziehungen und referenzielle Integrität in MySQL
- File Upload mit PHP
- Neue Website definieren mit dem Dreamweaver

- Riskant: Vertrauen auf korrekte Formularparameter. Jedes Skript kann auch von einem „selbstgebastelten“ Formular mit inkorrekten Parametern aufgerufen werden.
- Riskant: Vertrauen auf korrekt gesetzte versteckte Formularfelder und korrekt gesetzte Cookies. Auch diese können manipuliert werden.
- Riskant: Ungeprüfte Weiterverwendung von Formularparametern zur Erzeugung von Webseiten, SQL-Kommandos und Betriebssystemkommandos oder zur Verwendung als Dateinamen.
- Problematisch aber bei Webhostern üblich: Mehrere Websites auf demselben Server (denn Webserverprozess hat Zugriff auf alle Webverzeichnisse, auch die fremder Websites)

PROBLEME DURCH UNGEPRÜFTE FORMULARVARIABLE (HTML)

- Variablen aus Formularen werden manchmal auf Webseiten ausgegeben
- Probleme entstehen, wenn der User HTML-Code ins Formular eingibt.
- Abhilfe: HTML-Kommandos entfernen mit Funktion `strip_tags()` oder nur im Quelltext sichtbar machen mit `html_entities()`:



A screenshot of a web form with the following fields:

- Name: Riekert
- Betreff: Test
- Text des Aushangs: `<img src=http://V.HdM-Stuttgart.de/~riekert/i`

Below the text field is a button labeled "Aushaengen!". At the bottom of the form is a button labeled "Fertig".



HTML-CODE AUS AUSGABE ENTFERNEN

`htmlspecialchars()`

PHP-Code:

```
<?php
$str = "<b>fett</b>";
echo htmlspecialchars($str);
?>
```

Ausgabe im HTML-Code:

```
&lt;b&gt;fett&lt;/b&gt;
```

Anzeige im Browser:

```
<b>fett</b>
```

HTML-Code wird angezeigt,
aber nicht ausgeführt

`strip_tags()`

PHP-Code:

```
<?php
$str = "<b>fett</b>";
echo strip_tags($str);
?>
```

Ausgabe im HTML-Code:

```
fett
```

Anzeige im Browser:

```
fett
```

HTML-Code wird ignoriert

PROBLEME DURCH UNGEPRÜFTE FORMULARPARAMETER (SQL)

```
$text=$_POST["text"];
```

```
$abfrage = "INSERT INTO blackboard (name,betreff, text)  
            VALUES ('$name', '$betreff', '$text')";
```

Was passiert, wenn für `$text` Folgendes eingegeben wird?

```
blabla'); DELETE from blackboard;
```

Möglicherweise wird der DELETE-Befehl ausgeführt!

Abhilfe:

- Quotes (" bzw. ') unwirksam machen durch Funktion `addslashes()`
- Oder PHP mit Option "magic quotes" konfigurieren (ist auf Mars der Fall)

- Normalerweise ist der PHP-Quellcode nicht sichtbar
 - ⇒ User sehen nur den generierten HTML-Code
 - ⇒ Sicherheitsgewinn („security by obscurity“)
- Aber: PHP-Skripte werden serverseitig vom Webserver-Prozess ausgeführt. Dieser Prozess hat Zugriff auf alle PHP-Skripte im Quelltext.
 - ⇒ Sicherheitsproblem auf Servern, die mehrere Websites hosten.
 - ⇒ Ein Skript auf einer Website kann Dateien auf einer anderen Website öffnen, lesen und interpretieren.
- Abhilfe (für Webmaster): PHP im „Safe Mode“ konfigurieren. Der PHP-Interpreter verbietet dann solche Zugriffe

SCHLÜSSELBEZIEHUNGEN IN (MY)SQL UND REFERENZIELLE INTEGRITÄT



teilnahme

mid	pnr

Die Kombination (mid, pnr) ist Primärschlüssel
mid ist Fremdschlüssel
pnr ist Fremdschlüssel

mitarbeiter

mid	nachname	vorname

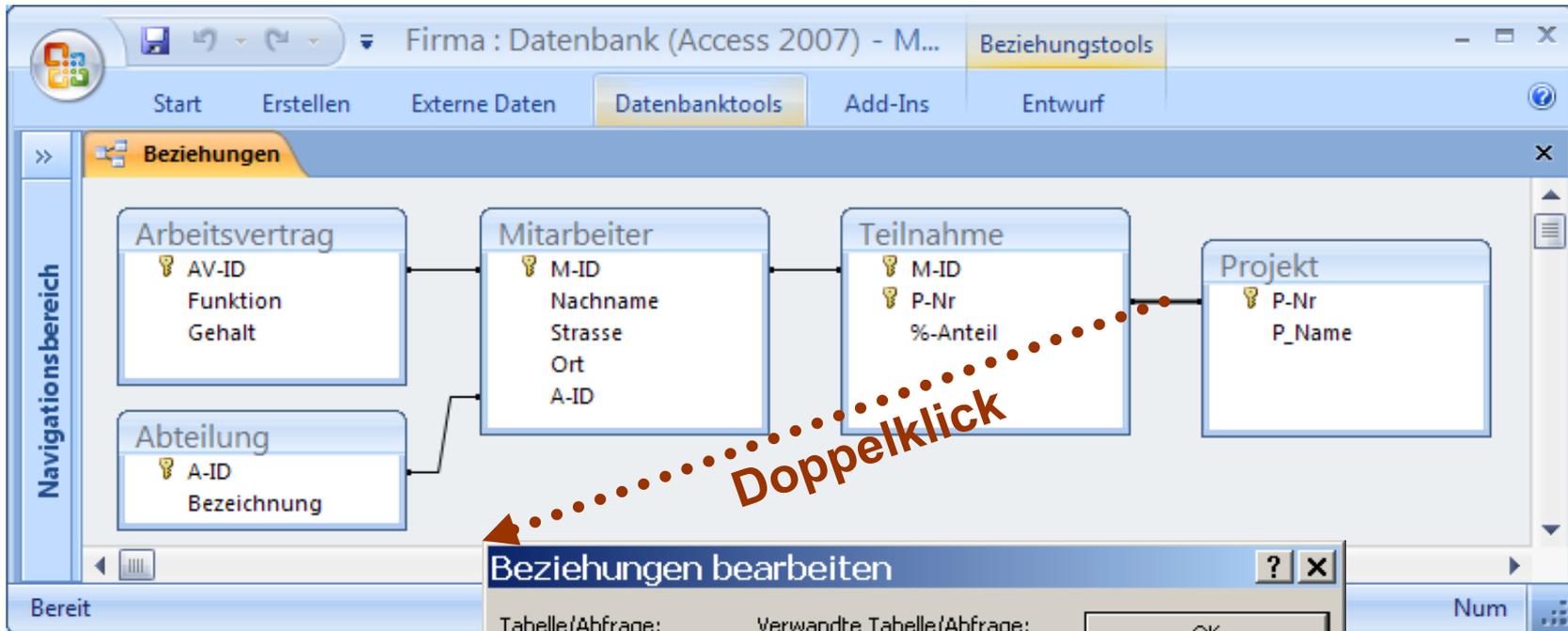
mid ist Primärschlüssel

projekt

pnr	bezeichnung

pnr ist Primärschlüssel

ZUR ERINNERUNG: BEISPIEL MS-ACCESS



Doppelklick

Beziehungen bearbeiten

Tabelle/Abfrage:	Verwandte Tabelle/Abfrage:
Projekt	Teilnahme
P-Nr	P-Nr

Mit referentieller Integrität
 Aktualisierungsweitergabe an verwandte Felder
 Löschweitergabe an verwandte Datensätze

Beziehungstyp: 1:n

Buttons: OK, Abbrechen, Verknüpfungstyp...

Ref. Integrität

Aktualisierungsweitergabe

Löschweitergabe

TABELLEN ERZEUGEN MIT PHPMYADMIN

The screenshot shows the phpMyAdmin interface in a Firefox browser window. The address bar shows the URL `https://mars.iuk.hdm-stuttgart.de/phpmyadmin/db_structure`. The interface displays the database structure for 'u-riekert' on 'localhost'. A table list shows three tables: 'blackboard', 'mitarbeiter', and 'projekt'. The 'mitarbeiter' and 'projekt' tables are circled in red. Below the table list, there is a dialog box titled 'Neue Tabelle in Datenbank u-riekert erstellen' (Create new table in database u-riekert). The dialog has a 'Name' field containing 'teilnahme' and an 'Anzahl der Felder' (Number of fields) field containing '2'. The 'OK' button is visible at the bottom right of the dialog.

	Tabelle	Aktion	Einträge	Typ	Kollation	Größe	Überhang
<input type="checkbox"/>	blackboard		2	InnoDB	utf8_unicode_ci	16,0 KiB	-
<input type="checkbox"/>	mitarbeiter		0	InnoDB	utf8_unicode_ci	16,0 KiB	-
<input type="checkbox"/>	projekt		0	InnoDB	utf8_unicode_ci	16,0 KiB	-
3 Tabellen		Gesamt	2	InnoDB	utf8_unicode_ci	48,0 KiB	0 Bytes

Alle auswählen / Auswahl entfernen markierte:

Druckansicht Strukturverzeichnis

Neue Tabelle in Datenbank u-riekert erstellen

Name Anzahl der Felder:

ZWISCHENTABELLE TEILNAHME: FELDER DEFINIEREN

The screenshot shows the 'tbl_create.pl' page in phpMyAdmin for a table named 'teilnahme'. The fields 'mid' and 'pnr' are defined with the following properties:

Feld	mid	pnr
Typ	INT	INT
Länge/Set ¹		
Standard ²	Kein	Kein
Kollation		
Attribute		
Null	<input type="checkbox"/>	<input type="checkbox"/>
Index	PRIMARY	PRIMARY
AUTO_INCREMENT	<input type="checkbox"/>	<input type="checkbox"/>
Kommentare		
MIME-Typ		
Darstellungsumwandlung		
Umwandlungsoptionen ³		

At the bottom, the 'Tabellenformat' is set to 'InnoDB'.

Kombination (mid, pnr) ist Primärschlüssel

Referentielle Integrität funktioniert nur mit InnoDB

ZWISCHENTABELLE TEILNAHME STRUKTUR

Firefox | mars.iuk.hdm-stuttgart.de / loca... | hdm-stuttgart.de | https://mars.iuk.hdm-stuttgart.de/phpmyadmin/tbl_structure | access sql export

localhost | u-riekert | **teilnahme**

Anzeigen | Struktur | SQL | Suche | Tracking | Einfügen | Exportieren

Importieren | Operationen | Leeren | Löschen

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion					
<input type="checkbox"/>	<u>mid</u>	int(11)			Nein	Kein							
<input type="checkbox"/>	<u>pnr</u>	int(11)			Nein	Kein							

Alle auswählen / Auswahl entfernen *markierte:*

Druckansicht | **Beziehungsübersicht** | Tabellenstruktur analysieren | Verfolge Tabelle

1 Felder hinzufügen An das Ende der Tabelle An den Anfang der Tabelle Nach

Indizes:

Aktion	Name	Typ	Unique	Gepackt	Feld	Kardinalität	Kollation	Null	Kommentar
		PRIMARY	BTREE	Ja	Nein	mid	0	A	
					pnr	0	A		

FREMDSCHLÜSSEL UND REFERENTIELLE INTEGRITÄT

The screenshot shows the phpMyAdmin interface. At the top, there is a navigation bar with buttons: Anzeigen, **Struktur** (circled in red), SQL, Suche, Tracking, Einfügen, and Exportieren. Below this is another row of buttons: Importieren, Operationen, Leeren, and Löschen. The main area displays a table structure with columns: Feld, Typ, Kollation, Attribute, Null, Standard, Extra, and Aktion. Two fields are listed: 'mid' and 'pnr', both of type 'int(11)'. Below the table, there are buttons for 'Alle auswählen / Auswahl entfernen' and 'markierte:'. At the bottom, there are more navigation buttons: Druckansicht, **Beziehungsübersicht** (circled in red), Tabellenstruktur analysieren, and Verfolge Tabelle.

The screenshot shows a Firefox browser window displaying the phpMyAdmin 'Beziehungsübersicht' (Relationships Overview) page. The browser address bar shows 'https://mars.iuk.hdm-stuttgart.de/phpmyadmin/index.php?d'. The page content includes the same navigation bar as the previous screenshot. Below the navigation bar, there is a section titled 'Verweise' (References). It contains a table with two columns: 'Interne Beziehungen¹' and 'FOREIGN KEY (INNODB)'. The table has two rows:

Interne Beziehungen ¹	FOREIGN KEY (INNODB)
mid	`u-riekert`.`mitarbeiter`.`mid` ON DELETE CASCADE ON UPDATE
pnr	`u-riekert`.`projekt`.`pnr` ON DELETE CASCADE ON UPDATE

Two callout boxes are present: a yellow one pointing to the 'ON DELETE CASCADE' part of the first row, labeled 'Löschweitergabe', and a yellow one pointing to the 'ON UPDATE CASCADE' part of the second row, labeled 'Aktualisierungsweitergabe'. A red arrow points from the 'Beziehungsübersicht' button in the top screenshot to the browser window.

ERZEUGUNG DER TABELLEN MIT SQL (1)

```
CREATE TABLE mitarbeiter
  (mid INTEGER NOT NULL AUTO_INCREMENT,
   nachname VARCHAR(50),
   vorname VARCHAR(50),
   PRIMARY KEY (mid))
ENGINE=InnoDB;
```

```
CREATE TABLE projekt
  (pnr INTEGER NOT NULL,
   bezeichnung VARCHAR(30),
   PRIMARY KEY (pnr))
ENGINE=InnoDB;
```

ERZEUGUNG DER TABELLEN MIT SQL (1)

```
CREATE TABLE teilnahme
  (mid INTEGER NOT NULL,
   pnr INTEGER NOT NULL,
   PRIMARY KEY (mid,pnr) ,
   FOREIGN KEY (mid)
     REFERENCES mitarbeiter (mid)
     ON DELETE CASCADE ,
   FOREIGN KEY (pnr)
     REFERENCES projekt (pnr)
     ON DELETE CASCADE
     ON UPDATE CASCADE)
ENGINE=InnoDB;
```

Kombinierter Primärschl.

Fremdschlüsseldeklaration

Löschweitergabe

Fremdschlüsseldeklaration

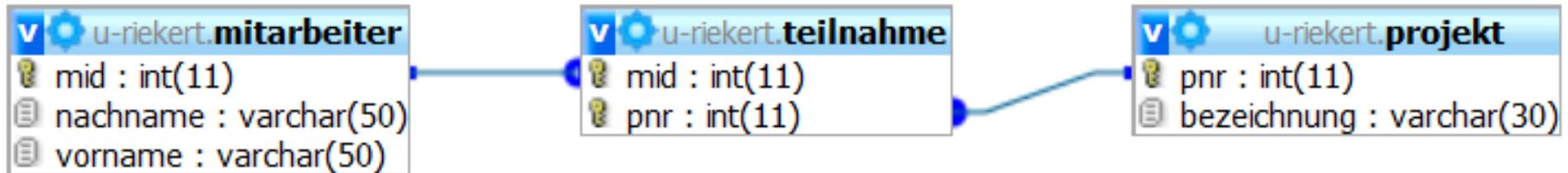
Löschweitergabe

Aktualisierungswweitergabe

InnoDB ermöglicht referenzielle Integrität

DATENBANK-SCHEMA UND – INHALTE DES BEISPIELS

Datenbank-Schema im „PHPMYADMIN-Designer“



Datenbank-Inhalte (Excel-Export)

mitarbeiter		
mid	nachname	vorname
1	Riekert	Wolf-Fritz
2	Thies	Peter
3	Rathke	Christian

teilnahme	
mid	pnr
1	20
1	21
3	21

projekt	
pnr	bezeichnung
20	Social Media
21	Stuttgart 21

UPDATE projekt SET pnr = '22' WHERE pnr = 21;

Ergebnis: Aktualisierungsweitergabe
Auch in teilnahme wird pnr verändert

projekt	
pnr	bezeichnung
20	Social Media
22	Stuttgart 21

teilnahme	
mid	pnr
1	20
1	22
3	22

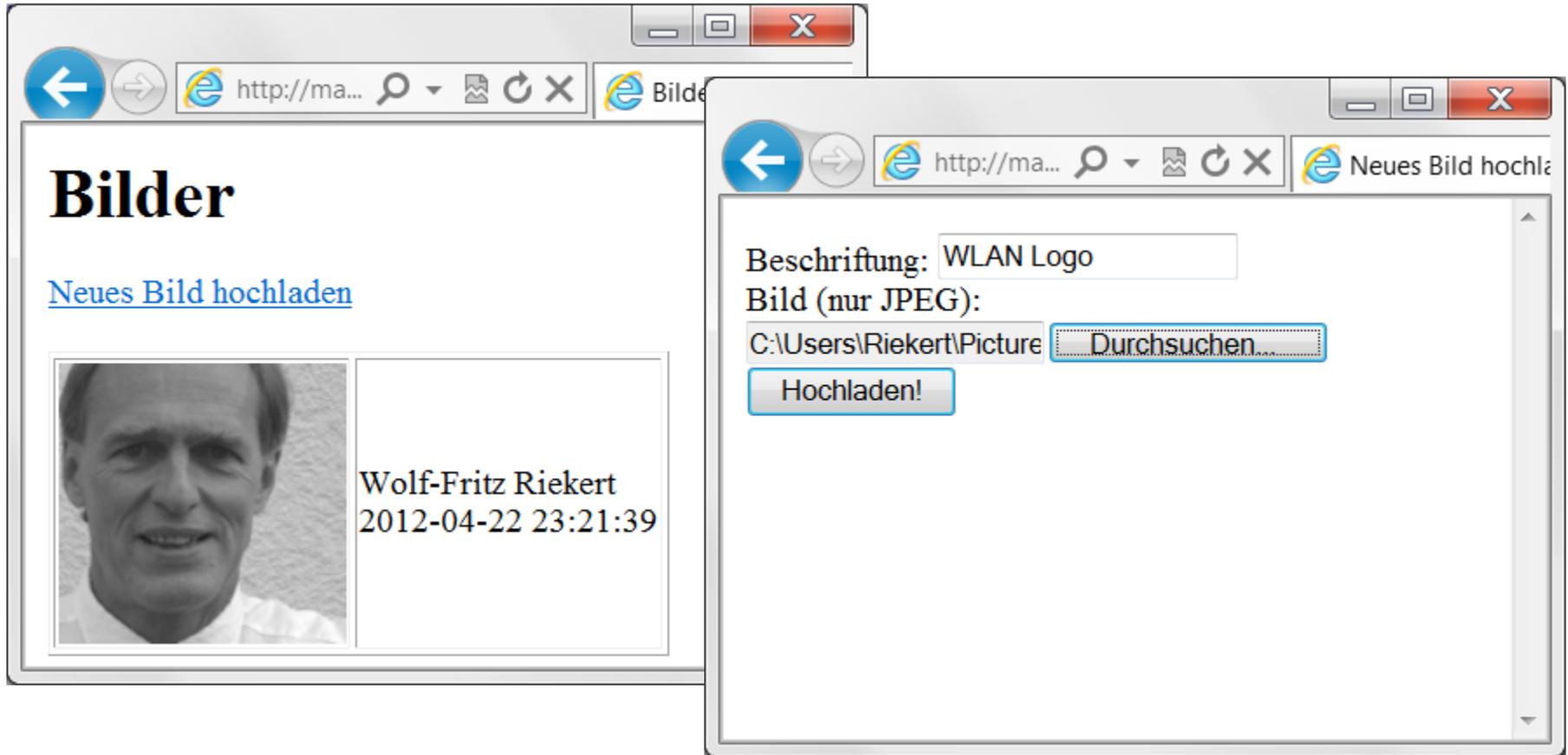
DELETE FROM mitarbeiter WHERE mid = 1;

Ergebnis: Löschweitergabe
Auch die Datensätze in teilnahme mit mid = 1 werden gelöscht

mitarbeiter		
mid	nachname	vorname
2	Thies	Peter
3	Rathke	Christian

teilnahme	
mid	pnr
3	22

FILE UPLOAD AM BEISPIEL BILDERALBUM



	Feld	Typ	Kollation	Attribute	Null	Standard	Extra
<input type="checkbox"/>	id	int(11)			Nein	Kein	AUTO_INCREMENT
<input type="checkbox"/>	beschriftung	varchar(80)	latin1_swedish_ci		Nein		
<input type="checkbox"/>	datum	timestamp		on update CURRENT_TIMESTAMP	Nein	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP
<input type="checkbox"/>	bild	blob		BINARY	Nein	Kein	

ALLE BILDER ANZEIGEN: PHP-CODE VON bilder_anzeigen.php

```
<table border="1">
<?php
try {
    include_once("userdata.php");
    $db = new PDO($dsn,$dbuser,$dbpass,$option);
    $sql = "SELECT * FROM bilder";
    $query = $db->prepare($sql);
    $query->execute();
    while ($zeile = $query->fetchObject()) {
        $id = $zeile->id;
        $beschriftung = $zeile->beschriftung;
        $datum = $zeile->datum;
        echo "<tr>\n";
        echo "<td><img src='bilder_jpg.php?id=$id' alt='Bild' />
            </td>\n";
        echo "<td>$beschriftung <br/> $datum</td>";
        echo "</tr>\n"; }
    $db = null; }
catch (PDOException $e) { ... Fehlerbehandlung ausgelassen ... }
?>
</table>
```



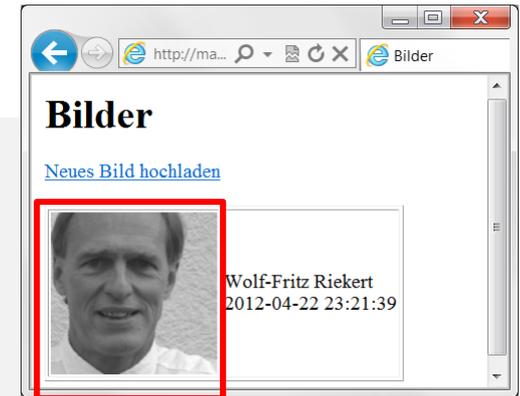
Siehe nächste Seite

AUSGABE EINES BILDES

bilder_jpg.php?id=

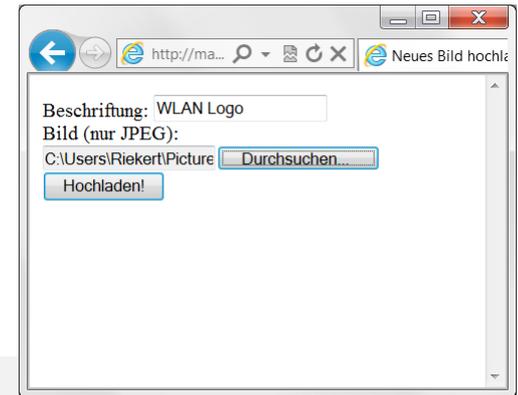
```
<?php
Header("Content-type: image/jpeg");
$id=(int) $_GET["id"];

try {
    include_once("userdata.php");
    $db = new PDO($dsn,$dbuser,$dbpass,$option);
    $sql = "SELECT bild FROM bilder WHERE id=$id";
    $query = $db->prepare($sql);
    $query->execute();
    $bild = $query->fetchObject()->bild;
    echo $bild;
    $db = null; }
catch (PDOException $e) {
    echo ""; }
?>
```



UPLOAD EINES BILDES (FORMULAR)

bilder_hochladen.html



```
<form action="bilder_hochladen.php"
  method=post enctype="multipart/form-data">
  <input type="hidden" name="max_file_size" value="100000">
  Beschriftung: <input type="text" name="beschriftung"> <br />
  Bild (nur JPEG): <input type="file" name="bild" /> <br />
  <input type="submit" value="Hochladen!">
</form>
```

UPLOAD EINES BILDES (SKRIPT)

bilder_hochladen.php (TEIL 1)

```
<?php
$beschriftung=htmlspecialchars($_POST["beschriftung"],
                               ENT_QUOTES, "UTF-8");

$filename=$_FILES['bild']['name']; // Name der Originaldatei
$filetype=$_FILES['bild']['type'];
$filesize=$_FILES['bild']['size'];
$tmpfilename=$_FILES['bild']['tmp_name'];
try {
    if(($filetype=="image/jpeg" || $filetype=="image/pjpeg") AND
        $filesize < 100000 AND !empty($beschriftung)){
        ... Eigentliches Hochladen siehe nächste Folie ...
    }
    else{ print „Error: Fehlerhafte Eingabe <br />“; }
}
catch (PDOException $e) {
    print "Error!: " . $e->getMessage() . "<br />";
    die();
}
?>
```

UPLOAD EINES BILDES (SKRIPT)

bilder_hochladen.php (TEIL 2)

Das eigentliche Hochladen eines Bildes erfolgt wie folgt:

```
include_once("userdata.php");
$db = new PDO($dsn,$dbuser,$dbpass,$option);
$fd = file_get_contents($tmpfilename);
$insert = $db->prepare("INSERT INTO bilder
    (beschriftung,bild) VALUES(:beschriftung, :bild)");
$insert->execute( array("beschriftung" => $beschriftung,
    "bild" => $fd) );

$db = null;
echo "Bild $beschriftung hochgeladen<br/>";
```

Zur Erinnerung:

```
$beschriftung=htmlspecialchars($_POST["beschriftung"],
    ENT_QUOTES, "UTF-8");
$tmpfilename=$_FILES['bild']['tmp_name'];
```

NEUE WEBSITE DEFINIEREN MIT DREAMWEAVER (1)

Site-Definition für mars

Grundeinstellungen **Erweitert**

Site-Definition

Dateibearbeitung Dateitest Dateifreigabe

Eine Site in Adobe Dreamweaver CS3 ist eine Sammlung von Dateien und Ordnern, die einer Website auf einem Server entsprechen.

Wie möchten Sie die Site nennen?

Beispiel: MeineSite

Wie lautet die HTTP-Adresse (URL) Ihrer Site?

Beispiel: <http://www.meinHost.com/meineSite>

Wenn Sie direkt auf einem Server mit FTP oder RDS arbeiten möchten, müssen Sie [eine FTP- oder RDS-Serververbindung erstellen](#). Beim Arbeiten direkt auf dem Server ist das Ausführen von Operationen wie die Überprüfung von Hyperlinks oder das Erstellen von Site-Berichten jedoch nicht für die gesamte Site möglich.

NEUE WEBSITE DEFINIEREN MIT DREAMWEAVER (2)

Site-Definition für mars

Grundeinstellungen **Erweitert**

Site-Definition

Dateien bearbeiten, Teil 2 Dateitest Dateifreigabe

Möchten Sie eine Servertechnologie wie ColdFusion, ASP.NET, ASP, JSP oder PHP verwenden?

Nein, keine Servertechnologie verwenden.

Ja, Servertechnologie verwenden.

Welche Servertechnologie?

PHP MySQL ▼

NEUE WEBSITE DEFINIEREN MIT DREAMWEAVER (3)

Site-Definition für mars

Grundeinstellungen **Erweitert**

Site-Definition

Dateien bearbeiten, Teil 3 Dateitest Dateifreigabe

Wie möchten Sie Ihre Dateien in der Entwicklungsphase bearbeiten?

- Lokal bearbeiten und testen (Testserver ist auf diesem Computer)
- Lokal bearbeiten, dann auf Remote-Testserver laden
- Über lokales Netzwerk direkt auf dem Remote-Testserver bearbeiten

Wo auf Ihrem Rechner möchten Sie Ihre Dateien speichern?



NEUE WEBSITE DEFINIEREN MIT DREAMWEAVER (4)

Site-Definition



Dateibearbeitung **Dateitest** Dateifreigabe

Wie stellen Sie eine Verbindung zum Testserver her?

FTP

Wie lautet der Hostname oder die FTP-Adresse Ihres Webserver?

mars.iuk.hdm-stuttgart.de

In welchem Ordner des Testservers möchten Sie die Dateien speichern?

public_html

Wie lautet Ihr FTP-Benutzername?

riekert

Wie lautet Ihr FTP-Kennwort?

●●●●●●●●

Speichern

Secure FTP (SFTP) verwenden

Testverbindung

NEUE WEBSITE DEFINIEREN MIT DREAMWEAVER (5)

Site-Definition für mars

Grundeinstellungen **Erweitert**

Site-Definition

Dateibearbeitung **Dateien testen, Teil 2** Dateifreigabe

Dreamweaver kommuniziert mit Ihrem Testserver über HTTP (genau wie ein Browser). Hierfür muss er den URL des Stammordners Ihrer Site kennen.

Welchen URL möchten Sie für den Stammordner der Site verwenden?

Beispiel: http://Server1/Stammordner/

URL testen

NEUE WEBSITE DEFINIEREN MIT DREAMWEAVER (6)

Site-Definition für mars

Grundeinstellungen **Erweitert**

Site-Definition

Dateibearbeitung Dateitest **Dateifreigabe**

Möchten Sie das Ein- und Auschecken von Dateien aktivieren, um zu verhindern, dass eine Datei zur selben Zeit von weiteren Personen bearbeitet wird?

Ja, Ein- und Auschecken aktivieren.

Nein, Ein- und Auschecken nicht aktivieren.

NEUE WEBSITE DEFINIEREN MIT DREAMWEAVER (7)

Site-Definition für mars

Grundeinstellungen **Erweitert**

Site-Definition



Zusammenfassung

Für Ihre Site gelten folgende Einstellungen:

Lokale Informationen:

Site-Name: mars

Lokaler Stammordner: C:\Users\Riekert\Documents\sites\mars\

Remote-Informationen:

Zugriff: FTP

FTP-Host: mars.iuk.hdm-stuttgart.de

Ein-/Auschecken: Deaktiviert

Testserver:

Zugriff: FTP

FTP-Host: mars.iuk.hdm-stuttgart.de

Sie können Ihre Site über die Registerkarte "Erweitert" weiter konfigurieren.

NEUE WEBSITE DEFINIEREN MIT DREAMWEAVER (8)

Site-Definition für mars

Grundeinstellungen | Erweitert

Kategorie

- Lokale Infos
- Remote-Informationen
- Testserver
- Cloaking
- Design Notes
- Sitemap-Layout
- Dateiansichtsspalten
- Contribute
- Vorlagen
- Spry

Lokale Infos

Site-Name: mars

Lokaler Stammordner: C:\Users\Riekert\Documents\sites\mars\ 

Standard-Bilderordner: 

Hyperlinks relativ zu: Dokument Stammordner

HTTP-Adresse: http://mars.iuk.hdm-stuttgart.de

Diese Adresse wird für Hyperlinks verwendet, die relativ zur Site angegeben werden. Ferner erkennt der Hyperlink-Prüfer anhand dieser Adresse HTTP-Links, die auf Ihre eigene Site verweisen.

Groß-/Kleinschr. bei Hyperlinks: Groß-/Kleinschreibung bei Hyperlink-Prüfung beachten

Cache: Cache aktivieren

Im Cache werden Datei- und Elementinformationen in der Site gespeichert. Dies beschleunigt die Funktionen des

DREAMWEAVER: WEBSEITE BEARBEITEN

Adobe Dreamweaver CS3 - [C:\Users\Riekert\Documents\sites\mars\index.html (XHTML)*]

Datei Bearbeiten Ansicht Einfügen Modifizieren Text Befehle Site Fenster Hilfe

Einfügen Allgemein Layout/Formul Daten Spry Text Favorite

index.html* Code Teilen Entwurf Titel: Riekerts Homepage

0 50 100 150 200 250 300 350 400

Wolf-Fritz Riekert



Dies ist meine *Homepage*

[HdM Stuttgart](#)

<body> <h1>

100% 429 x 311 15 K /

Titel: Riekerts Homepage Datum: 07 Protokoll...

CS3

- CSS
- Anwendung
- Tag-Inspektor
- ▼ **Dateien**

Dateien Elemente Codefragmente

mars Lokale Ansicht

Lokale Dateien	Gr...	Typ	Geände
create-blackb...	2KB	SQL-D...	22.05.20
div.php	1KB	PHP S...	19.10.20
divform.html	1KB	Firefox...	19.10.20
donald.jpg	27KB	IrfanVi...	20.10.20
hello.php	1KB	PHP S...	22.11.20
homepage.html	1KB	Firefox...	20.10.20
index.html	1KB	Firefox...	07.10.20
index.php	2KB	PHP S...	22.11.20
lehrveranstalt...	1KB	Firefox...	07.10.20
mailtest.php	1KB	PHP S...	04.12.20
mysql-zugang...	1KB	Textd...	03.04.20
quadratzahlfo...	1KB	Firefox...	04.11.20
quadratzahlte...	1KB	PHP S...	22.11.20
riekert.jpg	14KB	IrfanVi...	07.10.20